

Studies in Computational Intelligence 1147

Michael Karner · Johannes Peltola ·
Michael Jerne · Lukas Kulas ·
Peter Priller *Editors*

Intelligent Secure Trustable Things

OPEN ACCESS

 Springer

Studies in Computational Intelligence

Volume 1147

Series Editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland

The series “Studies in Computational Intelligence” (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution, which enable both wide and rapid dissemination of research output.

Indexed by SCOPUS, DBLP, WTI AG (Switzerland), zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

Michael Karner · Johannes Peltola · Michael Jerne ·
Lukas Kulas · Peter Priller
Editors

Intelligent Secure Trustable Things

 Springer

Editors

Michael Karner
Virtual Vehicle Research GmbH
Graz, Austria

Michael Jerne
NXP Semiconductors Austria
GmbH & Co. KG
Gratkorn, Austria

Peter Priller
AVL List GmbH
Graz, Austria

Johannes Peltola
VTT Technical Research Centre
of Finland Ltd.
Oulu, Finland

Lukas Kulas
Politechnika Gdańska
Gdansk, Poland



ISSN 1860-949X

ISSN 1860-9503 (electronic)

Studies in Computational Intelligence

ISBN 978-3-031-54048-6

ISBN 978-3-031-54049-3 (eBook)

<https://doi.org/10.1007/978-3-031-54049-3>

© The Editor(s) (if applicable) and The Author(s) 2024. This book is an open access publication.

Open Access This book is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this book are included in the book's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the book's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Paper in this product is recyclable.

Acknowledgements

InSecTT has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876038. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Sweden, Spain, Italy, France, Portugal, Ireland, Finland, Slovenia, Poland, Netherlands, and Turkey.



The document reflects only the author's view, and the Commission is not responsible for any use that may be made of the information it contains.

The publication was edited and partially written at Virtual Vehicle Research GmbH, Graz, Austria. Virtual Vehicle Research GmbH has received funding within COMET Competence Centers for Excellent Technologies from the Austrian Federal

Ministry for Climate Action, the Austrian Federal Ministry for Labour and Economy, the Province of Styria (Dept. 12), and the Styrian Business Promotion Agency (SFG). The Austrian Research Promotion Agency (FFG) has been authorized for the programme management.

We would like to thank all members of the InSecTT consortium for their fruitful cooperation in the project.

Contents

Introduction

Going to the Edge: Bringing Artificial Intelligence and Internet of Things Together	3
Michael Karner and Joachim Hillebrand	
The Development of Ethical and Trustworthy AI Systems Requires Appropriate Human-Systems Integration	11
Peter Moertl and Nikolai Ebinger	
The InSecTT Reference Architecture	29
Ramiro Samano-Robles	
Structuring the Technology Landscape for Successful Innovation in AIoT	61
Peter Priller and Michael Jerne	

Technology Development

InSecTT Technologies for the Enhancement of Industrial Security and Safety	83
Sasikumar Punnekkat, Tijana Markovic, Miguel León, Björn Leander, Alireza Dehlaghi-Ghadim, and Per Erik Strandberg	
Algorithmic and Implementation-Based Threats for the Security of Embedded Machine Learning Models	105
Pierre-Alain Moëllic, Mathieu Dumont, Kevin Hector, Christine Hennebert, Raphaël Joud, and Dylan Paulin	
Explainable Anomaly Detection of 12-Lead ECG Signals Using Denoising Autoencoder	127
Rok Hribar and Drago Torkar	
Indoor Navigation with a Smartphone	141
Drago Torkar	

Reconfigurable Antennas for Trustable Things	151
Mateusz Groth, Mateusz Rzymowski, Krzysztof Nyka, and Lukasz Kulas	
AI-Enhanced Connection Management for Cellular Networks	169
Bernd-Ludwig Wenning	
Vehicle Communication Platform to Anything-VehicleCAPTAIN	185
Christoph Pilz	
AI-Enhanced UWB-Based Localisation in Wireless Networks	201
Eshagh Dehmollaian, Bernhard Etzlinger, Philipp Peterseil, and Andreas Springer	
Industrial Applications	
Approaches for Automating Cybersecurity Testing of Connected Vehicles	219
Stefan Marksteiner, Peter Priller, and Markus Wolf	
Solar-Based Energy Harvesting and Low-Power Wireless Networks	235
Leander B. Hörmann, Julian Karoliny, and Philipp Peterseil	
Location Awareness in HealthCare	253
Frank van de Laar and Karin Klabunde	
Driver Distraction Detection Using Artificial Intelligence and Smart Devices	285
Efi Papatheocharous, David Buffoni, Matthias Maurer, Anders Wallberg, and Gonzalo Ezquerro	
Working with AIoT Solutions in Embedded Software Applications. Recommendations, Guidelines, and Lessons Learned	309
Christina Gratorp	
Artificial Intelligence for Wireless Avionics Intra-Communications	331
Ramiro Samano Robles, R. Venkatesha Prasad, Ad Arts, Mateusz Rzymowski, and Lukasz Kulas	
Use of Artificial Intelligence as an Enabler for the Implementation of ETCS L3 and Other Innovative Rail Services	353
Francisco Parrilla Ayuso, Jose Manuel González Delgado, Jose Antonio Giménez Gómez, Jorge Rubio Cañete, Alejandro Díaz Díaz, Rogelio Hernandez, Jaime Señor, Gabriel Mujica, Andrés Otero, Jorge Portilla, Jesús Félez, Miguel A. Vaquero Serrano, Arrate Alonso Gómez, Bernd-Ludwig Wenning, and Gonzalo Ezquerro	

Innovative Solutions for Maritime Infrastructures Monitoring and Protection 395
Francesco Pacini, Daniele Marroccella, Antonio Lagudi, Manuel Fortunato Drago, Francesco Buffone, Floriano De Rango, and Fabio Bruno

Security of Wireless IoT in Smart Manufacturing: Vulnerabilities and Countermeasures 419
Fatima Tu Zahra, Yavuz Selim Bostanci, and Mujdat Soyuturk

Contributors

Arrate Alonso Gómez Escuela Politécnica Superior de Mondragón
Unibertsitatea (EPS-MU), Mondragón, Spain

Ad Arts NXP Semiconductors, Eindhoven, The Netherlands

Yavuz Selim Bostanci Vehicular Networking and Intelligent Transportation
Systems Research Lab, Marmara University, Istanbul, Turkey

Fabio Bruno Department of Mechanical, Energy and Management Engineering
(DIMEG), University of Calabria, Arcavacata di Rende, Italy

Francesco Buffone Department of Informatics, Modelling, Electronics and
System Engineering (DIMES), University of Calabria, Arcavacata di Rende, Italy

David Buffoni Tietoevry, Stockholm, Sweden

Alireza Dehlaghi-Ghadim RISE, Västerås, Sweden

Eshagh Dehmollaian JKU Linz, Linz, Austria

Floriano De Rango Department of Informatics, Modelling, Electronics and
System Engineering (DIMES), University of Calabria, Arcavacata di Rende, Italy

Manuel Fortunato Drago Department of Mechanical, Energy and Management
Engineering (DIMEG), University of Calabria, Arcavacata di Rende, Italy

Mathieu Dumont CEA-LETI, Grenoble, France

Alejandro Díaz Díaz Indra Sistemas S.A. (INDRA), Madrid, Spain

Nikolai Ebinger Virtual Vehicle Research GmbH, Graz, Austria

Bernhard Etzlinger JKU Linz, Linz, Austria

Gonzalo Ezquerro JIG Advanced Solutions, Logroño, Spain;
JIG Internet Consulting SL (JIG), Logroño, Spain

Jesús Félez Universidad Politécnica de Madrid (UPM), Madrid, Spain

- Jose Antonio Giménez Gómez** Indra Sistemas S.A. (INDRA), Madrid, Spain
- Jose Manuel González Delgado** Indra Sistemas S.A. (INDRA), Madrid, Spain
- Christina Gratorp** Realtime Embedded AB, Stockholm, Sweden
- Mateusz Groth** Politechnika Gdańska, Gdansk, Poland
- Kevin Hector** CEA-LETI, Grenoble, France
- Christine Hennebert** CEA-LETI, Grenoble, France
- Rogelio Hernandez** Universidad Politécnica de Madrid (UPM), Madrid, Spain
- Joachim Hillebrand** Virtual Vehicle Research GmbH, Graz, Austria
- Rok Hribar** Jožef Stefan Institute, Ljubljana, Slovenia
- Leander B. Hörmann** Linz Center of Mechatronics GmbH, Altenberger Straße 69, Linz, Austria
- Michael Jerne** NXP Semiconductors Austria GmbH & Co. KG, Gratkorn, Austria
- Raphaël Joud** CEA-LETI, Grenoble, France
- Michael Karner** Virtual Vehicle Research GmbH, Graz, Austria
- Julian Karoliny** Silicon Austria Labs, Altenberger Straße 66c, Linz, Austria
- Karin Klabunde** Philips Research, Eindhoven, Netherlands
- Lukasz Kulas** Politechnika Gdańska, Gdańsk, Poland
- Frank van de Laar** Philips Research, Eindhoven, Netherlands
- Antonio Lagudi** Department of Mechanical, Energy and Management Engineering (DIMEG), University of Calabria, Arcavacata di Rende, Italy
- Björn Leander** ABB AB, Västerås, Sweden
- Miguel León** Mälardalen University, Västerås, Sweden
- Tijana Markovic** Mälardalen University, Västerås, Sweden
- Stefan Marksteiner** AVL List GmbH, Graz, Austria;
Mälardalen University, Västerås, Sweden
- Daniele Marrocella** Leonardo S.P.A, Electronic Division—L.o.B. Underwater Armaments and Systems, Livorno/Pozzuoli, Italy
- Matthias Maurer** Virtual Vehicle Research GmbH (VIF), Graz, Austria
- Pierre-Alain Moëllic** CEA-LETI, Grenoble, France
- Peter Moertl** Virtual Vehicle Research GmbH, Graz, Austria
- Gabriel Mujica** Universidad Politécnica de Madrid (UPM), Madrid, Spain

- Krzysztof Nyka** Politechnika Gdańska, Gdansk, Poland
- Andrés Otero** Universidad Politécnica de Madrid (UPM), Madrid, Spain
- Francesco Pacini** Leonardo S.P.A, Electronic Division—L.o.B. Underwater Armaments and Systems, Livorno/Pozzuoli, Italy
- Efi Papatheocharous** Research Institutes of Sweden (RISE), Kista, Sweden
- Francisco Parrilla Ayuso** Indra Sistemas S.A. (INDRA), Madrid, Spain
- Dylan Paulin** CEA-LETI, Grenoble, France
- Philipp Peterseil** Johannes Kepler University, Altenberger Straße 69, Linz, Austria
- Christoph Pilz** Virtual Vehicle Research GmbH, Graz, Austria
- Jorge Portilla** Universidad Politécnica de Madrid (UPM), Madrid, Spain
- Peter Priller** AVL List GmbH, Graz, Austria
- Sasikumar Punnekkat** Mälardalen University, Västerås, Sweden
- Ramiro Samano Robles** Research Centre in Real-Time and Embedded Computing Systems, Polytechnic Institute of Porto, Porto, Portugal
- Jorge Rubio Cañete** Indra Sistemas S.A. (INDRA), Madrid, Spain
- Mateusz Rzymowski** Politechnika Gdańska, Gdańsk, Poland
- Ramiro Samano-Robles** CISTER Research Centre on Real Time and Embedded Computing Systems, Polytechnic Institute of Porto, Porto, Portugal
- Jaime Señor** Universidad Politécnica de Madrid (UPM), Madrid, Spain
- Mujdat Soyturk** Department of Computer Engineering, Marmara University, Istanbul, Turkey
- Andreas Springer** JKU Linz, Linz, Austria
- Per Erik Strandberg** Westermo Network Technologies AB, Västerås, Sweden
- Drago Torkar** Computer Systems Department, Jožef Stefan Institute, Ljubljana, Slovenia
- Miguel A. Vaquero Serrano** Universidad Politécnica de Madrid (UPM), Madrid, Spain
- R. Venkatesha Prasad** Faculty of Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands
- Anders Wallberg** Research Institutes of Sweden (RISE), Kista, Sweden
- Bernd-Ludwig Wenning** Munster Technological University, Bishopstown, Cork, Ireland

Markus Wolf AVL List GmbH, Graz, Austria

Fatima Tu Zahra Vehicular Networking and Intelligent Transportation Systems
Research Lab, Marmara University, Istanbul, Turkey

Abbreviations

1D	One-Dimensional
2D	Two-Dimensional
3D	Three-Dimensional
AAA	Authentication, Authorization, and Accounting
AAMP	Additive Angular Margin Penalty
ACS	Adaptable Communication System
AES256	Advanced Encryption Standard, key size 256 bits
AI	Artificial Intelligence
AIoT	Artificial Intelligence of Internet of Things
API	Application Programming Interface
APS	Autonomous Positioning System
ASIL	Automotive Safety Integrity Level (classification system defined by the ISO 26262)
ASVS	Applications Security Verification Standard
AWGN	Additive White Gaussian Noise
BB	Building Block
BF	Business Finland (a national Finnish government organization for innovation funding and trade, travel and investment promotion)
BLE	Bluetooth-Low-Energy
BPM	Beats Per Minute
CANbus	Controller Area Network bus
CCM	Client Connection Manager
CEPT	Conference of European Posts and Telegraphs
CIR	Channel Impulse Response
CNN	Convolutional Neural Network
COTS	Commercial-Off-The-Shelf
CPSoS	Cyber-Physical System of Systems
CPU	Central Processing Unit
CRUD	Create, Read, Update, and Delete
D	Deliverable

DNN	Deep Neural-Networks
DOA	Direction of Arrival
DoS/DDoS	Denial of Service/Distributed Denial of Service
DSRC	Dedicated Short Range Communication
ECG	Electrocardiogram
EDF	European Defence Funds
ELSE	Emergency Logistics SERVICES
EMR	Electronic medical records
eMRTD	Electronic Machine-Readable Travel Document
ETSI	European Telecommunications Standards Institute
FHIR	Fast Healthcare Interoperability Resources
FMCW	Frequency Modulated Continuous Wave
FMEA	Failure Modes and Effects Analysis
FPS	Frames per Second
FR	Face Recognition
FRS	Face Recognition System
FW	Firmware
GA	Grant Agreement
GAN	Generative Adversarial Network
GDPR	General Data Protection Regulation
GeoJSON	Geographical JavaScript Object Notation
GMM	Gaussian Mixture Model
GPS	Global Positioning System
GPU	Graphical Processing Unit
GRU	Gated Recurrent Unit (a variant of LSTM)
GUI	Graphical User Interface
GUT	Gdańsk University of Technology
HD	High-Definition
HIPAA	Health Insurance Portability and Accountability Act
HL7	Health Level 7
HLA	High Level Architecture
HR	Heart Rate
HRV	Heart Rate Variability
HTML	HyperText Markup Language (https://html.spec.whatwg.org/)
HTTPS	Hypertext Transfer Protocol Secure
HW	Hardware
IDS	Intrusion Detection Systems
IEEE	Institute of Electrical and Electronics Engineers
IMU	Inertial Measurement Unit
IoT	Internet of Things
IR2SAM	Intelligent Rail-Road Shared Areas Management
ISM	Industrial, Scientific, and Medical radio band (a group of radio bands or parts)
ISO	International Organization for Standardization
ITS	Intelligent Transport Systems

iWSN	intelligent Wireless Sensor Network
JSON	JavaScript Object Notation
JTC	Joint Technical Committee
KPI	Key Performance Indicator
LED	Light Emitting Diode
LiDAR	Light Detection And Ranging/Laser Imaging, Detection, and Ranging
LPWAN	Low Power Wide Area Network
LSTM	Long Short-Term Memory (a neural network algorithm)
LTE	Long Term Evolution
MABASR	Multi-Armed Bandit Adaptive Similarity-based Regressor
MAC	Media Access Control
MAMs	Multi-Access Management Services
MCU	Microcontroller Unit
MEC	Multi-access Edge Computing
MFCC	Mel-Frequency Cepstral Coefficients
MIG	Multi-Interface Gateway
MIL	Multiple Instance Learning (a type of weakly supervised machine learning algorithm)
MIMO	Multiple Input Multiple Output
MitM	Man-in-the-middle
ML	Machine Learning
MoM	Minutes of Meeting
MOT	Multiple Object Tracking
MP2MP	Multipoint-to-Multipoint (Label Switched Paths (LSPs) in MPLS networks)
MQTT	Message Queue Telemetry Transport
MSE	Mean Squared Error
N/A	Not applicable
NB-IoT/LTE-M	Narrow Band IoT/Long Term Evolution for Machines
NCM	Network Connection Manager
NFC	Near Field Communication
NIR	Near-Infrared Radiation
NN	Neural Network
NOMA	Non-Orthogonal Multiple Access
NR	New Radio
NR_RSRQ	New Radio Reference Signal Received Quality
NTRU	N-th degree Truncated polynomial Ring Units (A ring-based public key cryptosystem)
OFDM	Orthogonal Frequency Division Multiplexing
OOBM	Out of Band Management
OPC	Open Platform Communications; an interoperability standard for the secure and reliable exchange of data in the industrial automation space and in other industries
OPC UA	Open Platform Communications

OPC-UA	Open Platform Communications United Architecture
OpenSSL	An open-source implementation of the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocol
OTA	Over-The-Air
P2P	Peer to Peer
PCA	Principal Component Analysis
PDR	Pedestrian Dead Reckoning
PHY	Physical Layer
PM	Phase Modulation
PMs	Person Months
PoC	Proof of Concept
PPE	Personal Protective Equipment
PPG	Photoplethysmography
PSIM	Physical Security Information Management
QoE	Quality of Experience
QoS	Quality of Service
QR	Quick Response code (a type of matrix barcode)
RadCom	Joint Radar and Communication
RAT	Radio Access Technology
REST	REpresentational State Transfer
REST API	REpresentational State Transfer Application Programming Interface
RESTful	REpresentational State Transfer and an architectural style for distributed
RF	Radio Frequency
RFC	Request for Comments (RFC) by the Internet Engineering Task Force (IETF)
RGB	Red-Green-Blue colour model for images and video
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
rPPG	Remote Photoplethysmography
RR	Respiratory Rate
RSRQ	Reference Signal Received Quality
RSSI	Received Signal Strength Indicator
RSU	Restricted Stock Unit
RUL	Remaining Useful Life
SAMC	Smart Adaptation Movement Control
SCN	Smart Communication Node
SDK	Software Development Kit
SDN	Software Defined Network
SDR	Software Defined Radio
SEA	Safe Error Attack
SNMP	Simple Network Management Protocol
SNR	Signal-to-Noise Ratio
SoA	State-of-the-Art

SODAQ	Solar Data Acquisition
SoTA	State of The Art
SRAS	Smart Rail Automation System
SSL	Secure Sockets Layer
STD	Standard Deviation
STCC	Smart Train Coupling Composition
SVR	Support Vector Regression
SW	Software
T	Task
TBB	Technology Building Block
T-CAM	Temporal Class Activation Map
TCN	Temporal Convolutional Network
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TDOA	Time Difference Of Arrival
TLS	Transport Layer Security
TN	Terminal Node
TOA	Time Of Arrival
TRL	Technology Readiness Level ¹⁰
UA	Unified Architecture
UC	Use Case
UC(s)	Use Case(s)
UI	User Interface
UID	Unique Identifier
USB	Universal Serial Bus
UTM	Universal Transverse Mercator
UWB	Ultra Wide Band
V&V	Verification and Validation
V2V	Vehicle to Vehicle
V2X	Vehicle to Anything
VAR	Vector Autoregression
WAICS	Wideband Acoustic Imaging Classification System
WG	Working Group
Wi-Fi	IEEE 802.11 wireless communications protocol
WP	Work Package
WSN	Wireless Sensor Network
XAI	eXplainable AI

Introduction

Going to the Edge: Bringing Artificial Intelligence and Internet of Things Together



Michael Karner and Joachim Hillebrand

Abstract Artificial Intelligence of Things (AIoT) is the natural evolution for both Artificial Intelligence (AI) and Internet of Things (IoT) because they are mutually beneficial. AI increases the value of the IoT through Machine Learning by transforming the data into useful information, while the IoT increases the value of AI through connectivity and data exchange. Therefore, InSecTT—Intelligent Secure Trustable Things, a pan-European effort with 52 key partners from 12 countries (EU and Turkey), provides intelligent, secure, and trustworthy systems for industrial applications. This results in comprehensive cost-efficient solutions of intelligent, end-to-end secure, trustworthy connectivity and interoperability to bring the Internet of Things and Artificial Intelligence together. InSecTT aims at creating trust in AI-based intelligent systems and solutions as a major part of the AIoT. This article provides an overview of the concept and ideas behind InSecTT, serving as a baseline for all subsequent chapters and articles.

Keywords Artificial intelligence · Internet of things · Artificial intelligence of things · InSecTT · Intelligent secure trustworthy things · Trustworthiness · Ethics · Industrial application

1 Introduction

The Internet of Things (IoT) is a revolutionary change for many sectors: Fitness trackers measure our movements, smart fire extinguishers monitor their own readiness for action, and cars turned out to become fully connected vehicles. The availability of the collected data goes hand in hand with the development of Artificial

© [2021] IEEE. Reprinted, with permission, from Michael Karner, Joachim Hillebrand, Manuela Klocker, Ramiro Samano-Robles; "Going to the Edge - Bringing Internet of Things and Artificial Intelligence Together"; 24th Euromicro Conference on Digital System Design (DSD); 2021.

M. Karner (✉) · J. Hillebrand
Virtual Vehicle Research GmbH, Inffeldgasse 21a, Graz, Austria
e-mail: michael.karner@v2c2.at

© The Author(s) 2024
M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_1

Intelligence (AI) and Machine Learning (ML) algorithms to process them. Despite numerous benefits, the vulnerability of these devices in terms of security remains an issue. Hacks of webcams, printers, children’s toys, and even vacuum cleaners as well as Distributed Denial-of-service (DDoS) attacks reduce confidence in this technology. Users are also challenged to understand and trust their increasingly complex and smart devices, sometimes resulting in mistrust, usage hesitation and even rejection.

These developments mostly cover processing of data in centralized Cloud locations and hence cannot be used for applications where milliseconds matter or for safety-critical applications. By moving AI to the Edge, i.e., processing data locally on a hardware device, real-time applications for self-driving cars, robots and many other areas in industry can be enabled. The push of AI towards the Edge can also be seen by recent announcements in consumer electronics. Google has reduced the size of the Cloud-based AI voice recognition model from 2 GB to only 80 MB, so that it can also be used on embedded devices and does not need an Internet connection [1]. The technological race to bring AI to the Edge can also be seen by very recent developments of hardware manufacturers. For example, Google released Edge TPU [2], a custom processor to run the specific TensorFlow Lite models on Edge devices. Many other, Asian companies like EdgeCortex, or US companies like Quadric are also developing custom silicon for the Edge.

This is where InSecTT¹ weighs in. The pan-European project InSecTT (Intelligent Secure Trustable Things) provides intelligent, secure, and trustworthy systems for industrial applications as well as comprehensive, cost-efficient solutions of intelligent, end-to-end secure, trustworthy connectivity and interoperability. The project with more than 50 partners, coordinated by VIRTUAL VEHICLE, aims at creating trust in AI-based intelligent systems and solutions as a major part of the Artificial Intelligence of Things (AIoT).

The InSecTT partners believe that AIoT is the natural evolution for both AI and IoT because they are mutually beneficial [3]. AI increases the value of the IoT through Machine Learning by transforming the data into useful information knowledge, while the IoT increases the value of AI through connectivity and data exchange:

$$\text{AI} + \text{IoT} = \text{AIoT}.$$

2 Objectives

The overall objectives of InSecTT are to develop solutions for (1) Intelligent, (2) Secure, (3) Trustable (4) Things applied in (5) industrial solutions for European industry throughout the whole Supply Chain (6). More precisely:

¹ “<https://www.insectt.eu>”.

1. Providing intelligent processing of data applications and communication characteristics locally at the Edge to enable real-time and safety-critical industrial applications.
2. Developing industrial-grade secure, safe and reliable solutions that can cope with cyberattacks and difficult network conditions.
3. Providing measures to increase trust for user acceptance, make AI/ML explainable and give the user control over AI functionality.
4. Developing solutions for the Internet of Things, i.e., mostly wireless devices with energy- and processing-constraints, in heterogeneous and also hostile/harsh environments.
5. Providing re-usable solutions across industrial domains.
6. Creating a methodological approach with the Integral Supply Chain, from academic, to system designers and integrators, to component providers, applications and services developers and providers and end users.

3 Trustworthiness

The issues of ethics and public trust in deployed AI systems are now receiving significant international interest. In InSecTT, we focus on robustness and ethics, ensuring our developed systems are resilient, secure, and reliable, while prioritizing the principles of explainability and privacy. In InSecTT and its predecessor projects (DEWI, SCOTT) we have investigated this problem. As part of this book, in Chap. “[The Development of Ethical and Trustworthy AI Systems Requires Appropriate Human-Systems Integration: A White Paper](#)” we summarize the lessons learned from several years of working with industrial and research partners on developing trustworthy technologies. As result, we propose an approach for research and development of trustworthy AI systems that is based on current EU guidelines of developing ethical AI as well as the proposed EU AI act. That approach puts the human concerns and needs at the center of the development process and consists sets of concrete recommendations for how to develop trustworthy intelligent systems.

4 Building on a Sound Basis

The InSecTT project is built on the basis of the predecessor projects DEWI² and SCOTT.³ They, among others, reuse and extend the well-established DEWI Bubble concept and the related, ISO 29182-compliant multi-domain High- Level Architecture [4]. Within the DEWI project key solutions for wireless seamless connectivity

² DEWI: Dependable Embedded Wireless Infrastructure, <https://cordis.europa.eu/project/id/621353>, last accessed September 2023.

³ SCOTT: Secure Connected Trustable Things, <https://cordis.europa.eu/project/id/737422>, last accessed September 2023.

and interoperability in smart cities and infrastructures were developed. DEWI was started in March 2014 as part of the ARTEMIS Joint Undertaking and ended in April 2017. The DEWI Bubble concept, the defined DEWI High-Level Architecture, as well as the DEWI technology items have been used as starting point for systems development within SCOTT and can be seen as the continuation of DEWI technology solutions.

Complementary to DEWI, the SCOTT project put additional focus on the following aspects:

- Extending and connecting Bubbles and integrating distributed Bubbles into the Cloud.
- Extending the High-Level Architecture concerning security, trustability and Cloud integration.
- The development of safe and secure solutions for wireless distributed systems: implementing a layer where multiple Bubbles need to cooperate in deterministic (real-time) and secure way to establish systems in distributed locations.
- Elaboration of new approaches for secure distributed Cloud integration—extending DEWI High-Level Architecture.
- Developing secure and trustable applications coming from new domains such as Health and Home (besides commercial/public buildings).

InSecTT now goes a significant step further and brings Internet of Things and Artificial Intelligence together. InSecTT builds on the results of DEWI and SCOTT with the goal to:

- Bring Internet of Things and Artificial Intelligence together (“Artificial Intelligence of Things”, AIoT)
- Move AI to the edge, i.e. provide intelligent processing of data applications and communication characteristics locally at the edge to enable real-time and safety-critical industrial applications
- Develop industrial-grade secure and reliable solutions that can cope with cyberattacks and difficult network conditions
- Enable AI-enhanced wireless transmission
- Provide measures for trust for user acceptance, make AI/ML explainable and not just a black box that cannot be understood
- Provide re-usable solutions across industrial domains

5 Driven Through Industrial Applications

InSecTT utilizes a clearly use-case driven approach with use cases from different areas of high relevance to European society and industry; all these use cases are designed for a cross-domain use. InSecTT provides, implemented in 16 different AIoT use cases, cross-domain solutions for 9 industrial domains (see Fig. 1):

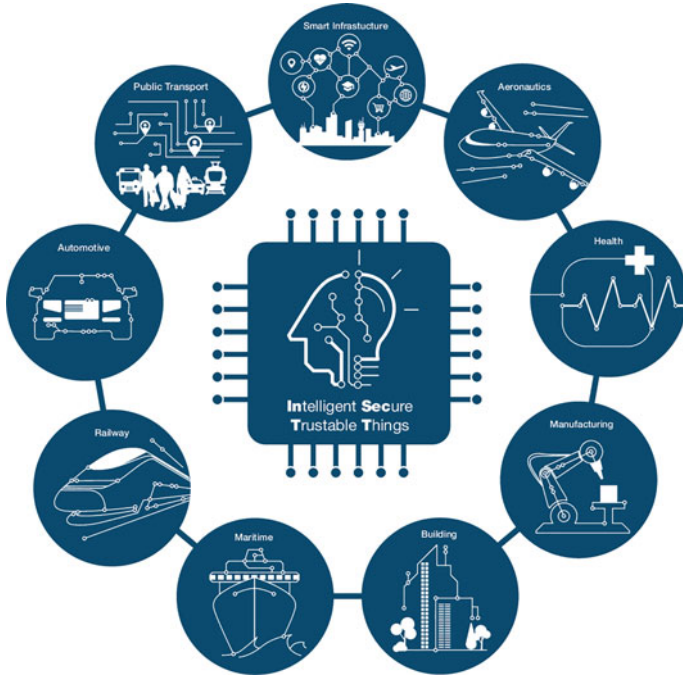


Fig. 1 Overview about the InSecTT industrial domains

- Health
- Smart Infrastructure
- Urban Public Transport
- Aeronautics
- Automotive
- Railway
- Manufacturing
- Maritime
- and Building.

The cross-domain aspect is not only realized by bringing in components to different domains, but also by interconnecting the domains in a truly cross-domain communication. This can be seen e.g., in use cases on airports or harbours, where information from buildings, vehicles and infrastructure needs to be exchanged with each other. A selection of InSecTT use cases is detailedly described in Part 3 “Industrial Applications” of this book.

6 Building Technology for Intelligent, Secure, Trustworthy Things

Based on the unique user-driven approach, the InSecTT project puts focus on:

- A representative set of Use Cases (UC) in the different domains and Technical Building Blocks (BB) jointly enabling the demonstration of business objectives in all industrial application domains.
- BBs derived from UCs (as methodologies, SW or HW components to build a SW-tool, a system, or a product, as services; as profiles; as tool or tool chains; as interfaces as well as processes). The BBs are the elements in the project, where most technical work is foreseen.
- Demonstrators: Every UC driven task and every BB must contribute to a demonstrator. This approach ensures to reach the targeted Technology Readiness Level (TRL) of InSecTT (TRL 7–8).

Figure 2 gives an example of how the use cases and building blocks are related. Each use case consists of a composition of selected BBs. In addition, each UC may have an additional UC-specific adaptation block, i.e., the necessary adjustments to form the UC. To achieve a high degree on interoperability, the definition and the requirements of the BBs are also derived from the targeted UCs.

More information on this topic is presented in Chap. “Structuring the Technology Landscape for Successful Innovation in AIoT”.

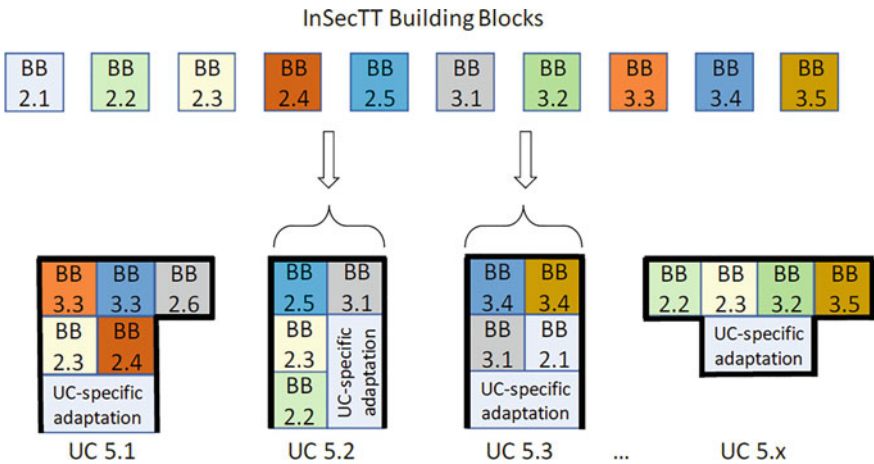


Fig. 2 Relation of Building Blocks (BB) and Use Cases (UC) (Exemplary Illustration)

7 Reference Architecture for Trustworthy AIoT

The InSecTT Reference Architecture (RA) is the set of guidelines for infrastructure organization of IoT use cases targeting industrial-grade connectivity, security, dependability, interoperability and trustworthiness with the help of AI. It provides the high-level view of building blocks, interfaces, vulnerabilities, security solutions, protocols, and in general the detailed information/control flow of InSecTT use cases in different industrial domains (aeronautics, automotive, railway, building, health-care, maritime, etc.). This provides us with a tool to analyse reusability, standardization, certification, and verification issues across domains. The InSecTT RA hosts a set of best practices collected across three EU projects: DEWI, SCOTT and InSecTT. The DEWI RA focused on dependability, using IoT protocols as a method to provide interoperability using the concept of DEWI Bubble as the encapsulation of legacy infrastructure. The DEWI RA was built on top of the ISO SNRA (Sensor Network Reference Architecture) [4]. The SCOTT project saw the extension towards a full IoT architecture with high level aspects such as Edge/Fog processing, security, privacy, safety and trustworthiness combining multiple standard architectures. The InSecTT RA re-takes the DEWI/SCOTT frameworks and the Bubble to investigate the impact of AI on IoT architectures, particularly on the standard views or perspectives of the system. The result is an extended 3D functionality model that captures the ability of AIoT systems to host functionalities related to AI such as learning, adaptation, feature extraction, detection, etc. The 3D extension also contains the type of application and the sub-building structure studied in the project. The RA also defines the relations of this new AI perspective with the rest of the views, particularly the physical entity model and the domain model, where we identified the need for an overall approach to organize, manage and schedule distributed leaning resources in a secure and trustworthy manner. The mapping of all sub-building blocks revealed important aspects such as the stress in some interfaces that need to consider the future growth and the demand of learning layers for lower layer information.

More information on this topic is presented in Chap. “[The InSecTT Reference Architecture](#)”.

8 Summary

In this article, the importance of bringing together Artificial Intelligence and the Internet of Things was highlighted. This so-called Artificial Intelligence of Things is their natural evaluation, enabling key developments based on constant interplay and integration between AI and IoT. The European project InSecTT was described as a key enabler for the AIoT. After a motivation and analysis of the initial situation, the overall objectives and goals of the project were discussed in detail. It develops intelligent, secure and trustworthy systems for industrial applications to provide comprehensive cost-efficient solutions of intelligent, end-to-end secure, trustworthy

connectivity and interoperability for the AIoT. The Reference Architecture allows to deliver a more secure AIoT solution with reduced design effort, decreased costs, and increased quality. In the following chapters, details about the results generated through both technology development as well as industrial applications are shown. In addition, as part of the section “Introduction”, deep insight into the topics of development of trustworthy AI systems, the InSecTT reference architecture and how to structure the technology landscape for successful innovation in AIoT is provided.

Acknowledgements InSecTT has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876038. The JU receives support from the European Union’s Horizon 2020 research and innovation programme and Austria, Sweden, Spain, Italy, France, Portugal, Ireland, Finland, Slovenia, Poland, Netherlands, Turkey.

The document reflects only the author’s view, and the Commission is not responsible for any use that may be made of the information it contains.

Virtual Vehicle Research GmbH has received funding within COMET Competence Centers for Excellent Technologies from the Austrian Federal Ministry for Climate Action, the Austrian Federal Ministry for Labour and Economy, the Province of Styria (Dept. 12) and the Styrian Business Promotion Agency (SFG). The Austrian Research Promotion Agency (FFG) has been authorised for the programme management.

References

1. ZDNet: Move over Siri, Alexa: Google’s offline voice recognition breakthrough cuts response lag. <https://www.zdnet.com/article/move-over-siri-alex-google-offline-voice-recognition-breakthrough-cuts-response-lag/>. Accessed 13 March 2019, last accessed September 2023
2. Google: Edge TPU. <https://cloud.google.com/edge-tpu/>. Accessed September 2023
3. Karner, M., Hillebrand, J., Klocker, M., Sámano-Robles, R.: Going to the edge—bringing internet of things and artificial intelligence together. In: 2021 24th Euromicro Conference on Digital System Design (DSD), Palermo, Italy, 2021, pp. 295–302. <https://doi.org/10.1109/DSD53832.2021.00052>
4. Sámano-Robles, R., Nordström, T., Kunert, K., Santonja-Climent, S., Himanka, M., Liuska, M., Karner, M., Tovar, E.: The DEWI high-level architecture: wireless sensor networks in industrial applications. *Technologies* **9**, 99 (2021). <https://doi.org/10.3390/technologies9040099>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



The Development of Ethical and Trustworthy AI Systems Requires Appropriate Human-Systems Integration



Peter Moertl  and Nikolai Ebinger 

Abstract The development of Artificial Intelligence (AI) technologies experiences worldwide an ongoing challenge to become trustworthy and ethical for users and the general public. This challenge currently stands between the promise of AI to create immense societal and individual impact and its realization. Because of this, possible large marketplaces still remain hesitant or closed. We have investigated this problem and identified potential solutions in the InSecTT project, a large international EU research and development project that investigates ethical, smart technologies. Thereby, working with industrial and research partners, we assert that developing trustworthy AI technologies is not foremost a technical challenge, but increasingly an organizational and process challenge that results from applying traditional ways of conceiving, designing, and selling technologies to technologies with very new types of user and societal implications. Because AI technologies can shift the role that humans and society see as acceptable, traditional development processes that rely on the strict separation of specialties are overburdened. In our view, a research and development approach for trustworthy AI systems should put human concerns and needs at the start of the development process to effectively integrate humans and systems and thereby have a chance to meet EU guidelines for developing ethical AI. Our proposed approach to develop such ethical, trustworthy AI systems centers around the assessment of trustworthiness risks through intensive user involvement prior to the elicitation of system requirements that are then managed throughout the system's life cycle. Also, the approach includes concrete recommendations to establish the organizational prerequisites for producing ethical, trustworthy AI systems. In this chapter, we describe the Human-Systems Integration (HSI) approach and motivate the underlying principles in some detail. The chapter intends to inform managers of technical organizations and product managers, as well as principal investigators, to set up the prerequisites for trustworthy AI, while also soliciting inputs for further refinement and discussion.

Keywords Trustworthy AI · Human-centered process · Ethical AI

P. Moertl (✉) · N. Ebinger
Virtual Vehicle Research GmbH, Inffeldgasse 21a, 8010 Graz, Austria
e-mail: Peter.Moertl@v2c2.at

Abbreviation

HIS Human Systems Integration

1 Is Trustworthiness of AI a Problem?

The market for smart technologies that utilize Artificial Intelligence (AI) is growing at high rates,¹ but uptake of these technologies is lagging behind due to increasing lack of user trust and acceptance of these technologies. Among many examples, in 2020, Amazon had to stop selling its face recognition software due to racially biased categorizations and Microsoft soon followed suit.² Similarly, the German Bundesnetzagentur warned against the use of certain intelligent toys because of potential spying on customers.³ There are many examples that have triggered debates on how to create ethical AI systems that are consistent with the interests and rights of their users. This emphasizes the drawbacks of smart technologies that result from the often unintended consequences of users interacting with them in the real world.

Smart technologies often shift the roles and responsibilities of those surrounding them and these impacts are often difficult to see beforehand. A single police officer with biased racial views is one thing but an automated facial recognition system with such biased views shifts the problem to another level as hundreds of thousands of biased categorizations could occur.

Thereby we define a trustworthy smart system as one that facilitates its user’s trust as “... *the attitude that an agent [smart system] will help achieve an individual’s goals in a situation characterized by uncertainty and vulnerability*” [1]. Trustworthy systems must be lawful, ethical, and robust [2], which makes ethical compliance a necessary but not sufficient precondition for trustworthiness.

Based on the recently proposed EU AI act [3], we consider an ethical system as one that honors the rights of the human user and thereby conforms to moral principles, specifically ensuring the autonomy of humans to decide for themselves, not exploit their data and rights for other purposes, and ultimately serve the human, rather than purely those selling products.⁴

¹ <https://www.insect.eu/>.

² <https://www.reuters.com/technology/exclusive-amazon-extends-moratorium-police-use-facial-recognition-software-2021-05-18/>.

³ <https://www.heise.de/news/Smart-Toys-Bundesnetzagentur-warnt-vor-Spionage-Spielzeug-6300179.html>.

⁴ The term ethical is commonly defined as conforming to “*the moral principles that govern a person’s behaviour or how an activity is conducted*” [4]. This definition is rather abstract and to create a helpful understanding for this context, we select, based on the EU guidelines for trustworthy AI, the moral principles of “autonomy”, “privacy”, and “human well-being and diversity” as most critical enablers of ethics. Examples may be useful to exemplify this: A system that changes the human role from an active, decision making role into a passive, reactive role is, according to this

“Smart” functions increasingly take over the cognitive and manual tasks that previously only humans could perform. Smart systems recognize preferences, understand voice commands, keep the distance to the vehicle driving ahead, or provide diagnostic information to medical doctors. Thereby in most cases beyond the most repetitive and simple environments, smart systems do not usually replace the human operator but assist them in their work and daily life. Full automation is often too expensive and not realistic, given liability concerns. A survey among 500 decision-makers from German companies shows that smart AI systems are mostly intended to assist human operators but not to replace them [5]. Whereas fully automated systems are often implied by public debates about the capabilities of AI, the limits are often not clearly stated and fiction and reality are blurred. Thereby, AI-developing organizations are motivated to overstate the capabilities of their smart technologies. This can result in misunderstandings about their actual ability and produce societal backlashes against the technology without understanding the real use cases. Because smart systems cannot yet decide about their contextual enabling and disabling, solve untrained situations, or take responsibility for failures, realistic use cases usually involve the human operator to decide in these situations. This is the case in many different operational domains such as medicine or security: a medical diagnostic system can help the medical doctor during the diagnosis process but not take responsibility for medical decision making; an effective security system can smartly detect anomalies within large data sets and then inform the operator with the needed information to resolve a security breach. These represent new roles for the human and the smart technology needs to be designed for such teamwork.

The main challenge of teaming AI with humans is that humans who are for some periods of time *out-of-the-loop*, are difficult to bring *back-into-the-loop*. The user of a highly automated vehicle who is watching a movie and is not aware of an imminent dangerous situation needs some time and active effort to understand the critical situation in order to take back control and safely maneuver the vehicle. This “*bringing-back-in-the-loop*” can be unsafe and cause risks of accidents or biased decisions. To get back into the loop requires the human to re-establish an understanding of the situation, either through establishing situation awareness themselves (a driver) or receiving explanatory information from the automation (e.g., a medical doctor receiving diagnosis suggestions). Out-of-the-loop problems are not new to human-system integration with complex technologies: nuclear power plants, modern aircraft, and many military applications exhibit high degrees of automation but still require humans for specific decision making. For such large systems, standard development processes have been developed to early on integrate the human role into the system design. Accidents happen if such processes are not followed such as the Chernobyl nuclear disaster in 1986 [6] and the Boeing 737 Max accidents in 2019 [7]. Also, users of end-user-devices have been observing these problems; for example, vehicle

definition, not ethical. A system that exploits sensitive data from human users for other purposes than those to whom they belong, is not ethical. Finally, a system that only serves a select few and discriminates against others, is also not ethical. In this way, a system that is ethical may be trusted, therefore, it is trustworthy.

navigation systems are known to “occasionally” lure truck drivers to tiny mountain passes due to incorrect map data: it is difficult for even professional drivers to calibrate their trust for a system that seems to work right most of the time but then occasionally fails. As automation becomes pervasive, such experiences will multiply if not appropriately designed for calibrated trust.

The described challenges have been well recognized and are starting to be addressed worldwide. We start with an overview of available guidelines and approaches toward trustworthy AI in the next section. Then we identify remaining gaps that we address by introducing our approach in the subsequent section.

2 Current Initiatives to Address Trustworthiness of AI

2.1 Guidelines and Regulations

Governments and private companies address the challenges to develop ethical and trustworthy AI by proposing guidelines. A review shows that 84 guidelines on ethical AI were published worldwide until 2019 [8]. Analysing the authors of these guidelines reveals that private companies and governments seem to have a common interest in guiding ethical AI development. Private companies (22.6%) provided the highest number of guidelines, closely followed by governmental agencies (21.4%). Furthermore, most guidelines include similar aspects. The requirements transparency, justice & fairness, non-maleficence, and responsibility are represented in a minimum of 71.4% of guidelines. The most frequently stated requirement of transparency (in 86.9%) focusses on explainability, interpretability, data use and human-AI interaction. Figure 1 provides an overview of the principles that are suggested by the AI guidelines.

An AI ethics guideline that got high attention and serve as basis for further considerations of AI ethics (e.g., in [9, 10]) was provided by the European Commission’s High-Level Expert Group (HLEG) on AI. The AI ethics guidelines focus on following seven dimensions:

- **Human agency and oversight:** AI applications should support the user’s agency, autonomy, and decision-making.
- **Technical robustness and safety:** technical robustness is central for preventing harm. AI applications should be developed to prevent risks and to ensure reliable functioning.
- **Privacy and data governance:** to prevent harm, the privacy and data need to be protected and used data sets need to be of high quality.
- **Transparency:** an AI application needs to be traceable and explainable. It should be transparent that an AI is in operation.
- **Diversity, non-discrimination, and fairness:** diversity and inclusion needs to be ensured throughout the AI application’s lifecycle.



Fig. 1 Numbers of most-frequently addressed ethical principles

- **Societal and environmental well-being:** to ensure fairness and prevent harm, the environment should be considered as stakeholder.
- **Accountability:** auditability and reporting of negative impacts are important.

Building upon the EU ethics guidelines, the European Commission proposed regulations for high-risk AI. The proposed act includes ethics aspects and is currently in the European legislative process [3]. The proposed legislation includes similar aspects of trustworthy and ethical AI and will make them mandatory for systems with high safety risks such as applications for biometric identification, critical infrastructure, or employment access. Such mandatory requirements increase the urgency of the AI industry to adopt ethics assessments in the development processes.

The presented guidelines have a large area of applicability but lack details to support implementation for which several implementation support options have been proposed that will be described next.

2.2 Implementation Support

One type of implementation support consists of checklists that are intended to allow AI developers to quickly estimate how well AI ethics are considered in their AI application. The AI ethics guidelines directly come with a checklist for the ethics criteria (ALTAI). The checklist allows developers to self-assess their AI application regarding ethics requirements. Based on the developers' responses, the assessment list provides explanations on which ethical aspects are missing. Similarly, checklists exist for software development applications like regression test selection [11].

While checklists are applied towards the end of development, the Eccola approach aims at starting ethical discussions throughout the development process. The approach summarizes different ethics guidelines to provide cards with easy understandable explanations and questions on ethical AI topics [10]. Eccola invites developers to discuss relevant topics throughout their development sprints. Concrete discussions outputs are documented, and the process aims at making developers aware of AI ethics.

The Z-Inspection approach takes the responsibility off the developers and involves interdisciplinary experts [9]. The expert team follows a process to identify AI risks and to provide development with concrete recommendations. Within the Z-Inspection process, socio-technical scenarios are used to identify ethical issues and tensions. Furthermore, the results are mapped with the AI ethics guidelines.

2.3 Observed Gaps in Current Initiatives

The guidelines toward developing ethical and trustworthy AI are comprehensive and go far beyond the amount of available implementation support that is rather limited in detail and depth. For example, while simple checklists are easy to apply by developers, they do not capture critical contextual information outside the developer's expertise. Also, the checklists are not part of a comprehensive development approach that clarifies responsibilities and tasks for the ethical outcomes of the developed AI system.

Similarly, the Eccola approach raises important ethics related questions but does not *per-se* prioritize the questions and does not provide success criteria. Also, other domain experts are not foreseen to participate which limits the outcomes purely to the perspective of the developers.

Z-Inspection provides a process of how interdisciplinary experts develop recommendations for specific AI applications, but again, no responsibilities and tasks are assigned to actually implement them. We expect that considering the ethics recommendations from the Z-Inspection will add development costs. In consequence, an organizational process to ensure the consideration of AI ethics in actual development is required.

The needs for an ethical AI development process that includes an organizational framework becomes apparent when analyzing the proposed EU AI act. Based on a review of the EU AI act, we categorized a selection of regulations (as shown in Fig. 2) and complemented them by aspects out of the EU ethics guidelines that are not included in the AI act.

The algorithm requirements bring the need for continuous engineering processes that do not end with fielding an AI system [12]. In detail, algorithm requirements specify how AI models are developed, how AI is documented, how data is handled, and how the system's activities shall be recorded. In traditional production, the manufacturers' active role ends with selling a product or, at the latest, when the product

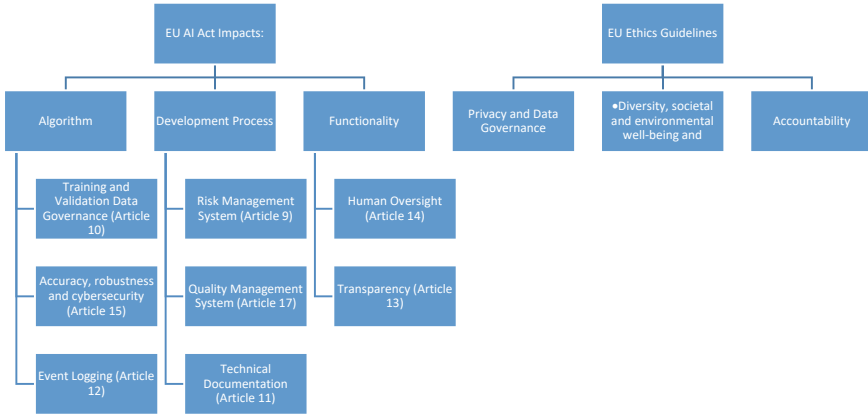


Fig. 2 Impact areas of EU AI act and EU ethics guidelines

warranty ends. However, such active role is already extended by the security requirements for software products that require continuous updates. The proposed AI act brings a further extension and makes risk and quality management throughout the system’s life cycle mandatory for high-risk AI systems.

Functionality requirements bring the need to involve users in developing AI systems from early on. The upcoming regulations require high-risk AI to be designed transparently, so that users can “interpret the system’s output and use it appropriately.” Furthermore, they shall be designed and developed so that they “can be effectively overseen by natural persons” (Human Oversight) [3]. As a consequence, functionality requirements enable the user to interpret the system’s output and oversee its activities. We argue that approaches of conducting user testing on prototypes and final systems are not sufficient *per-se* anymore because they often come too late to have a high impact on the product’s concept. Therefore, intensive involvement of stakeholders from the very beginning of AI development is necessary.

3 From Technology-Centered to Human-Centered Development of Smart Technologies

As outlined in Sects. 2 and 3, the development of trustworthy smart technologies requires shaping the development process toward the specific user and user context situations out of which ethical and trustworthiness issues only emerge. This can be difficult in traditional development environments where technology-centered development processes shape the role of users and their tasks quasi as byproducts of the technology (as shown in the block (A) in Fig. 3). This can result in “unbalanced” systems where the tasks may not be acceptable and trustworthy by users. Instead, what is needed for trustworthy development is shown to the right of Fig. 3, block (B),

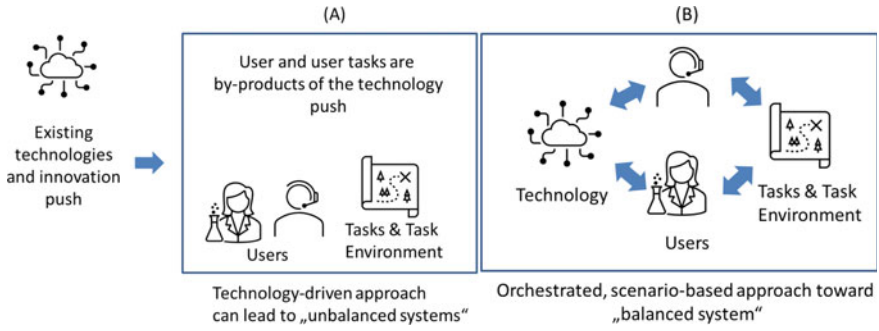


Fig. 3 Technology (a) versus orchestrated (b) processes

where the joined considerations of user, tasks, and technology, as well as task environment lead to a balanced systems where trustworthiness and acceptability are part of the whole development process. The second process seems necessary to conform with the EU AI act.

To exemplify the differences between approach (a) and (b), let's imagine a developer implementing a facial recognition system that recognizes criminals and handing it to the users, without knowing or clearly considering the possible consequences of algorithmic problems. Therefore, the technology is then transitioned to an operational use (a). As a result, the users (and the larger public) find out that the system incorrectly categorizes minority members more likely as criminals than majority members. This is unintended and results in loss of trustworthiness on the societal level. In contrast, following the approach (b), the developing organization undergoes a detailed analysis of the use-situation and extracts such risks of biases and addresses them as part of the technology development. This is, in a nutshell, what the EU AI act attempts to do.

Therefore, a key aspect to develop trustworthy systems is to sufficiently analyze and address the specific context of use and involved users and tasks early on and allow this to shape the product design and development. Specifically, the analysis of the user and use-situation should result in a set of trustworthiness risks, i.e., risks that, if not addressed, may lead to loss of trustworthiness. Those trustworthiness risks are managed in a life-long risk management process and thereby guide the product development and life-long operational process.

Moving from technology-centered to human-centered development methods requires an orchestration framework and process, as detailed in the following.

3.1 *Orchestrating the Development of Ethical and Trustworthy AI*

In order to move from traditional technology-centered approaches to human-centered approaches, we came to realize that is necessary not only to *first* establish appropriate organizational structures, but *then* also to establish the necessary processes for organizations to interact and achieve trustworthy outcomes (see e.g., [13]). The situation of developing AI resembles the development of safety and security critical systems, where many requirements originate from the use of the system in its real application context, that are often not available at design time until explicitly brought into the process through specific analytic and research activities. The safety challenges of an airplane originate in the real world of flight operations. Similarly, security breaches in the real world bring knowledge to prioritize security requirements. A similar situation results out of the use of AI systems during actual operations that make apparent the ethical and trustworthiness considerations that should have been considered during design time.

3.1.1 The Human Systems Integration Framework

Realizing trustworthy systems means therefore to “break down the silos of excellence” within which most normal technological developments currently occur. Traditional development organizations often focus their expertise on innovations at the technical level that knows little or nothing about the actual objective or mission: the experts of Machine Learning algorithms usually have no idea about the requirements of an end-user to understanding the outputs of the algorithm for his/her work environment. This is however critical for acceptance. Technological competences are necessary but not sufficient prerequisites for ethical and trustworthy products.

One approach to break down the silos of excellence is through applying the Human Systems Integration (HSI) framework that postulates three interconnecting cornerstones: (A) an organization that is able to conceptualize and investigate the use of technology within a sociotechnical context, (B) a holistic development organization that is able to identify solutions in a strong multi-disciplinary effort, and (C) a life cycle long learning and maintenance operations that addresses continuously changing aspects of the system. These cornerstones are linked via tools, processes, and standardized certification schemes that help to bound the solution space and aid collaboration and teamwork, as shown in Fig. 4.

HSI Cornerstone (A)—“Assess, Understand, Conceive” is to provide the necessary information about the intended use-situation for the development of smart systems that are to achieve sustainable acceptance and use. Such information includes the context of use, the goals to be achieved, as well as the user needs and important limitations of use and their situation, and forms an essential starting point for the system design. Technical feasibility and cost-effectiveness are thereby concept-forming factors equal to the usage situation information; this is a novelty here. Such

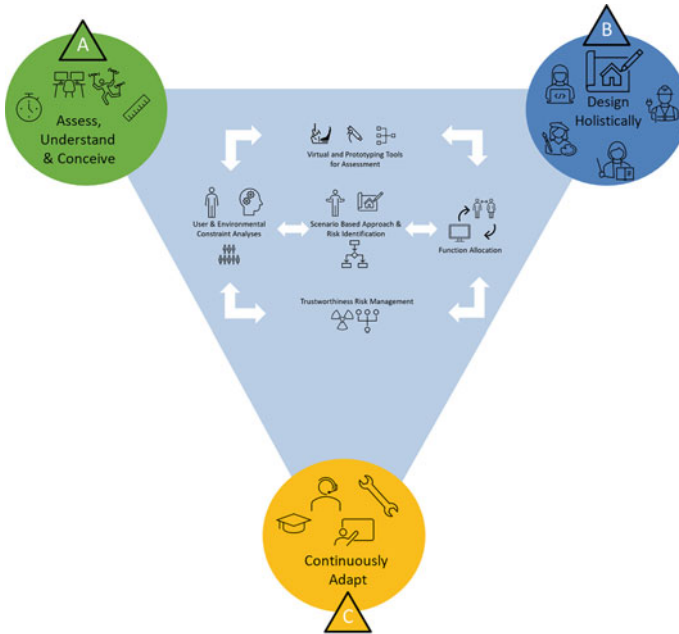


Fig. 4 HSI framework to facilitate trustworthy system’s development processes

information about the usage situation is only available to the developers of today’s systems to a limited extent. Usage situation information also includes characteristics of the user population and the tasks to be performed, including criticality, responsibilities, and influences of the organizational environment, as well as the work environment. In particular, organizational context and processes within which the system is used are important for design decisions, for example, to select appropriate methods of explaining smart technologies to the user. Data collections include observations, interviews, surveys, analyses, and especially virtual methods that allow users to make contextualized assessments (e.g., driving or flight simulators), as well as physical methods (e.g., Wizard-of-Oz studies). User and context are captured and translated into a high-level vision for how the system consisting of human, technology, and task & situation constraints could work.

HSI Cornerstone (B)—“Design Holistically” translates the vision from cornerstone (A) into a holistic design of the system. The word “holistic” means that orchestrated teams of multidisciplinary specialists work together to develop solutions across the various discipline-overarching dimensions. In the so-called “living labs” products are co-designed to achieve a trustworthy, acceptable, and safe usage of the systems. This serves as a point of convergence across the disciplines and teams. Technical and cost factors act as limiting modulators. The challenge consists of making these larger contextual perspectives visible and overcome traditionally isolated disciplinary hierarchies so that experts from different disciplines can effectively work toward

such convergence. This requires sufficiently large, multidisciplinary research environments in a climate of positive holistic goal orientation and go beyond use and stakeholder abstractions as “matchstick men” (see the block (A) in Fig. 3). Especially virtual simulation and modeling tools can support this process to combine the expertise of human factors, science, and the various technical engineering disciplines.

HSI Cornerstone (C)—“Continuously Adapt” consists of continuous adaptation and updating of products, as well as the education of users during the lifecycles of smart technologies are expected. System adaptations require detailed information about user and usage conditions. This requires a certain level of trust so that the user does not feel exploited or observed but sees himself as part of an improvement cycle. This also includes the possibility of user feedback which can not only promote user trust but also requires it. In addition to product adaptations, it is also important to promote the standardization of user knowledge and digital competencies in the form of standardized competence modules that enable users to find their way over time in what is otherwise perceived as a digitalization jungle. The creation of European training curricula for end-users, employees, and employers is a goal that must be initiated by technology developers, as this is where the critical information in the HSI process is available. The implementation of the digital competence modules in training curricula will then take place at the European level.

3.1.2 The HSI Process Model

Whereas the HSI framework postulates the organizational prerequisites for trustworthy and ethical smart technologies, how are these cornerstones stitched together into a working whole?

In Fig. 5, the HSI process model shows the orchestration of the three cornerstones. Boxes 1–4 (green) indicate the processes in cornerstone (A), including the risk management process. Boxes 5–9 indicate the activities of cornerstone (B). Box 10 indicates the activities of cornerstone (C) [14].

Critical in the HSI process model are the interactions between the three cornerstones to maintain the focus on the overall user experience concerning trustworthiness and ethical acceptability.

3.1.3 Extraction of Trustworthiness Risks Using Scenario-Based Methods

Scenario-based methods are commonly used in user-centered development efforts (e.g., [15]). Such methods can be used to identify risks by imagining the user within realistic and concrete environments risks [4]. As outlined above, the principles of ethical and trustworthy AI need to be contextualized in specific use conditions to become meaningful (see box 2 in Fig. 5). Otherwise, they are too abstract to be able to derive specific requirements for implementation.

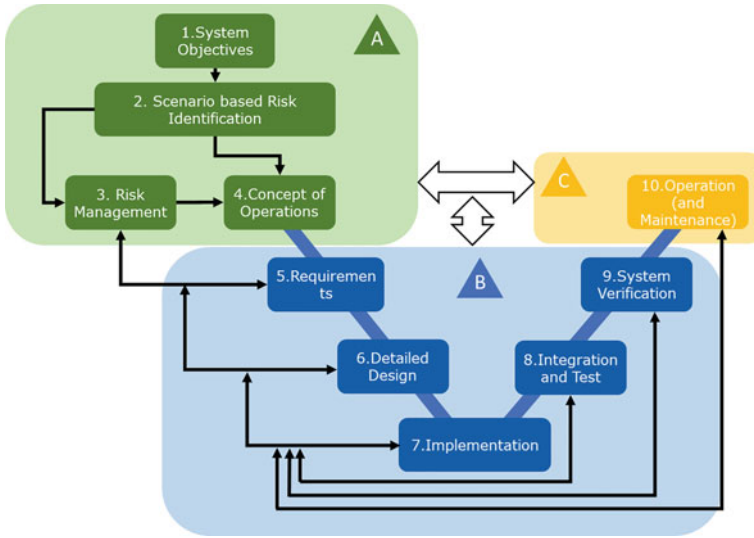


Fig. 5 HSI process model for the orchestration of trustworthy systems

With a scenario we mean here a description of how an intended function can be accomplished under a realistic set of use conditions and stakeholder characteristics. A scenario thereby makes constraints visible that remain otherwise invisible. A risk consists of the description of a situation that, if it became real, would expose an undesired danger. We suggest that risks are at its core, formulated in simple sentences containing a precondition and a consequence, for example: “If a driver is not informed about his/her responsibilities, the risk for accidents is increased.”

The scenario description is the results of an analysis in which the ethic trustworthiness criteria are asked as questions: “What could that criteria mean in a specific condition for a specific user?”. In Table 1 we give examples of how this can be done around partial driving automation (SAE Level 2 [16]) that supports driving with longitudinal and lateral control, but leaves the driver fully responsible for monitoring the assistance and stepping to manual driving anytime when needed.

How can scenarios help to identify risks? Figure 6 shows a scenario that brings out EU ethics criteria in a concrete context and a concrete user. The scenario description focuses on a specific stakeholder. To ensure diversity, different scenarios representing different drivers (young/older, gender, etc.) are required in praxis.

Table 1 Building blocks mapping to components and scenarios

EU ethics criteria for trustworthy AI	Contextualized EU ethics criteria	What does this criterium mean in a specific context?
Accountability	<p>The driver is solely responsible for the safety of the vehicle, the SAE Level 2 serves only as assistance</p> <p>The driver must clearly know the situations in that the partial driving automation is safe to use</p>	<p>Real drivers sometimes may not be aware or willing to complete their assigned roles, resulting in foreseeable errors and violations.</p> <p>In reality, drivers often do not read the vehicle manuals where the responsibilities are defined. Therefore, the scenario should describe a realistic driver, not an ideal one, i.e., a driver who may not always remember all of his/her responsibilities</p>
Human agency and oversight	<p>The human driver is responsible to monitor the driving environment and has to recognize when to take back control again</p> <p>Drivers should not use the system if it is not working reliably in a situation</p>	<p>Real drivers sometimes may not be able to complete their assigned roles of human agency and oversight, resulting in foreseeable errors.</p> <p>For example, a driver may get tired over time and not be able to keep up. Therefore, the scenario should reflect realistic driver behavior, not idealistic behavior (e.g., driver is sometimes distracted, writes text messages on the phone, etc.)</p>
Technical robustness and safety	<p>The SAE Level 2 vehicle requires sensors to adequately detect the road environment, these sensors have to be kept clean, otherwise the partial driving automation will have problems to detect the street</p>	<p>Technical robustness and safety may be accepted from a testing perspective but unacceptable from the operational perspective: e.g., when the user’s manual specifies that a system should only be used on dry roads, this shifts the problem from the system to the human who has to know the system limitations (which the user may not be aware of, willing to, or able to consider). The scenario should therefore consider technical robustness and safety from an operational user perspective</p>

(continued)

Table 1 (continued)

EU ethics criteria for trustworthy AI	Contextualized EU ethics criteria	What does this criterium mean in a specific context?
Privacy and data governance	According to the EU AI act, data logs need to be kept for safety and quality assurance Anonymized data on the partial driving automation are send to the manufacturer for improving functionality. As the data is anonymized it is not possible to relate them to a driver e.g., in case of an accident	In reality, the logged data may be used for other purposes than initially intended, for example to rate the driving behavior. The scenario should reflect how the collected data could be misused (a risk assessment how a human could be made aware about possible solutions)
Transparency	The vehicle’s automated driving modus is indicated to the driver. As drivers recognize the changed system state, they override or deactivate it to ensure a safe drive	In reality, the driver is confronted with many vehicle status lights and indicators that may make it difficult for the driver to recognize the automated driving state indication. Therefore, the scenario should describe a holistic environment within a user interacts with the system (not just the to-be-designed system <i>per-se</i>)
Diversity, non-discrimination, and fairness	A SAE Level 2 system should be able to adjust its headway to the lead vehicle based on user preferences	In reality, the adjustment of headway parameters can be hidden in many submenus and difficult to change. Therefore, the scenario should include different user groups (e.g., different age, experience, gender) to identify different preferences
Societal and environmental well-being	SAE Level 2 should assist drivers and enhance their lives by increasing their safety and comfort. In consequence it intends to have positive impact on overall road traffic	In reality, an ill-designed system may put the driver into stressful and uncomfortable situations. The scenario should explore these situations. (e.g., continuous enabling and disabling of SAE Level 2, unexpected or unacceptable driving behavior, etc.)

4 Conclusions

The key to creating trustworthy and ethical smart technologies consists of integrating humans and the system from the beginning of the system design and having the processes and organizational structures in place that allow to do so. Whereas current

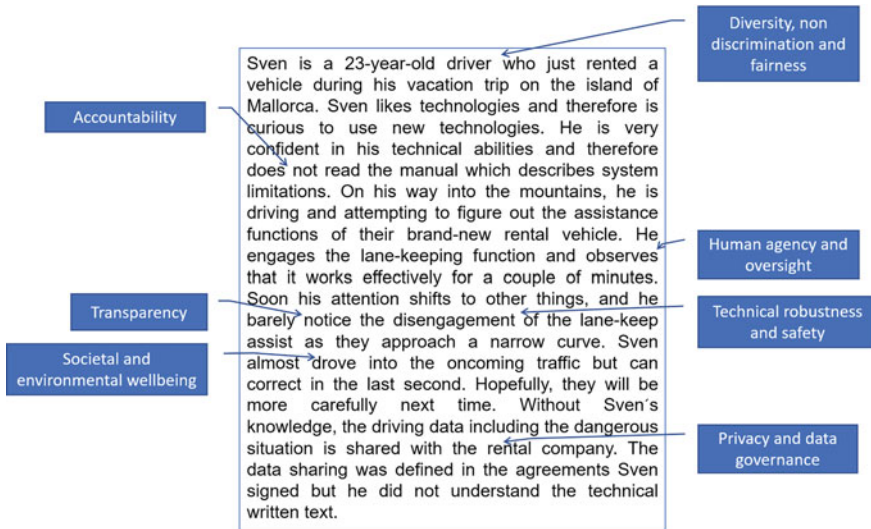


Fig. 6 Example scenario to show the link between scenario and trustworthiness criteria

AI guidelines already prepare the principles for ethical and trustworthy AI, they are lacking the implementational considerations that paramount for realizing such systems. In this chapter we described a Human-Systems Integration approach with the organizational preconditions consisting of three functional cornerstones to (A) systematically assess the context of use and insert this into the (B) holistic designs, and to (C) continuously adapt the system while keeping the human in the loop. These organizational cornerstones are interweaved using HSI processes centered around continued risk-management of trustworthiness and ethical risks. We have introduced several methods to facilitate the assessment of trustworthiness and ethical risks using scenario-based methods.

Standard engineering processes in companies of today are often rather isolated in their specialties and separate from user and use contexts. This hinders the development of trustworthy, ethical systems as foreseen by the EU rule on AI and many guidelines that have emerged worldwide. Instead of seeing such guidelines as burden and impediment, we propose that unique selling propositions can be created through offering products that are trustworthy and ethically aligned. In the end, such products are closer aligned to the needs and constraints of end users than systems that are produced without detailed knowledge of the context of use and therefore are more prone to sink within the storms of user-indignation and public outcries. We had started this chapter with a reminder of some of these and hope to have contributed methods to avoid such commercial and human mishaps while still taking advantage of the immense capabilities of AI for systems from which everybody profits.

References

1. Lee, J.D., See, K.A.: Trust in automation: designing for appropriate reliance. *Hum. Factors J. Hum. Factors Ergon. Soc.* **46**(1), 50–80 (2004)
2. High-Level Expert Group on Artificial Intelligence, *Ethics Guidelines for Trustworthy AI*. Brussels (2019)
3. European Commission: Proposal for a Regulation of the European Parliament and of the Council Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act) and Amending Certain Union Legislative Acts (2021)
4. Cahill, D.J.: CIHS White Paper: The Specification of a ‘Human Factors and Ethics’ Canvas for Socio-technical Systems, S. 18 (2020)
5. PwC: Künstliche Intelligenz in Unternehmen: Eine Befragung von 500 Entscheidern deutscher Unternehmen zum Status quo - mit Bewertungen und Handlungsoptionen von PwC (2019). <https://www.pwc.de/de/digitale-transformation/kuenstliche-intelligenz/kuenstliche-intelligenz-in-unternehmen.html>
6. Salge, M., Milling, P.M.: Who is to blame, the operator or the designer? Two stages of human failure in the Chernobyl accident. *Syst. Dyn. Rev.* **22**(2), 89–112 (2006). <https://doi.org/10.1002/sdr.334>
7. NTSB: Assumptions Used in the Safety Assessment Process and the Effects of Multiple Alerts and Indications on Pilot Performance, S. 13 (2019)
8. Jobin, A., Ienca, M., Vayena, E.: Artificial Intelligence: the global landscape of ethics guidelines. *Nat. Mach. Intell.* **1**, 389–399
9. Zicari, R.V.: Z-Inspection\circledR : A Process to Assess Trustworthy AI. *IEEE Trans. Technol. Soc.* **2**(2), 83–97 (2021). <https://doi.org/10.1109/TTS.2021.3066209>
10. Vakkuri, V., Kemell, K.-K., Jantunen, M., Halme, E., Abrahamsson, P.: ECCOLA—a method for implementing ethically aligned AI systems. *J. Syst. Softw.* **20**(3), 111067 (2021). <https://doi.org/10.1016/j.jss.2021.111067>
11. Strandberg, P.E., Frasheri, M., Enoiu, E.P.: Ethical AI-powered regression test selection. In: 2021 IEEE International Conference on Artificial Intelligence Testing (AITest), Oxford, United Kingdom, pp. 83–84 (2021). <https://doi.org/10.1109/AITEST52744.2021.00025>
12. Eitel-Porter, R.: Beyond the promise: implementing ethical AI. *AI Ethics* **1**(1), 73–80 (2021). <https://doi.org/10.1007/s43681-020-00011-6>
13. Boy, G.A.: *Orchestrating Human-Centered Design*. Springer, London (2013)
14. Walden, D.D., Roedler, G.J., Forsberg, K., Hamelin, R.D., Shortell, T.M.: International council on systems engineering, Hrsg. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, 4th edn. John Wiley & Sons Inc, Hoboken, New Jersey (2015)
15. ISO: *Ergonomics of human-system interaction—Part 210: Human-centred design for interactive systems*. International Organization for Standardization, Geneva, Switzerland, ISO 9241–210:2010 (2010)
16. SAE International: *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles* (2021)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



The InSecTT Reference Architecture



Ramiro Samano-Robles

Abstract This chapter presents an overview of the Reference Architecture (RA) of the project InSecTT. The chapter makes emphasis on the impact generated by the new AI (Artificial Intelligence) algorithms and their associated building blocks (BBs) on standard architectures for IoT. The chapter focuses on the main views or perspectives of the architecture, on how AI interacts with each one of these perspectives and in the end we provide two examples of use case alignment and the analysis of the impact of AI on the individual architectures.

Acronyms

5G	Fifth Generation mobile systems
3GPP	Third Generation Partnership Project
AI	Artificial Intelligence
AIOT	Artificial Intelligence of Things
AIOTI	Alliance for Internet of Things Innovation
AFDX	Avionics Full-Duplex Switched Ethernet
ARINC	Aeronautical Radio, Incorporated
BB	Building Blocks
BGW	Bubble Gateway
BLE	Bluetooth Low Energy
BT	Bluetooth
BS	Base Station
CAL	Cloud and Application Layer
CAN	Controller Area Network
CL	Cloud Layer
CLM	Cross-Layer Management
CMMS	Computerized Maintenance Management System
CNN	Convolutional Neural Network
CoAP	Constrained Access Protocol

R. Samano-Robles (✉)
CISTER Research Centre on Real Time and Embedded Computing Systems, Polytechnic Institute of Porto, Porto, Portugal
e-mail: rsr@isep.ipp.pt

CSV	Comma Separated Values
DEWI	Dependable Embedded Wireless Infrastructure
DL	Device Layer
ECAL	Edge, Cloud and Application Layer
EMR	Electronic Medical Record
EU	External User
ETSI	European Telecommunication Standards Institute
GDPR	General Data Protection Regulation
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HLA	High Level Architecture
HTTP	Hyper Text Transfer Protocol
InSecTT	Intelligent Secure Trustatble Things
IEEE	Institute of Electrical and Electronic Engineers
IoB	Internet-of-Bubbles
IoT	Internet-of-Things
IP	Internet Protocol
ISO	International Standards Organisation
ITU	International Telecommunications Union
ITS	Intelligent Transportation System
IU	Internal User
JSON	Java Script Object Notation
LTE	Long Term Evolution
M2M	Machine to Machine
MAC	Medium Access Control
MIMO	Multiple Input Multiple Output
MQTT	Message Queuing Telemetry Transport
MTP	Media Transfer Protocol
NAT	Network Address Translationn
NB-IoT	Narrow Band IoT
NFC	Near Field Communications
NL	Network Layer
NR	New Radio
NOMA	Non-Orthogonal Multiple Access
OTA	Over-The-Air
PHY	Physical Layer
RA	Reference Architecture
REST	Representational State Transfer
RFID	Radio Frequency Identification System
RNN	Recurrent Neural Network
RSSI	Received Signal Strength Indicator
RSU	Road Side Unit
RTLS	Rela Time Locating Systems
SCOTT	Secure COnnected Trustable Things
SL	Security Layer

SLM	Security Layer Management
SNRA	Sensor Network Reference Architecture
SOAP	Simple Object Access Protocol
SSL	Secure Socket Layer
TCN	Train Communication Network
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TLV	Type length Value
TMS	Transport Management System
UDDI	Universal Description, Discovery, and Integration
USB	Universal Serial Bus
UWB	Ultra-Wide-Band
WAICs	Wireless Avionics Intra-Communications
WiFi	Wireless Fidelity
WSANs	Wireless Sensor and Actuator Networks
WSN	Wireless Sensor Networks
XLM	eXtensible Markup Language
V2V	Vehicle-to-Vehicle
V2x	Vehicle to Everything
VBGW	Virtual Bubble Gateway

1 Introduction

The proliferation of objects with embedded processing and networking capabilities is regarded as the new industrial revolution that will radically change our daily lives. Millions of distributed sensors and actuators will be controlled and automated directly by algorithms running in the cloud or edge processing servers.

The success of the Internet-of-Things (IoT) is being reflected on an increasing demand for more complex and critical applications. Examples of the new wave of applications are autonomous vehicles, automated wireless aircraft operation, remote health monitoring, virtual reality real-time surgeries, etc. New wireless technologies and in general improved IoT infrastructure are being designed particularly to support object/machine connectivity with higher data rates, higher security levels, and with real-time capabilities (e.g., 5G or fifth generation systems).

The fast market penetration of IoT has created many interoperability, design, compatibility, and regulation issues. Standardization bodies have proposed a series of reference documents, guidelines, standards and recommendations that are meant to facilitate the design of such massive critical systems.

Perhaps the first reference architecture for connected objects was the ETSI M2M [1] and ETSI oneM2M [2] architectures that addressed inter-operability and interfaces of M2M (machine-to-machine) systems. One of the first reference architectures for IoT was proposed in the European project IoT-A [3], using the concept of multiple views or perspectives of the architecture. This multi-dimensional approach matches

the multiple factor and multiple stakeholder framework for the design of modern systems. Major standardization bodies proposed their own architectures, for example the ITU architecture in [4], the IEEE architecture [5] and the ISO-standard architecture [6]. The alliance for IoT industrial innovation (AIOTI) [7] proposed a framework of interoperability between many of the existent reference architectures.

The predecessor projects of InSecTT, the project SCOTT [8] and the project DEWI [9] have also proposed high-level architectures compatible with international standards and supporting details of the objectives of each project. In the case of InSecTT [10], the major goal is to investigate the impact of AI and the new supporting technical building blocks (see Tables 1 and 2) on the different views and perspectives of standard reference architectures. This also includes to determine in which parts of the architecture AI algorithms reside, or in which parts the different functionalities associated with AI operate or can be supported. This analysis also conveys a stress and security tests for the different interfaces of the architecture. The final objective is to have an overview of how AI interacts with the IoT world, particularly using edge infrastructure and provide useful recommendations to the general user/designer based on the outcomes of the analysis of the different domains and use cases.

The organization of this chapter is as follows. Section 2 provides an overview of existing approaches investigating the impact of AI on IoT architectures. Section 3 provides an overview of the InSecTT architecture. Section 4 presents the entity model of the architecture, while the domain model is described in Sect. 6. Section 7 presents the functionality model, while Sect. 8 presents the information and communication models of the architecture. Section 9 presents the new AI perspectives of the architecture and their impact on the rest of the views. Finally, Sect. 10 presents examples of the alignment of use cases to the InSecTT reference architecture.

2 AI in IoT Architectures

The last decade has witnessed an exponential increase in applications of AI for a variety of aspects of IoT applications. These aspects range from the lower layer transmission improvements, to upper layer applications including intelligent services, consumer preference prediction, etc. However, the impact on the IoT architectures is rarely addressed consistently in the literature. One example is the work in [11], where the authors study the use of specific AI functionalities across different layers and entities of an IoT architecture enabled with block-chain technology. The use of AI in Edge computing architectures is presented in [12]. This work focuses more on the entity and logical model views of Edge processing architecture and the impact of AI.

Other works offer a semantic decomposition of AI algorithms and their specific processes in IoT architectures or applications. The authors in [13] propose the use of specific sub-functionalities generic to different AI algorithms such as feature extraction, learning, knowledge storage, decision making and automation control.

This type of decomposition seems the most attractive to include specific AI processes in future AIoT (Artificial Intelligence of Things) architectures.

The work in InSecTT was to propose a relative advance in the state of the art on how AI tools have an impact on IoT reference architectures. More specifically the work was initially intended to decide whether the AI impact is high enough to include specific sublayers or views or another types of tools in the official InSecTT reference architecture. The next step was to modify the official architecture and align the existing use cases and the InSecTT universe of AI algorithms. Some of our conclusions are expressed in this chapter.

3 Overview of the InSecTT Architecture

Let us now define the InSecTT RA as *the set of guidelines for infrastructure organization of IoT (Internet-of-Things) use cases targeting industrial-grade connectivity, security, dependability, interoperability and trustworthiness with the help of artificial intelligence (AI)*. It provides the high-level view of (sub-)building blocks, interfaces, vulnerabilities, security solutions, protocols, and in general the detailed information flow of InSecTT use cases in different industrial domains (aeronautics, automotive, railway, building, healthcare, maritime, etc). This provides us with a tool to analyse reusability, standardization, certification and verification issues across domains.

The InSecTT RA hosts a set of best practices collected across three EU projects: DEWI [9], SCOTT [8] and InSecTT [14]. The DEWI RA focused on dependability, using IoT protocols as a method to provide interoperability. The concept of DEWI Bubble was used as the encapsulation of legacy infrastructure. The DEWI RA was built on top of the ISO SNRA (sensor network reference architecture) [15]. The extension considered dependability improvement features. The SCOTT project saw the extension towards a full IoT architecture with improvements on high level aspects such as Edge/fog processing, security, privacy, safety and trustworthiness. This was achieved by combining multiple standard architectures. The InSecTT RA reuses the DEWI/SCOTT frameworks and the Bubble concept to investigate the impact of AI on IoT architectures. The material used in this chapter is an extension of the work previously published in [14, 16].

The core of the InSecTT solution is the concept of Bubble. An InSecTT Bubble is a logical entity formed by a group of nodes, gateways, internal users and existing (legacy or new) industrial infrastructure. The main property of a Bubble is that it provides a single point of access (the Bubble Gateway) to the information of the entities in the intra-Bubble space. The InSecTT Bubble is therefore useful to encapsulate multiple industrial protocol standards into a consolidated IoT technology format improving and enforcing inter-operability, dependability and cross-domain development. The Bubble recommendations target the dependable integration of wireless/wireline industrial infrastructure using a three-layered intra-Bubble hierarchy that facilitates intra-domain adaptation and protocol translation, and a new *trustworthiness-by-design* philosophy. InSecTT foresees a landscape of communi-



Fig. 1 Evolution of the Bubble

cating intelligent Bubbles implemented in different industrial use cases that can be called the Internet-of-Bubbles (IoB). Each Bubble can decide, if convenient, to allow transparent access to the nodes inside the Bubble or provide only consolidated, aggregated or processed information.

3.1 Evolution of the Bubble

In the DEWI project, the Bubble was introduced for the first time as encapsulation of industrial infrastructure which was useful to improve dependability and interoperability. The Bubble in DEWI was also created with a mechanism of technology cross-domain utilization, using a proprietary protocol for dynamic sharing of software or middle-ware components between Bubbles. By contrast, in SCOTT the Bubble evolved towards a type of advanced Edge infrastructure with security as added value. The extension to SCOTT included the use of multiple modifications to interfaces and functionalities to enable the Bubble as a secure encapsulation of industrial communications and secure exchange of technical building blocks. The evolution from DEWI to SCOTT also included the use of security and trustworthiness metrics for different layers of the architecture. The Bubble was therefore modified to support communications between Bubbles based on trustworthiness metrics, which leads directly to the use of adaptive security enhancement solutions. The SCOTT architecture was created based on the combination of multiple standards for IoT architecture, preserving the Bubble for encapsulation, dependability and security of industrial use cases. In InSecTT, the Bubble evolved to cope with a more dynamic interface scenario inside the Bubble and with the ability to host AI algorithms in different levels of the architecture or in different entities. The Bubble became “intelligent”. This evolution is reflected in Fig. 1.

Table 1 InSecTT sub-building blocks

SBBs	Name
2.1	<i>AI for applications</i>
2.1.A	AI for audio-visual data
2.1.B	AI for multidimensional data
2.1.C	AI for communication(s) data
2.2	<i>AI for wireless</i>
2.2.A	Interference and signal processing
2.2.B	Localization, DOA , beamforming and PHY-security
2.2.C	Link estimation, prediction, routing and con. management
2.2.D	Anomaly detection
2.3	<i>AI for wireless</i>
2.3.A	Algorithms & Architectures for AI on device/edge
2.3.B	Design time distribution of AI
2.3.C	Dynamic Distribution and Management of AI
2.3.D	Privacy and security of distributed AI
2.4	<i>AI V&V</i>
2.4.A	V&V of AI-based systems
2.4.B	AI for audio-visual data
2.4.C	AI-based evaluation and testing methods
2.5	<i>AI for wireless</i>
3.1	<i>Security</i>
3.1.A	Access control and authentication Infrastructure
3.1.B	Intrusion detection systems
3.1.C	IoT privacy & security mechanisms
3.1.D	Secure IoT applications
3.1.E	Security guidelines
3.1.F	Tools and simulators

3.2 Modern Reference Architectures

The InSecTT RA follows the multiple-perspective approach used by modern IoT systems matching the needs of multiple stakeholders and multi-level quality of service end user applications. Figure 2 shows the perspectives of the InSecTT RA in contrast with the views of the ISO IoT reference architecture in [6]. In blue are shown those views that are part of the InSecTT RA and that were extended from the ISO model. In orange are those that belong to the ISO model but were not entirely developed in

Table 2 InSecTT sub-building blocks

SBBs	Name
3.2	<i>Reliability</i>
3.2.A	Gateway
3.2.B	Devices functionalities/management
3.2.C	Communication layers improvement
3.2.D	Secure IoT applications
3.2.E	IoT network
3.2.F	Wireless link monitoring
3.3	<i>Reliability</i>
3.3.A	AI-based anomaly detection & response and/or real-time security mechanisms in a delimited environment
3.3.B	Quality of service monitoring & response
3.3.C	Communication layers improvement
3.3.D	Secure IoT applications
3.3.E	IoT network
3.3.F	Wireless link monitoring
3.4	<i>Real time</i>
3.4.A	Real-time Interface Management for Cellular Connections
3.4.B	Devices functionalities/management
3.4.C	Communication layers improvement
3.4.D	Secure IoT applications
3.4.E	IoT network
3.4.F	Wireless link monitoring
3.5	<i>Real time</i>
3.5.A	Co-Simulation platform for cybersecurity testing environment
3.5.B	Automotive testbed interface
3.5.C	Security testing core system
3.5.D	Security testing core system
3.5.E	Channel and physical layer simulation
3.5.F	V2x connected cars (Network) digital twin

this project. In yellow the views that are not ISO and were not developed in detail here, but they are borrowed from other projects. In green those in the ISO model but not included in InSecTT. Finally, in grey the additional views included in this project that are not ISO. We will describe the main views of the architecture and the impact of artificial intelligence on each perspective. We will describe the main views of the architecture and the impact of artificial intelligence on each perspective. Additionally we describe our own AI perspective of the reference architecture. It is worth pointing

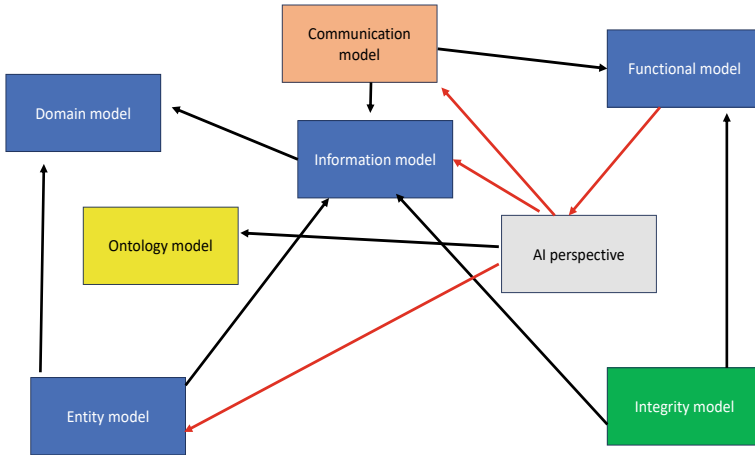


Fig. 2 Architecture perspectives

out that the different perspectives of the RA are not completely independent and we usually find overlapping aspects between them. In several alignment exercises it is sometimes necessary to use a hybrid view representation, which combines relevant aspects of two or more of the perspectives portrayed in this framework.

4 Entity Model

The main perspective of the InSecTT RA is the layered physical entity model portrayed in Fig. 3. In addition to the definition of each entity in an IoT network, the InSecTT RA provides a layered hierarchy which has been specifically designed for the integration of new and legacy wireless and wireline industrial infrastructure in a dependable and secure manner. The proposed three-layers reflect the interaction between the wireless or Level 0 (L0) world, with the existing and potentially critical wireline industrial infrastructure (called Level 1 or L1), and Level 2 (L2) that acts as the encapsulation of the previous two layers. This encapsulation is in the form of a Bubble using a physical or virtual InSecTT Bubble Gateway (BGW) providing external services that can be invoked by other applications or other Bubbles based on multiple trustworthiness metrics. Note that AI algorithms or functionalities of these AI algorithms can be distributed in different elements and levels of this layered model, and each implementation has different implications in terms of complexity, security, communication interfaces and functionalities.

Each layer of the architecture can also host different functionalities related to the new wave of AI algorithms. The BGW is therefore the main entity controlling access to the information of the internal nodes of the Bubble. Other GW entities can be defined inside the Bubble to deal with decentralized processing (AI) and depend-

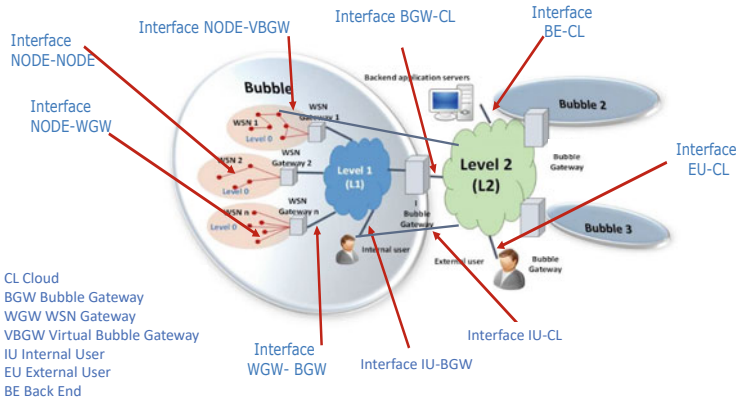


Fig. 3 Entity model

ability control between L0 and L1. The three-layer architecture allows designers to distribute complexity and AI functionalities in different layers and different types of gateways, providing encapsulation of legacy technology in modern IoT protocols for interoperability and secure information transport. Level 0 (L0) is the wireless technology used inside the Bubble for one or more WSNs. Level 1 (L1) is the infrastructure inside the InSecTT Bubble to connect several WSNs to the corresponding BGW. This can be for example, the internal bus of a vehicle or the proprietary network of a building. Level 2 (L2) is the infrastructure providing a common external access to the Bubble (usually based on a request-response paradigm).

The main physical entity in the InSecTT architecture is called Bubble. A Bubble will be able to contain one or more wireless sensor or node networks. Each WSN or node network will be managed by an L0 Gateway (also called WSN gateway or WGW). Therefore, the Bubble Gateway will be in charge of managing all the L0 Gateways inside the Bubble, while also providing external/internal secure access to the information of Nodes (sensors and actuators) of each WSN or node network. This internal/external access to the Bubble information is encapsulated in a set of Bubble Services that can be invoked by internal and/or external users (AI algorithms external to the Bubble can invoke these services or bubbles can also host AI services). This also means that all Bubble operation and management of high-level services will be hosted by the Bubble Gateway. In principle, the bubble services can also include the functionalities related to AI, such as learning, or sharing data sets (federated learning). This infrastructure arrangement naturally leads to the definition of three levels or layers of communication (one of them being optional):

Level 0 (L0) is the communication technology/architecture inside a specific wireless sensor network (intra-WSN). Each WSN can have a different Level 0 technology. Several WSNs can be hosted by one Bubble Gateway. L0 is in general the most unreliable of the three levels of the architecture. Therefore, multiple AI algorithms can be

destined to overcome the issues found in L0, such fading, shadowing, interference, eavesdropping, etc. Particular changes have been implemented in the final version of the architecture to deal with long range L0 technologies such as 5G.

Level 1 (L1) is the communication technology/architecture inside the InSecTT Bubble to connect several WSNs to the corresponding Bubble Gateway. L1 technology is **optional**, mainly because some scenarios will deploy only one WSN per Bubble. In such cases, the Bubble Gateway will control all the nodes directly. L1 technology depends on the use-case (UC). Some of the UCs require L1 to be a real-time technology. Middleware solutions for L1 usually deploy publish-subscribe approaches to reduce operations and latency. Several AI implementations at the edge can take advantage of L1 technology being real time to reduce latency and improve reliability. In particular, resource allocation AI algorithms to improve L0 performance can reside in this L1 hierarchy. L1 technology is largely vulnerable to new attacks that come from the new wireless medium. AI algorithms can help in protect L1 infrastructure which in general is more critical than the L0 technology.

Level 2 (L2) is the communication technology/architecture outside the Bubble (extra-Bubble). It provides a common(standard) external access to the Bubbles. This technology should be a standard for all industrial domains, so clients (humans and machines) can gain access to any kind of Bubble to use the available Bubble Services. Middleware approaches usually deploy a request-response approach consistent with an Internet cloud technology with non-critical traffic support. External AI algorithms can run in this level of the architecture, but usually in a non-critical fashion. In general they can be used to update data sets using learning algorithms resident in the cloud or back end servers. This distributed learning scheme needs security and privacy.

The Bubble helps designers to enforce different trustworthiness metrics inside the Bubble. By explicitly isolating critical infrastructure and providing specific mechanisms (secured) that external entities are allowed to access or request, security is improved and therefore external attacks can be controlled or reduced. In addition, the concept of the Bubble has been found compatible with modern technologies such as block-chain, Edge/fog computing, and now AI. The Bubble is well suited for distributed AI in the three levels of the architecture with different levels of complexity that have been showcased in different industrial domains. The virtual Bubble GW is adapted to include direct cloud links or hybrid combinations of short range with long range direct cloud links inside the Bubble. This means that the BGW can be completely virtualized in the cloud or edge infrastructure of a service provider. This is also compatible with futuristic implementations of 5G/6G systems with network slicing. The layered approach of the InSecTT Bubble is shown in Fig. 3 with the different types of HW interfaces between entities.

The InSecTT RA hosts a set of entities with different roles and functionalities. The main entities and the hardware interfaces enabled between them are shown in Fig. 3. The main entities are the Bubble nodes, the different types of Bubble Gateways, the different types of users of Bubble services and the external entities to the Bubble. The Bubble GW has a dominant role in being the enabler of the Bubble services and controls all access to the information inside the Bubble. The Bubble will also host the majority of AI algorithms. We highlight the possibility of the virtual Bubble

GW (VBGW) to deal with those use cases where direct cloud links can be used by nodes inside the Bubble. The virtual and physical BGW can coexist, but always they should be integrated to mimic a single entity for security reasons. This also leads to the concept of hybrid user which is particularly suited for modern terminals with multiple radio interfaces and flexible mobility that can roam in and out the Bubble providing different levels of connectivity between nodes and external entities or with the virtual Bubble GW. We highlight the use of multiple gateways per level of the hierarchy to preserve the quality of service, delay, security, and offer encapsulation of underlying industrial technologies. Unlike other standard architectures, the InSecTT RA provides with specific procedures to support this detailed industrial connectivity and dependability issues between wireless and internal industrial wireline protocols. This is particularly useful, for example, in automotive use cases where wireless sensor readings are relayed to the internal network of the car, or also on board aircraft where sensor nodes using the new wireless avionics technologies relay information to the internal critical aeronautics network. The node and entities of InSecTT are allowed to use multiple interfaces creating new challenges in routing, security, authentication and privacy that can be addressed by the InSecTT building blocks. The InSecTT building blocks are displayed in Tables 1 and 2. The RA also has specific procedures for service and object virtualization which are important in applications such as digital twins and for security enhanced remote control.

The following elements constitute the intra- and extra-Bubble space of the RA:

Bubble Node (BN): Any tag, sensor, and actuator (or combination of some of them as one single node entity) working under the Bubble framework. Bubble Nodes implement several of the functions of the functionality model described in subsequent sections of this document. The nodes may have different radio technologies for communication and may not necessarily lie within the wireless range of each other. Multiple interfaces simultaneously are allowed in a single device. Some nodes could also host AI algorithms, depending on the complexity constraints.

Bubble Gateway (BGW): The entity that acts as the main logical interface of a Bubble to communicate with other Bubbles, users (internal and external), and with other external (extra-Bubble) entities. It also provides protocol translation and management functionalities for all the InSecTT nodes and all the WSNs inside the InSecTT Bubble. For redundancy purposes, alternative gateways can be deployed, but there must be always only one logical gateway acting as interface of the InSecTT Bubble. The majority of AI algorithms or associated functionalities are expected to be hosted by one of the BGWs described here.

EDGE Bubble Gateway (EBGW): An InSecTT Bubble Gateway with explicit Edge processing capabilities. An Edge BGW is particularly suited for low latency and security enforced near the edge of the network. In InSecTT, the Edge Bubble Gateway has a particular importance for hosting most of the AI processing capabilities.

Virtual Bubble Gateway (VBGW). A cloud-mirror implementation of an InSecTT Bubble Gateway in charge of the remote control of external connections invoking specific out of band connections to the Bubble Gateway services. The virtual Bubble Gateway can act in collaboration with a physical Bubble Gateway, or as standalone, thus defining a virtual Bubble completely managed by the Virtual

Gateway. The need of a Virtual Gateway comes from the new complex use cases with multiple users or nodes using a variety of short-range or direct cloud technologies to interact with the Bubble infrastructure. An advantage of the virtual Bubble implementation is the invoking of AI algorithms centralized in the cloud or in Edge processors. This enables low latency end to end for AI services.

Relay Gateway (RGW): A device that allows direct Bubble-to-Bubble (B2B) communication without the need of specific infrastructure. It also relays the functionalities of a DEWI Bubble Gateway. The relay gateway allows the concept of the Bubble to acquire full strength. A Bubble makes the information available to the external world only by accessing the boundary of the Bubble, through the designated relay or main Bubble gateways. A pending issue is the use of relay gateways to invoke AI services of other bubbles. This is a security issue that must be tackled in the future.

WSN Gateway (WGW): A device in charge of the management and protocol translation of an intra-Bubble WSN, e.g., a ZigBee gateway, or a Bluetooth master device. The WSN Gateway can be a legacy device. The WSN Gateway communicates with the Bubble Gateway, which is in charge of the management of all the WSN Gateways inside the same Bubble. In some cases, these devices host some of the AI algorithms, particularly those focused on the improvement of the physical layer.

Users (internal and external): The entities that can access and use the set of Bubble services provided by the Bubble Gateways or Relay Gateways. Internal Users are meant to be inside the Bubble accessing the Bubble Gateway via the intra-Bubble communication infrastructure. However, in InSecTT the internal users can be provided also with a direct connection to the cloud, usually a virtual mirror cloud application of the Bubble Gateway. The external users access the Bubble services through the extra-Bubble communication framework (cloud). This access is mainly via Internet and over open standard middleware application program interfaces. With the use of AI, new issues arise related to the privacy and confidentiality of data used to train different algorithms. Some of the users of the bubble services could access data used by learning algorithms in different parts of the architecture. We highlight the need to protect data confidentiality even from the internal users of the Bubble.

Service providers and back-end servers: An entity or set of entities that provide services to third parties based on the aggregation of multiple Bubbles. In distributed AI, we can have AI services or AI functionalities invoked as services in different bubbles. Therefore, the InSecTT Bubble can also become an intelligence service provider

5 Layered Model

5.1 Level 0

Level 0 (L0) is the technology used inside each WSN of a InSecTT Bubble. L0 technology can be used to interconnect the InSecTT Nodes with two different entities:

1. the WGWs (in case the InSecTT Bubble contains more than one WSN) or
2. the InSecTT BGW (in case a InSecTT Bubble only hosts one WSN, i.e. L1 does not exist).
3. directly to the cloud, in case the nodes have multiple interfaces and one of them provide direct 5G access to cloud infrastructure.

Level 0 technological choice is independent for each use-case. Level 0 is usually the wireless technology used to interconnect, manage and retrieve the information from/to sensors and actuators (InSecTT Nodes). The challenge in the design of the HLA and its appropriate infrastructure is to host such a diverse and heterogeneous landscape of technologies communicating with each other. The project DEWI focused on the dependability issues of L0 infrastructure. SCOTT project dealt with security and trustworthiness (trust) issues of IoT using a diverse set of L0 technologies. InSecTT project now builds upon these two previous projects dealing with the improvement of both dependability, trust, and security/safety using AI technology at the edge of the network. It is worth pointing out the generational change experienced from the beginning of the DEWI project and now in the course of the InSecTT project regarding wireless technology. 5G is now a reality in commercial implementations, which allows direct cloud connections from virtually anywhere inside the cellular provider's coverage area. In addition, vehicular technologies and local area networks have also been evolving to become more reliable with higher capacity and lower latency. The technology MIMO (Multiple-Input Multiple-Output) is now not only a fact but a necessity in the new standards. New challenges are also emerging such as 6G and the concept of THz communications. It is expected that the algorithms and building blocks proposed by InSecTT are reusable for upcoming L0 technologies.

We recall here some of the important technologies and issues found in L0 and the potential of AI to counteract such issues. Wireless technologies experience different issues in different frequency bands. In general, higher frequencies achieve higher data rates at the expense of increasing path loss, which limits coverage thus increasing infrastructure requirements to provide urban footprint. Different transmission technologies also possess different qualities in face of impairments. While cellular communications are currently dominated by OFDM-based transmission technology, including the 5G new standards, a multitude of technologies are currently coexisting across different standards, including spread spectrum frequency hopping, direct sequence CDMA, ultra-wide band, etc. Each one of these choices has different virtues in the face of issues such as fading, shadowing, path loss, non-line of sight, etc. CDMA-based are more resistant to fading due to their diversity combining gains, but OFDM are attractive to deal with dense multipath interference. UWB technology has been found useful in localization applications in dense multipath situations such as in metallic environments. Joint sensing and transmission technologies also make use of the properties of wireless transmission standards, particularly using WiFi and/or the new MIMO features of 5G. The issue of jamming is closely correlated to the ability improvement brought by MIMO or other interference cancellation schemes. In general, the use of artificial intelligence for the improvement of

the reception/transmission of wireless signal was tested in the project with different degrees of success. In the case of MIMO, an AI-MIMO transceiver was successfully designed and tested by simulation. The advantages seem very clear, surpassing some well know signal processing tools, but several disadvantages were also found. In wireless networks, the major disadvantage seems to be the rapid statistical changes experienced in wireless networks, and the need to select rapidly different data sets, or conversely to train models using a huge set of wireless measurements. These are open implementation issues that can be solved in the future.

5.2 Level 1

Level 1 (L1) is the technology used to interconnect all the intra-Bubble WSNs to their corresponding InSecTT BGW (in case the InSecTT Bubble hosts more than one WSN).L1 is an optional technology, but in some use cases it is an integral part of the legacy infrastructure that must be protected from issues due to the addition of new features . There are two main reasons why L1 is an optional feature in the HLA:

- There are some use-cases in which the InSecTT Bubble hosts a single intra-Bubble WSN. In these cases, the InSecTT BGW and the WGWs converge in one single entity: L1 disappears and only L0 and L2 are implemented; and
- Different use cases have different requirements to interconnect the WSNs.

There are several advantages of using a different L1 technology for different use cases. Some of them are listed below:

- It allows network designers to meet domain-specific or use-case-specific design criteria. This also has an impact on AI implementation by controlling the environment and coverage footprint, which in turn controls the data sets and reduces the non-stationary effects that can reduce the effectiveness of AI for wireless.
- It also improves scalability in the HLA, mainly because several WSNs can be hosted by one single InSecTT BGW. Thus, the number of InSecTT Nodes hosted per InSecTT BGW increases according to the supported WSNs hosted by L1 technology. AI solutions can further increase the scalability provided by L1.
- It provides a useful organization of resources in dense object deployments. It also allows centralization of intelligent services, and the share of learning weights or data sets among different networks inside the bubble.
- It allows for an efficient resource allocation by centralizing all the WSN management in one single location (logical): the InSecTT BGW. This means, for example, that adjacent WSANs can be allocated in different channels by the InSecTT BGW, thereby reducing interference inside the same InSecTT Bubble (intra-Bubble interference). AI algorithms for channel prediction can be used for resource allocation of adjacent WSNs inside the same Bubble
- The existence of L1 technology facilitates the integration of wireless InSecTT Bubble solutions with the existing wired technologies of each domain, which is a

unique aspect of InSecTT. This aspect also improves the technology re-usability features of the HLA, mainly because some wireline technologies share common aspects across different industrial domains,

- L1 can provide wired/wireless infrastructure for internal users to obtain access to InSecTT Bubble services,
- L1 allows network designers to create a private internal network where high security and quality of service requirements can be enforced, and
- L1 can naturally hosts AI algorithms to be reused across different L0 technologies.

The decision of using L1 also conveys some additional efforts summarized as follows:

- L1 needs an additional (Radio Resource Management) layer to organize the different WSNs inside the Bubble.
- L1 needs to design gateways and scheduling policies that match the quality of service and dependability requirements between L0 and L1. Mainly in case of critical designs.
- The InSecTT Bubble becomes a complex set of heterogeneous WSNs and wireline transmission protocols that need special cross-layer optimization algorithms. Some of these can be based on AI.

Level 1 technology must also comply with several of the requirements and attributes of InSecTT for intra-Bubble communications. Some of these requirements are:

- L1 must provide reliable, transparent and secure access to the information of the InSecTT Nodes inside each Bubble.
- L1 must preserve the dependable, self-configurable and secure data transfer of the InSecTT Bubble.
- L1 must provide the InSecTT Bubble with a framework for the management, resource allocation, configuration, troubleshooting and maintenance of the intra-Bubble WSNs.
- L1 must provide conflict resolution or deterministic allocation of the data streams and traffic of all WSNs. Prediction algorithms based on AI, as well as fault detection can be implemented to improve performance.
- L1 provides an interface for internal users of the InSecTT Bubble to access the InSecTT Bubble Services.

Level 1 in InSecTT HLA is closely related to the existing infrastructure (wired) of each industrial use-case. Since L1 must preserve the dependability requirements inside the Bubble, middleware tools with message-oriented and application-oriented approaches are recommended. Service-oriented architecture for L1 provides the HLA with the flexibility to organize resources across different WSNs and model them as available services inside L1. Publish-subscribe middleware technologies fit perfectly the requirements of L1 high quality of service.

Regarding AI algorithms, L1 technology based on a real time technology offers interesting options to implement improved resource allocation based on AI between WSNs and use deterministic deadline calculation for services of different WSNs with controlled interference. L1 is also the side of the network enabled by EDGE Bubble

Gateway implementations of AI algorithms. This means that L1 will be probably the recipient of most of the traffic generated by the new resource-hungry AI processing requirements running at the Edge of the network. The accurate estimation of resources needed by this type of AI is not yet clear, thus being an open issue on the evaluation of the impact of AI in L1.

Since L1 is related to the existing infrastructure in different use cases, there has been not much change with respect to the technologies described in SCOTT and DEWI. The domains of automotive, railway, and aeronautics continue with their bus standards but now they have included stronger cybersecurity recommendations focusing on the new types of threats. In the domain of building, healthcare and maritime, the use of conventional Ethernet-base infrastructure provides a higher degree of interoperability between building blocks.

5.3 *Level 2*

Level 2 (L2) is the layer in charge of the communication between InSecTT Bubbles and external user(s) or agent(s). The external agents can be other InSecTT Bubbles. L2 is thus responsible for the interoperability and cross-domain application development. Level 2 is the way an InSecTT Bubble exposes its information and services to the outer world, including external AI in the cloud. There are several features unique of Level 2 in comparison with the other two Levels of the HLA:

- L2 uses Internet-based (IoT) protocols to enable external users with access to the set of Bubble services from anywhere in L2.
- L2 is also responsible to provide federated access to a set of different InSecTT Bubbles and to data sets or information learned from the different Bubbles.
- L2 must support interoperability, cross-domain application development and technology (AI) re-usability.
- L2 must enforce security and protection to the data and operation of the InSecTT Bubble against external malicious attacks (AI-based)
- L2 must use international standards to enable interoperability with other external systems.

In order to achieve these goals, L2 has adopted a service-oriented architecture (SOA) with a middleware paradigm based on request-response. This paradigm is inherently non-real time. Therefore, L2 is envisioned as a layer where external users request access to a summarized version or non-real time information generated by the InSecTT Bubbles. The InSecTT functional model includes specific L2 services. The previous projects DEWI and SCOTT have not proposed a unique L2 communication technology. Instead, the approach was to provide generic guidelines to achieve the goals of Bubble-to-Bubble communication and interoperability goals according to the INTEROP standard model and their definition of interoperability. In this project there has been a narrowing of the protocols used for L2 communication. Since the

focus is on AI, the variety of protocols used declined with respect to the previous two projects. We saw an increase of the influence of the MQTT protocol versus other L2 technologies such as HTTP, COAP, etc. MQTT showed great flexibility for the variety of use cases and technological solutions based on AI.

5.4 Hardware Interfaces

The following interfaces between entities are allowed in the InSecTT architecture:

Node-Node. This can be the same as the L0 Node to WGW link. However, in some situations this technology can be implemented using another technology, using multiple interface devices. The impact of AI is not expected to be huge on this interface, except in some application such as platooning, where V2V technology is reliable enough to achieve ultra-low latency and real time communications.

Node-WGW. The main L0 wireless technology to provide pervasive coverage from the WGW to the nodes inside the Bubble. It can also be used by AI algorithms dealing with the physical layer.

WGW-BGW. The L1 intra-Bubble infrastructure communication that connects all WSNs inside the Bubble to the Bubble Gateway. This interface is used to transmit messages containing data from the nodes and sensor network management messages to/from the WGW. The protocols defined in this interface depend on the application requirements and on the WSN architecture chosen to build up the network. The data information transmission from the nodes to the WGW might be scheduled periodically, triggered by events, or on request by the WGW. The scheduling might also occur in case AI algorithms need to update the training information.

Node-BGW. The L0 technology directly between the Nodes and the BGW, in case there is no L1 technology.

Node-EBGW. The L0 technology directly between the Nodes and the BGW, in case there is no L1 technology

Node-VBGW. The L0 technology directly between the Nodes and the BGW, in case there is no L1 technology and no Physical BGW. This interface can be used to connect directly nodes to AI algorithms in the cloud.

IU-BGW. The interface between the internal user and L1 technology of the BGW

IU-VBGW. The interface between the internal user and the Virtual Bubble GW

EU-CL. Interface between an external user and the cloud servers. This is the L2 technology.

BGW-CL. Interface between the Bubble Gateway and the cloud back-end servers.

6 Domain Model

InSecTT has adopted a modified domain model of the IoT-A reference architecture European project [3]. The domain model is displayed in Fig. 4, showing the main physical entities to be represented in the cyberspace: tags, actuators, nodes and the

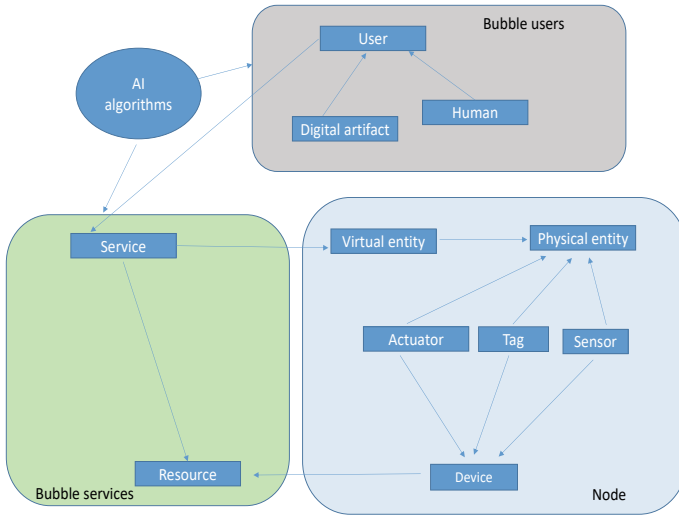


Fig. 4 Domain model

InSecTT Bubble, as well as their virtual representations and the link to the external users through a service-oriented implementation. It provides the virtualisation of objects to be represented in the cyberspace. It also defines the relationships between services, external entities, devices and virtual representations. This model has been modified in the project SCOTT to account for different trustworthiness, security and privacy metrics. The conclusion of our investigation in the project InSecTT is that the AI dimension has impact on this perspective, particularly if we use federated learning or another type of distributed learning processes. In addition, explainable and trusted AI distributed algorithms need a domain model approach to enable the addressing, organization, and discovery of distributed AI and learning resources in the cyberspace. In InSecTT, we envision a future cyberspace with multiple resources and components of AI algorithms with different levels of trust and different properties that need a mechanism to be organized, addressed, listed, discovered, and managed. Therefore, the domain model will be essential to characterize the new elements of the new and potentially distributed intelligence dimension.

This domain model allows objects to be mapped directly into a virtual entity that can be reached via Internet protocols and tools. This is the main concept behind the IoT: physical objects accessible by the cyberspace. However, the InSecTT project provides another level of abstraction: the InSecTT Bubble. The bubble can provide transparent access to the objects inside the Bubble or it can decide to provide limited access. Therefore, the entity that is directly mapped into the virtual space will be the Bubble rather than individual objects. It depends on the particular implementation if the Bubble provides transparent access to objects or entities inside the Bubble. This is an aspect that will be defined in each use case or implementation of the reference architecture. It is not yet clear for the future if the Bubble will be able to provide

intelligent services or these will be located in other edge or cloud entities. In this architecture we leave both options as feasible to cover future decisions.

7 Functionality Model

The functionality model in Fig. 5 is a combination of the ISO IoT/SNRA [6, 15], the ITU [4], IEEE [5] and the AIOTI functional models [7]. The layers will be implemented by the physical entities of the reference architecture. Each of the layers of the functionality model can communicate with other layers using software interfaces. Each SW interface is potentially a standard data format or protocol and it can be subject to vulnerabilities. The InSecTT RA provides the option to have security mechanisms for each layer, in addition to the conventional security network layer included in the service and virtualization layer. It also includes a security management vertical layer that coordinates all security/trustworthiness solutions across different layers. The four horizontal layers are device (DL), network (NL), service (SL), and the cloud and application layers (CAL). The DL includes functions near the hardware, such as energy harvesting, sensor-related and basic MAC-PHY functionalities. The NL maps information into the cyberspace of L2 level. The SL encapsulates the lower layers presenting them as services, including virtualization services. This layer includes a security layer that runs on top of the conventional networking OSI layer. Finally, the CAL layer invokes the services of the SL as applications.

The InSecTT functionality model includes specific service virtualization features and cross-layer management. In addition, it includes a detailed functional model decomposition with multiple trustworthiness metrics evaluation models to investigate how these different metrics evolve across layers and entities of the RA. This has led to L2 adaptation based on trustworthiness metrics (indicators) and online certificates

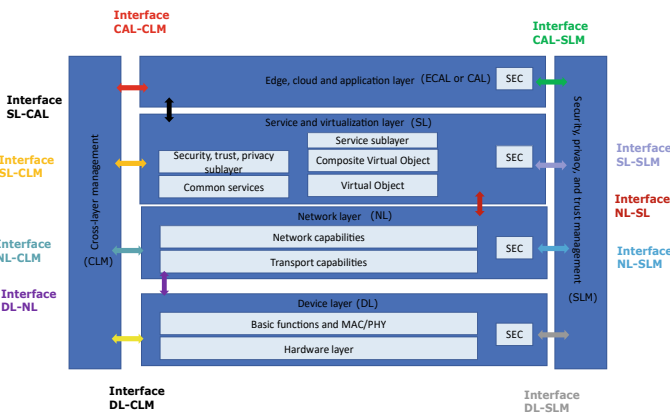


Fig. 5 Functionality Model

between Bubbles. Our vision is that communicating Bubbles in the cyberspace will be able to exchange trustworthiness metrics, indicators or information with online certification entities or anchors and this exchanged information can be used to adapt security, communication, semantics and other features in the interaction between Bubbles and other entities. InSecTT aims to use AI to improve several of these inter-Bubble interactions. More details of the layers of the functional view of the reference architecture are the following:

7.1 SW Interfaces

These interfaces between adjacent layers are the direct extension of the interfaces proposed by the ISO SNRA standard in [6]. These interfaces were adopted and modified in the SCOTT project to address the different issues of security, privacy and trustworthiness of IoT.

Interface DL-NL: This interface between the device layer (DL) and the network layer (NL) is usually part of the protocol stack definition. The network layer involves the basic functionalities of the communication interface (MAC-PHY) to provide network connectivity via a higher-level protocol, format, and frame definition. The network layer encompasses the network and transport layers of the OSI model. It also covers network management, troubleshooting, sensor information processing, fusion, etc. Security schemes are also important to avoid misuse of the functions invoked by the network layer, particularly for the operating system of one of the physical entities of the reference model. This is a relevant interface for AI algorithms that request information from the DL and transport it to another entity.

Interface NL-SL: This interface is used for the service layer to invoke the functionalities of the lower layers, encapsulated by the network layer. The service layer of the IoT has many different variations, particularly due to the diversity of use cases and applications. This interface is particularly prone to attacks or other vulnerabilities. Code injection, man in the middle attack and denial of service attacks can exploit this interface. AI algorithms for anomaly detection can be used in this interface. AI algorithms running at the edge might exploit this interface to obtain information.

Interface SL-CAL: Interface that allows the upper layer applications to access the services enabled by the BGW and all the encapsulated technology building blocks or functionalities. The virtualization of AI functionalities is directly related to topics such as Federated learning.

Interface DL-CLM: This interface allows cross-layer algorithms to collect information from the device layer and optimize system performance. The information from the device layer varies according to the application: channel state, energy, power, user ID, etc. AI algorithms for improvement of wireless networks are expected to use heavily this interface, particularly in scenarios with non-stationary statistics.

Interface DL-SLM: This interface focuses on the multi-layer security with the device layer. Examples of this interface allow MAC-PHY algorithms to identify jammers or directions of eavesdroppers. Node identification using direction of arrival

algorithms, or statistical signal processing are also possible. Redundancy of source and channel coding can also be used. AI for anomaly detection will exploit this interface.

Interface NL-CLM: In this interface, the network layer provides information to cross-layer optimization algorithms. Routes, addresses, traffic state, quality of service, etc. are some of the metrics and information that can be requested through this interface. AI algorithms for link/route selection are expected in this interface.

Interface NL-SLM: The network layer interacts with the security layer management via a set of specific protocols. Tunnelling, virtual links, security layers, etc. are examples of specific implementations of this interface.

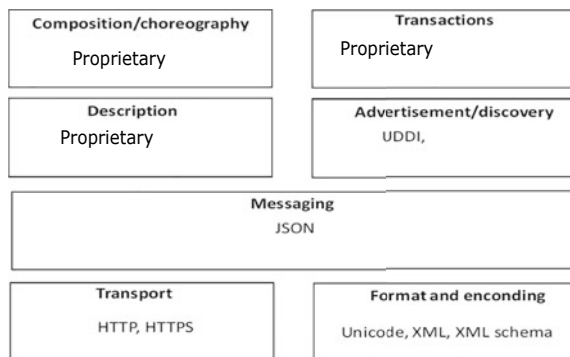
Interface ECAL-CLM: This is an innovative interface analysed in InSecTT project. It defines the interactions between cloud computing platforms and cross-layer management. It is assumed that the cross-layer management and optimisation take place locally in the Bubble, whereas cloud computing takes place in the external L2 world. Therefore, this interface has close connection with modern approaches such as Edge or Fog computing.

Interface ECAL-SLM: The interface between cloud and security layer management.

8 Information Model

The information model is crucial for the communication between different Bubbles. An information model deals on how the information is transported, encoded, formatted, encapsulated and how the resources are organized and managed, including aspects such as discovering, addressing, enumeration and scheduling. The information model also includes high level aspects such as semantics representation, transactions and business modelling. Figure 6 presents an example of an information model for the use case of wireless avionics. Typical technologies for format and encoding are the family of extensible markup languages (XML), TLV (Type-Length-Value)

Fig. 6 Information model



or CBOR (Concise Binary Digital Object). Advanced data exchange models can combine messaging and format encoding, for example JSON (Java Script Object Notation), the OWL (Ontology Web Language), Sensor ML (Mark up language), etc. For transport, solutions can be broadly divided in RESTful, which are request response , such as HTTP, MQTT, SOAP, and CoAP,etc. and publish/subscribe solutions that are more apt for real time applications. Multiple semantics solutions for IoT described in our previous paper for the DEWI architecture also fall in the information model here described. Resource description and discovery schemes with indexes or addresses is necessary in any information model (e.g., UDDI). This allows users of the information model to discover, invoke and use the available network resources through service oriented architectures.

Regarding AI solutions, there was no major modification of the protocols and transport/messaging solutions to enable the new layer of intelligence services. InSecTT proposed a unified database or repository of data sets generated for the different AI algorithms of the different sub-building blocks. The format used is mainly CSV, but there have been several exceptions to allow for different types of data. Several open issues were identified for future data set representation and encoding. In principle, the information collected from different parts of the network to provide the training of the learning models can provide a strain to some hardware and software interfaces, particularly those located at the edge of the network and in the lower layers. In addition, the work on verification and validation (V&V) of AI algorithms under impairments and security attacks highlights the issue of protecting the data sets and the learning infrastructure, leading to solutions for encryption, authentication, encoding, redundancy, etc. This means another increase of stress on the supporting infrastructure. In the trustworthiness studies, InSecTT also identified the need to protect the privacy of the data used for training the learning models, and perhaps extend the privacy protecting regulation to the storage of data sets, or even the information learned from the data sets (learned weights). This means ensuring the end user that privacy is protected in the full cycle of the learning and actuation process of future AI implementations.

9 AI Perspective of the Architecture

The work in InSecTT proposes an advance in the state of the art on how AI tools have an impact on IoT reference architectures. More specifically, our work has looked into the details of the different AI algorithms implemented in the different industrial domains and we have provided a recommendation on how AI can be adapted into modern IoT architecture or how to officially implement a new perspective of AI in the architecture (Fig. 7).

The AI algorithms can eventually form one or more perspectives that complement or that extend the views of the RA. The prime candidate was the addition of sublayers to the functionality model regarding different sub-functionalities that are common to typical AI algorithms. An example of modified functionality layer with specific

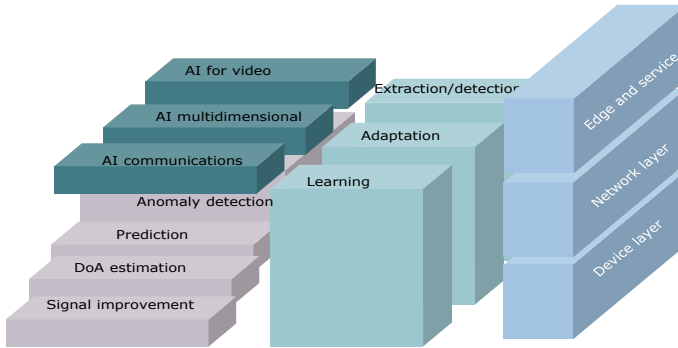


Fig. 7 3D-model AI perspective of the RA

AI of th project wsteps such as learning, feature extraction, class detection, model optimization, etc. is shown in Fig. 8.

A mapping of the requirements related to the different AI algorithms as conducted, producing a type of heating maps where the main AI-related functionalities reside in the functionality model of the project. An example is shown in Fig. 9.

The final AI perspective of the InSecTT RA is a 3D representation of the functional model of the RA (see Fig. 7). The extension consists of the functional decomposition of the learning mechanisms: data set collection, learning, training, storage, actuation, detection, observation, etc. In addition, we consider a second layer of the AI perspective that consists of the sub-building block representation of the project. This

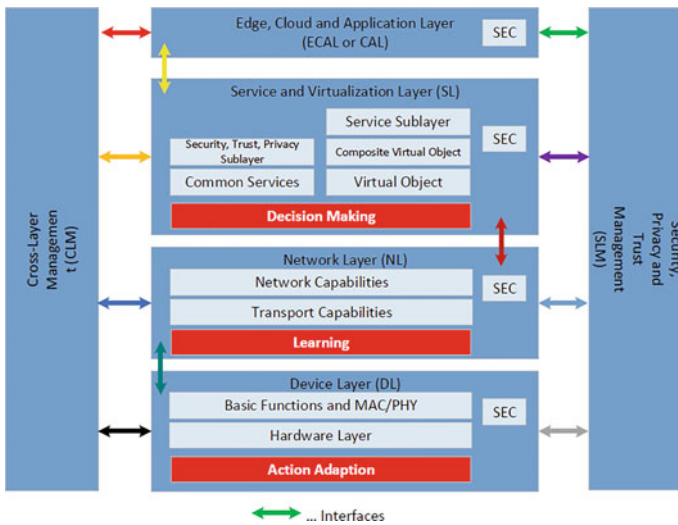


Fig. 8 2D-model AI perspective of the RA

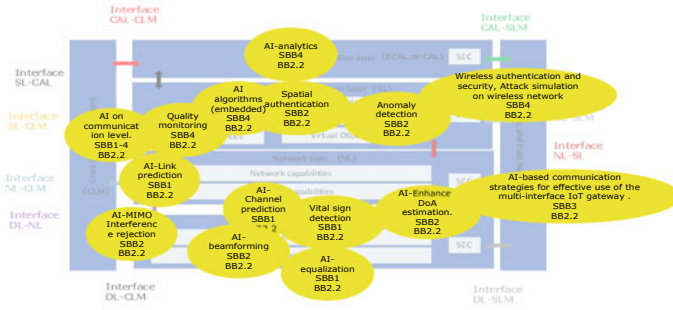


Fig. 9 Mapping of functionalities of AI algorithms of BB2.1 AI for communications

second layer of aggregation complements the first level and groups the types of AI algorithms , applications and puts them in perspective to be used or mapped to the use cases of the project.

10 Example Use Cases Alignment

10.1 Overview

We will show two examples of use cases and their preliminary alignment with the InSecTT RA. The full analysis is out of the scope of this paper. Therefore, we focus on the general overview of the two central models of the RA for two selected use cases. The first use case refers to a recent technology called wireless avionics intra-communications (WAICs). The second one is in the automotive domain targeting AI for wireless platoon intra-communications.

The term WAICs is used to describe any wireless sensor and/or actuator network operating on board an aircraft. While WAICs have been tested using multiple technologies and different frequency bands to verify potential interference to on-board equipment, this technology has been recently standardized by the ITU (International Telecommunications Unions) in the frequency bands of 4GHz (see [17–20]). WAICs is expected to be used mainly to replace or provide redundancy of wired infrastructure, such as control, sensing, and equipment management on board aircraft. In terms of cable infrastructure, gains can be expected for the reduction of aircraft design complexity. Reduced cable infrastructure also leads to weight losses, which in turn minimize fuel consumption, improve operational ranges and/or increase the size of the payload. In terms of configurability, wireless technology provides over-the-air (OTA) management and troubleshooting capabilities that facilitate network control and operation. Finally, wireless links can reach places of an aircraft difficult to cover with cables, thus facilitating design and reducing maintenance and troubleshooting costs for aircraft manufacturers.

Platoons are sets of cooperative autonomous or semi-autonomous vehicles with similar or identical routes that act as a single entity in terms of control and communication. The vehicles are usually arranged in linear convoys that communicate with each other control decisions that are usually made by one of the vehicles acting as leader or by a road side infrastructure with nearly real time traffic control information. The reliability of communication with low latency between the vehicular entities is critical to avoid any potential issue in the coordination between vehicles that could lead to safety issues. The emergence of 5G/6G technologies targeting ultra-low values of latency will enable the control of multiple autonomous or semi-autonomous vehicles in multiple platoons assisted by edge/cloud infrastructure.

10.2 Entity Model

The ITU recommendations define two types of WAICs network topologies depending on the location: internal or external to the cabin. The gateways are positioned in places to provide good coverage for the intended applications. The entities of a WAICs network can be rearranged as a Bubble of the InSecTT reference architecture. Sensors or groups of sensors can constitute an InSecTT Bubble node. Several Bubble Nodes can form a Wireless Sensor Network (WSN) which is assumed to be controlled by a WSN Gateway (WGW). One or more WSNs can be designed to operate in different parts of the aircraft, using different channels, different frequency bands or different hopping or spreading sequences. This reduces interference between WSNs. All the WSNs that belong to the same Bubble are assumed to be controlled by a unique InSecTT Bubble Gateway (BGW). The Bubble GW is therefore the central control entity of all Bubble Nodes and WSNs inside the aircraft information system. The WSNs are thus interlinked to each other and to the Bubble GW using the internal aeronautics bus network. The most used standard is ARINC 664 or the commercial version called AFDX (Avionics Full-Duplex Switched Ethernet). This technology is a modified version of the Ethernet standard based on the concept of virtual links that ensure real-time and deterministic deadline allocation. The concept of Bubble is especially fit for aeronautical applications, where L1 is the internal, real time aircraft network, L0 is the wireless links, and L2 is the cloud external connection of the aeronautical Bubble. We should emphasize that there are other ways of configuring the aeronautical infrastructure to have different deployments of the InSecTT Bubble. For example, different Bubbles can be operating in the same aircraft using an external L2 technology to achieve communication between Bubbles. The use of one Bubble per aircraft is illustrated in Fig. 10 for the aeronautical use case.

In the automotive use case, each platoon can be considered as a Bubble, with the leader being the Bubble Gateway (see Fig. 11, top sub-figure). The Bubble Gateway uses a 5G link to connect to the Cloud. In addition, each node of the Bubble has also a link with the 5G BS/RSU. In this case, we can consider that the 5G RSU/BS is a direct connection with a virtual Bubble Gateway, as the 5G Base Station (BS) acts as relay and assistant of the main Bubble GW. This leads to an interesting feature

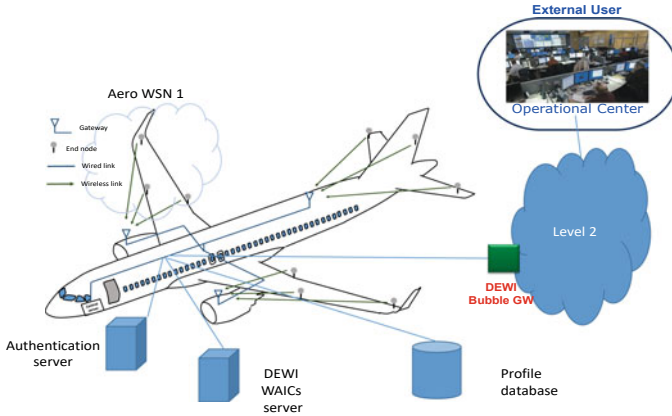


Fig. 10 Example of a WAICs network using the Bubble concept

of the Bubble and InSecTT architecture. The physical Bubble GW is not the unique access point to the Bubble Nodes from the external world. Nodes can have another link to the outer Bubble space using another direct cloud interface. This issue paved the way to the concept of virtual Bubble Gateway to control the connections to the Bubble using modern devices with multiple interfaces. This preserves the properties of the Bubble in a modern multiple interface environments.

The platoon-BS architecture can also be adapted in a different way to the InSecTT RA by considering that nodes can communicate with two WSN gateways over two different L0 technologies. The 5G link can be regarded as L1 technology, and the Bubble GW is represented by the 5G BS. This is also illustrated in Fig. 11 (the bottom sub-figure). This last option implies that the 5G BS or RSU are included in the Bubble, and therefore it can be inadequate for high mobility scenarios.

10.3 Functionality model

In both use cases we have mapped all the requirements to the functionality model in Fig. 5. This information is useful to identify the type of functionality needed in each scenario of the use case and the different interfaces with other functionalities or building blocks. The functionality model of the two uses cases are shown in Figs. 12 and 13 for the WAICs and platoon use cases, respectively. The detailed functional decomposition is the basis for trustworthiness metrics evaluation. Each function of a use case is weighted by a vector of trustworthiness metric using different models. An overall composite metric can be calculated per entity or per Bubble. This methodology allows us to find potential issues, vulnerabilities or strengths of different building blocks.

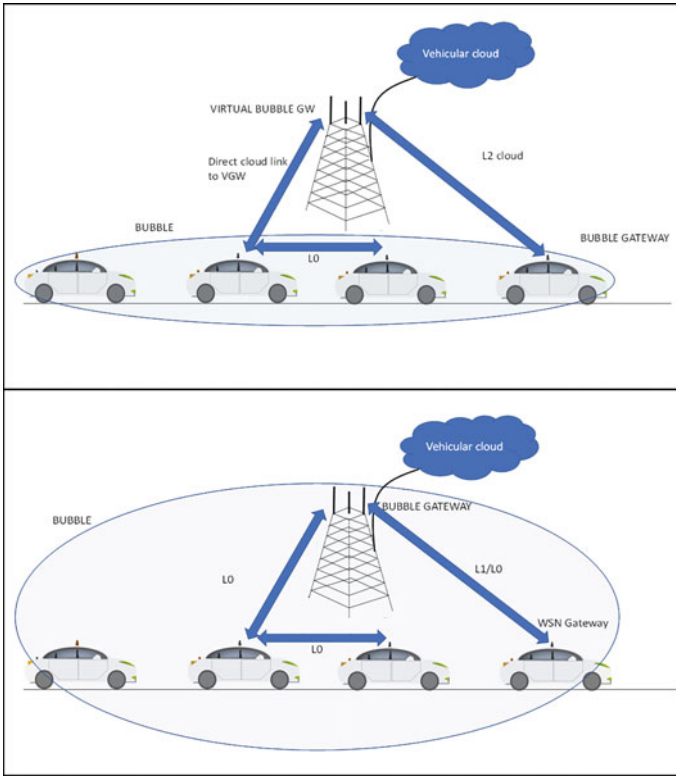


Fig. 11 Example of a Platoon network using the Bubble concept

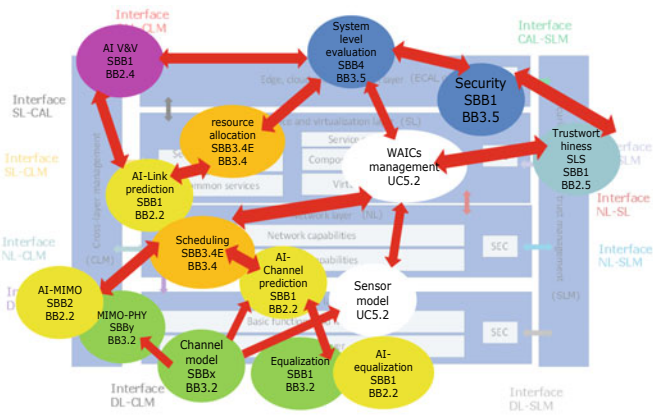


Fig. 12 Functional model WAICs use case

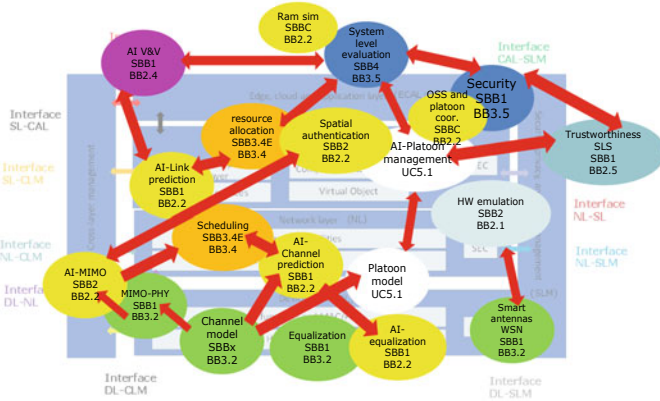


Fig. 13 Functional model platoon use case

10.4 Interfaces

The mapping between the entity and functionality models provides the detailed information of software and hardware interfaces. Interfaces between entities are hardware interfaces, while interfaces between layers of the functionality model are software interfaces. An example of this bi-dimensional mapping for the aeronautics use case can be seen in Fig. 14. This bi-dimensional mapping provides a good overview of the communication protocols per layer and per entity and the type of software related to the encapsulation of each layer of the functionality model. The guidelines for trustworthy design of the InSecTT RA include the expertise in the design of each one of these interfaces to improve a number of metrics or solve different security/safety issues. This is done using empirical and/or numerical metric models.

10.5 General Project Overview for Architecture Alignment

The main aspect considered in the preliminary analysis of the use cases of the project is the identification of the Bubble and possible configurations. In transportation systems, it was observed that at least two different ways are distinguished regarding the definition of the Bubble. For example, in autonomous diving use cases, the Bubble can be defined on each vehicle. However, in coordinated transportation systems, the Bubble can include multiple nodes or entities. Even in some cases, the Bubble may or not include the edge gateway in the road side units or the fixed access point. This selection depends on the needs of each use case. A similar approach can be followed for other types of use cases. For example, in the healthcare domain, the Bubble can be defined on the basis of isolation of entities or patients. Body area networks can

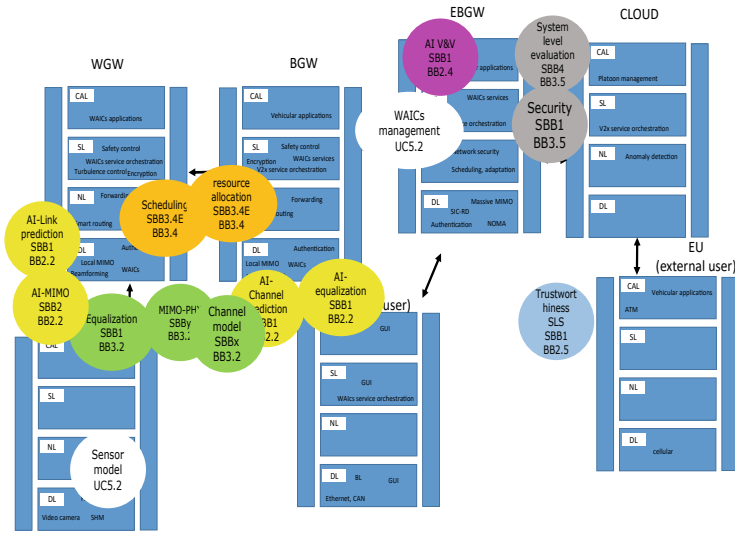


Fig. 14 Mapping functional versus entity models of the aeronautics use case

lead to define an individual Bubble for each patient, but in some cases it is better to define Bubble for a full patient room or ward. In manufacturing, the Bubble can also be defined using the isolation provided between Bubble gateways. We recall that each Bubble can have several wireless sensor networks, using L1 technology to organize and schedule a different WSNs with potentially different technology. This makes the Bubble concept very flexible to adapt to a variety of scenarios, even with dynamic decomposition of the Bubble. The concept of virtual gateway allows us to expand the concept of Bubble to long range direct cloud connections with 5G and 4G technologies. This adds an extra degree of flexibility with the definition of the Bubble that can be adapted to different scenarios in manufacturing, for example logistics, tracking, access control, and V2I solutions.

References

1. ETSI M2M architecture. <http://www.etsi.org/technologies-clusters/technologies/m2m>. Accessed 15 Oct. 2021
2. oneM2M architecture. www.onem2m.org. Accessed 15 Oct. 2021
3. Internet of Things Architecture. EU project. <https://www.iot-a.eu/>
4. ITU- Y.2060: Overview of the Internet of things (Reference Architecture). <https://www.itu.int/rec/T-REC-Y.2060-201206-1.29>
5. IEEE P2413. <https://standards.ieee.org/develop/project/2413.html>. Accessed Apr. 2018
6. ISO/IEC 30141, Internet of Things (IoT)-Reference architecture
7. Alliance for Internet of Things innovation. <http://www.aioti.eu/>. Accessed Apr. 2020
8. Secure Connected Trustable Things (SCOTT) EU ECSEL project. <https://scottproject.eu>

9. Dependable Embedded Wireless infrastructure (DEWI) ARTEMIS EU project. <http://www.dewiproject.eu/>
10. Intelligent Secure Trustable Things (InSecTT) ECSEL Grant Agreement, Number 876038-InSecTT (2020)
11. Kumar Singh, S., Rathore, S., Park, J.H.: BlockIoTIntelligence: a blockchain-enabled intelligent IoT architecture with artificial intelligence. *Future Gener. Comput. Syst.* **110**, 721–743 (2020). ISSN 0167-739X, <https://doi.org/10.1016/j.future.2019.09.002>
12. Calo, S.B., Touna, M., Verma, D.C., Cullen, A.: Edge computing architecture for applying AI to IoT. In: 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, 2017, pp. 3012–3016 (2017). <https://doi.org/10.1109/BigData.2017.8258272>
13. Wu, Q., et al.: Cognitive internet of things: a new paradigm beyond connection. *IEEE Internet of Things J.* **1**(2), 129–143 (2014). <https://doi.org/10.1109/JIOT.2014.2311513>
14. Karner, M., Hillebrand, J., Klocker, M., Sámano-Robles, R.: Going to the edge-bringing internet of things and artificial intelligence together. In: 2021 24th Euromicro Conference on Digital System Design (DSD), Palermo, Italy, 2021, pp. 295–302 (2021). <https://doi.org/10.1109/DSD53832.2021.00052>
15. ISO/IEC 29182, Information technology-Sensor networks: Sensor Network Reference Architecture (SNRA)-Part 1 to 7
16. Sámano-Robles, R., Nordström, T., Kunert, K., Santonja-Climent, S., Himanka, M., Liuska, M., Karner, M., Tovar, E.: The DEWI high-level architecture: wireless sensor networks in industrial applications. *Technologies* **9**(4), 99 (2021). <https://doi.org/10.3390/technologies9040099>
17. Technical characteristics and operational objectives for Wireless avionics intra-communications (WAIC) Report M.2197 (ITU-R Report). <http://www.itu.int/pub/R-REP-M.2197>. Accessed Dec. 2020
18. Technical characteristics and protection criteria for Wireless Avionics Intra-Communication systems, Recommendation ITU-R M.2067, approved Nov. 2014. <http://www.itu.int/rec/R-REC-M/recommendation.asp?lang=en&parent=R-REC-M.2067>. Accessed Dec. 2020
19. Technical conditions for the use of the aeronautical mobile (R) service in the frequency band 4 200- 4 400 MHz to support wireless avionics intra-communication systems, Report ITU-R M.2283, approved July 2015. <http://www.itu.int/rec/R-REC-M/recommendation.asp?lang=en&parent=R-REC-M.2085>. Accessed Dec. 2020
20. Technical characteristics and spectrum requirements of Wireless Avionics Intra-Communications systems to support their safe operation, Report ITU-R M.2283, approved Dec. 2013. <http://www.itu.int/pub/R-REP-M/publications.aspx?lang=en&parent=R-REP-M.2283>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Structuring the Technology Landscape for Successful Innovation in AIoT



Peter Priller and Michael Jerne

1 Motivation

Digitalization has entered our lives in many subtle, and not so subtle ways. Many of us got used to witness one technology revolution after another. Incredible wealth, at least measured by stock value, has been created in comparatively short time by companies in the IT domain, most prominently those called Big Tech.

Digital computers have gone a long way, from huge mainframes filling data centres to personal workstations, and can now be found as smartphones in our pockets and even close and in our bodies as wearables. But the real breakthrough came by combining those approaches. Linking powerful computing centres (now called cloud) with personal devices and local smart devices, embedded in everyday objects, via network of networks, opened plenty opportunities for digitization. Something we have started to call the Internet of Things (IoT), joining locality and interaction in the physical world we live in, with digital services across layers unlocked the virtual world.

But how is this digital world built? Similarly to digitization itself, engineering digital systems has seen (and still experiences) its fair share of fast-paced revolutions. For a long time, computing paradigms have been governed by the von-Neumann architecture, and software was mostly developed by legions of human programmers, explicitly writing down the desired behaviour using some of a wide variety of computer languages. After decades of research and development, it appears that this paradigm is to change, or at least it is joined by an alternative way: having the computer to learn (and at some time, to reason) what we expect it to do. Branded as AI, with ML being arguably the most well-known subset, we start seeing real progress

P. Priller (✉)
AVL List GmbH, Graz, Austria
e-mail: peter.priller@avl.com

M. Jerne
NXP Semiconductors Austria GmbH & Co. KG, Gratkorn, Austria

in real-world problems, like object detection, speech recognition or automation of simple administrative tasks.

The InSecTT consortium was established in 2019 with a ground-breaking vision to combine these two approaches: IoT and AI, or in short: $AI + IoT = AIoT$. Also considering aspects of security will finally ensure that the AIoT will be resilient and trustworthy and thus accepted by users.

2 How to Structure Research and Development to Enact an Ambitious Project Vision

With the two base ingredients, IoT and AI, being already such huge fields of technologies, the question came up quite early during the conception of the InSecTT project. It became even more relevant when planning and implementing the concrete work: how to structure these fields?

Structure is needed for several reasons: (i) to capture the current state of the art for the fields in scope and identify the gaps, (ii) to formulate research questions and problem descriptions setting objectives for the project, (iii) to identify stakeholders and form teams to handle the tasks efficiently in the consortium, (iv) to identify relevant technology constraints and system boundaries.

Another important aspect comes from the fact that project InSecTT intends to demonstrate all research results by applying into real-world use-cases. InSecTT is designed as an use-case driven, pan-European project, showing solutions in domains like sustainable mobility, smart health, or smart production. This required the consortium to find a structure which is able to detail technology needs and technology results. It allows to link and map technology providers with technology users, to ensure all demonstrators can be implemented to the desired technology readiness level (TRL, for most use cases between 5 and 7) within the planned project duration of 3 years.

The following sections in this chapter describe the approach chosen by the InSecTT consortium to tackle this challenge.

3 Requirements and Constraints

The InSecTT consortium designed the processes and structures used to navigate the technological landscape in scope of the project, and to govern the research and development work spread over multiple work packages and tasks.

While overall project objectives clearly have served as reference and starting point for all technical work in InSecTT, an effective and efficient implementation also had to consider the high number of project partners spread over different countries and cultures, as well as the diversity of addressed domains and related interests,

including aspects of IP management and know-how sharing. It needs to be aware of and support the fact that partners follow their individual innovation processes and might have different IP strategies (e.g., open/shared/protected). Given the objective to ensure multi-domain re-use of technical building blocks, a wise balance between re-usability on one hand, and “fit-for-purpose” usage in Use Cases on the other hand had to be found. For the requirements engineering process this means that a top-down approach (Use Cases to define their requirements for components) has been leading, but in parallel and driven by technology experts, bottom-up input has been made to find the best possible balance between both challenges.

4 Requirement Engineering Process

An important part of any engineering endeavour is to understand the requirements and expectations of the application domain in order to drive the design. In order to foster both, deep-dive research combined with applied engineering, the consortium settled on an iterative process, dividing the project runtime into 3 iterations (Y1, Y2 and Y3). While the first iteration was dedicated to classic requirement engineering and first technology evaluations/demonstrations, Y2 and Y3 allowed to integrate new inputs during the project, react to requests from the application domain in a more agile way, and to learn from results of the first generations of demonstrators.

An important decision to make was the intended level of detail, the granularity, of requirements defined in this process. A typical product development process usually needs detailed specification of functional and non-functional requirements, with unambiguous description of each and every aspect (agile processes might accumulate this information over several iterations). In project InSecTT however, the requirements serve primarily as an interface between technology work packages (WP2, WP3) and use cases (WP5), specifying the needs for research and technology development. The consortium therefore decided deliberately to focus on and describe domain- and application specific needs and constraints through the requirement specification, and to define functionalities on a rather high level only.

Each requirement is mapped onto one technology Building Block (BB), and defines clear role responsibilities. A partner of the consortium needs to take on the *lead implementer* role, and to confirm the requirement as a first step. With these four entities (requesting Use Case, supplying building block, lead implementer, requirement author), all required relations and responsibilities for the life cycle of the requirement were defined.

As requirements are independently created by different use cases, requesting certain technologies from a specific BB, they might address similar but not or only partly congruent aspects. InSecTT uses therefore the *requirement harmonization process* during this first phase in the project, where requirement creators and technology supplier can negotiate, trying to refine the requirements in a way which optimizes synergies through reuse and interoperability of the to-be-developed BBs.

Finally, the iterative approach towards defining requirements described above also allows a clearly structured change process for requirements.

5 Navigating the Landscape: Planning R&D Work

After setting the vision and defining the mission, followed by a definition of the main use cases by the industrial partners, the main focus of the consortium was on defining the technology landscape and setting up processes and data structures to guide the main work streams in InSecTT. Again, the concept of re-usability on one hand and clear accountability in supporting use cases on the other hand have been the leading principles. From the requirements defined (see above) the consortium derived and further detailed the research and development needed to satisfy these requirements.

Therefore both, methodology as well as execution approach used in InSecTT technical Work Packages, have been defined in a way that (a) is in line with the overall InSecTT implementation framework, (b) enables lean but still efficient WP- and Task management, as well as interaction within the overall InSecTT governance framework, (c) allows partners to mostly focus on content related technical work and cooperation, and (d) provides an unified model between the two technical work packages, in order to foster co-operation and joint innovations.

This unified model has been developed in a cooperative project effort and has led to a three-layer approach as far as the technical work (i.e. the work on the technical Building Blocks) is concerned:

- **Layer 1: Building Block (=Task)**

This layer reflects the high-level structure as used in the project submission and for managerial and reporting purposes. It is absolutely needed for organizational reasons, but it has been experienced as being too comprehensive and too diverse from a technical perspective to ensure clear ownership on partner level and to facilitate efficient cooperation. As a result, the Sub-BB concept has been developed and introduced in InSecTT.

- **Layer 2: Sub-Building Blocks**

This layer has been developed as the main layer to organize and manage *technical cooperation and alignment*, offering the right granularity level for joint development targets and to exploit potential development and exploitation synergies. Also (technical) dependencies have been highlighted mostly on this layer.

- **Layer 3: Component**

This layer has been introduced to understand, how the different partner level contributions do contribute to certain Sub-BBs and how they feed into the UC integration work. This also ensures clear accountability on partner level. “Component” in this context can mean any contribution, be it hardware (HW), software (SW), methodology, or any combination of them.

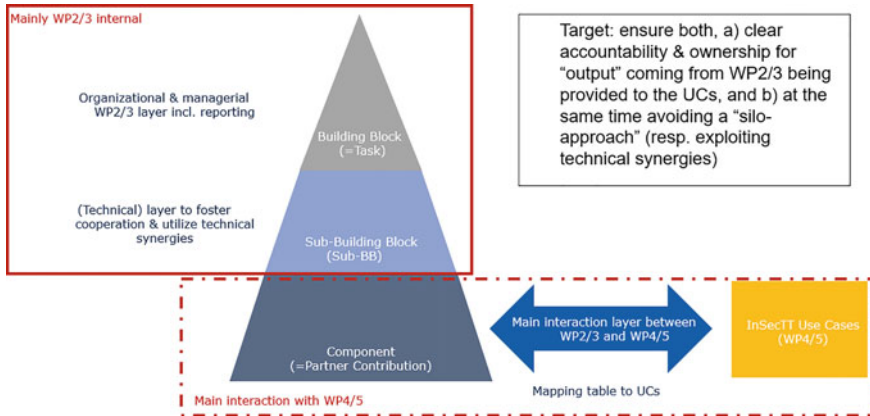


Fig. 1 InSecTT layer model and terminology

Figure 1 highlights how the three layers interact.

We see this three-layer model as an appropriate methodology to effectively address the significant size and the high complexity and diversity of a project like InSecTT. The target is to not de-focus partners from their technical work by too much managerial and methodological “overhead”, but to still foster and facilitate cooperation and synergies as much as possible. In addition to the mapping table between use cases and (Sub) Building Blocks it offers an efficient methodology for a project of the size of InSecTT to stay on top of activities, both in terms of technical work and use case integration activities, while still ensuring clear accountability on one hand and facilitating technical cooperation on the other hand.

As an example for the structure described, Fig. 2 depicts the composition of BB 3.5 (Verification, Validation, Accountability) into sub-BB (A..D) and respective components (a..h).

The final result of this structuring work provides the complete map of research and development in the technological landscape of AI and IOT. It provides full traceability top-down from the application (the use case) through specified requirements to technology needs, and maps these into building blocks (BB), sub-building-blocks (sub-BB) and components as defined. In the other direction (bottom-up) it allows to align and aggregate activities in the technological fields, with the goal to support re-use and interoperability through compatibility for the components developed.

To facilitate these aspects, all BB and sub-BB were analyzed to identify their relationships (dependencies, input, output). Figure 3 shows Sub-BB 3.1.B (Intrusion detection systems) as an example.

Another aspect of planning technical work is to identify if there are any gaps in the landscape designed, in other words if there could be technologies missing, but required for the use cases defined. One way to do this analysis is to map technologies against layers, as shown for example in Fig. 4 (shown for WP2).

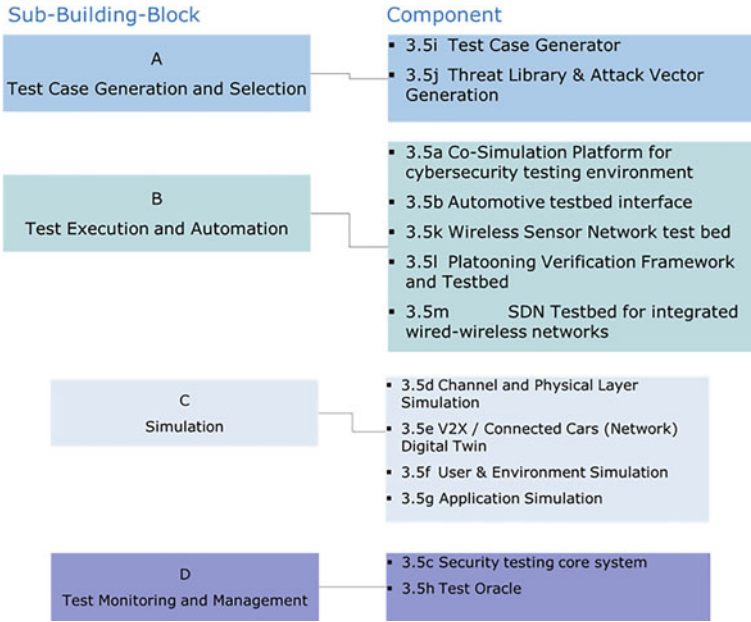


Fig. 2 Structure of BB 3.5 (verification, validation, accountability) into sub-BB (A..D) and respective components (a..h)

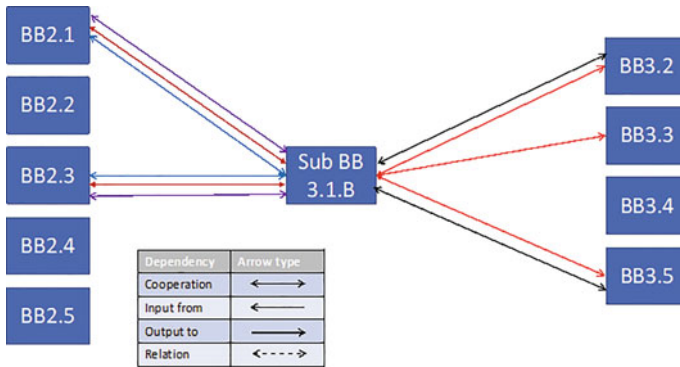


Fig. 3 Relationship analysis on sub-BB level (example: Sub-BB 3.1.B)

6 External Alignment

Along with the structuring and mapping done within project InSecTT it is certainly beneficial to align with external architectural models and research agendas. Highly relevant is the Electronic Components & Systems (ECS) Strategic Research and Innovation Agenda (SRIA) released by the three Industry Associations AENEAS,

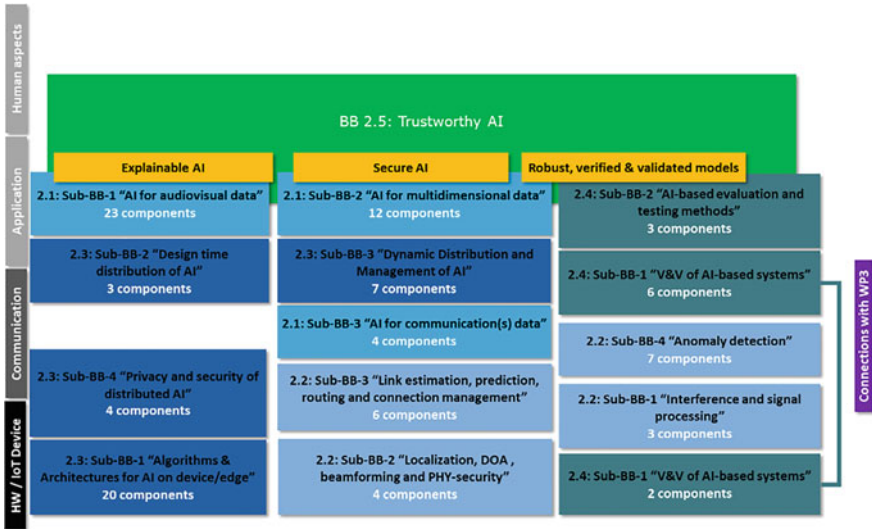


Fig. 4 Mapping technologies along layers

EpoSS, and Inside, under the joined umbrella of the KDT JU (Key Digital Technologies Joint Undertaking) (since end of 2023 called Chips JU). A more detailed discussion and analysis of the technologies and mappings can be found in chapter “Reference architecture for AIoT systems”.

Even a large assembly of experts like the InSecTT consortium can benefit from additional expertise from outside. This is why during the first two years several liaisons were established between InSecTT and other European research projects running in parallel:

- ERATOSTHENES, IoT Trust and Identity Management Framework, funded under H2020-EU.3.7.4. (Liaison established 2021)
- DAIS, Distributed Artificial Intelligent System, a KDT JU project (liaison established 2022)
- ANIARA, Automation of Network edge Infrastructure and Applications with aRtificAI intelligence, an EU flagship project, focusing on edge cloud (liaison established 2022).

7 Documenting Scope, Work and Results

An important aspect of research is to communicate its existence, its progress and especially its results to the intended target groups. Naturally, this faces challenges like (i) efficient soliciting and collecting information in a large and heterogeneous team of a cross-European project, (ii) finding suitable channels for communication, and (iii) processing and transforming information to draw the interest of target groups,

while honouring the interests and potential confidentially aspects of project partners. InSecTT is set up so that most dissemination and exploitation is done directly by the partners. However, a dedicated task (T6.1, Dissemination and Exploitation) was established to support such activities, by creating opportunities, networking, counselling partners etc.

InSecTT has been designed in a way that motivates broad exploitation of the work done in the technical Work Packages (WP2 and WP3), beyond the integration into the InSecTT use cases (WP5). Progress of the technical work was reported annually, highlighting progress beyond state-of-the-art and planned exploitation activities. Two public deliverables at the end of the project, summarizing technological step-ups achieved in WP2 and WP3 support broad dissemination of results.

Each use case is represented by a task in WP5, forming T5.1 through T5.16.

A dedicated work package WP4 (Cross-domain use case coordination) supports identifying and exploiting synergies between use cases, and provides uniform means of documentation and reporting like the Use Case Booklet (<https://www.insectt.eu/download/d4-4-insectt-industrial-use-case-booklet/>).

Another example for unifying use case management is the common deliverable ('D') structure and release schedule all use case have to follow:

- D5.1-5.16 Use Case specification (M6)
- D5.17-5.32 Use Case progress report Y1 (M11)
- D5.33 InSecTT industrial demonstrators Y1 (aggregates all use cases, M12)
- D5.34 Use Case specification update (aggregates all use cases, M18)
- D5.35-5.50 Use Case progress report Y2 (M23)
- D5.51 InSecTT industrial demonstrators Y2 (aggregates all use cases, M24)
- D5.55-5.71 Use Case progress report Y3 (M36)
- D5.72 InSecTT final demonstrators (aggregates all use cases, M36).

8 Progress Assessment and Validation

InSecTT is a use-case centred project. As a result, research work and development of technologies are guided by the needs specified in the use cases. Structuring, managing and monitoring of work is important, especially in a large, distributed team of a large consortium.

R&D results are supposed to fill the gaps (as identified by the use cases) in the technological landscape. But how can the consortium, and in general any stakeholder, validate that the results really fit the landscape as intended? How can we evaluate progress of work, as well as its practical use?

Project InSecTT uses requirement-based progress assessment. The status (definition, confirmation, grade of implementation) of each requirement is individually assessed by the requirement owner (typically the stakeholder who defined the requirement). This is done as a coordinated effort in the whole project in three assessment time windows throughout the three years of the project. As all requirements are



Fig. 5 A mandatory attribute of each requirement is its state

managed in a SharePoint repository, so can the assessment be done interactively in SharePoint as well. Each requirement has a state, see Fig. 5.

The fine granular progress assessment is done by the owner of the requirement. This person has to provide a number (0..100%) as progress estimation on technical level. As a pragmatic guideline, the following definitions are suggested:

- 40% ... Implemented on BB level to small extent
- 70% ... Implemented on BB level to large extent
- 100% ... Fully implemented on BB level.

One number per requirement can be easily used to summarize and report, but obviously is quite a generalization and might aggregate too many aspects. This is why each assessment can be accompanied by a short, written comment (free text), to explain the rationale or describe the context.

Results are used to manage and monitor work on individual requirements level, as well as on BB level and on UC level, see Fig. 6.

9 Demonstrators

Demonstrators are a major pillar of InSecTT. Typically, demonstrators integrate implementations of research results from one or more partners in a prototypical way, and evaluate it in the intended context (TRL 6: **Technology** demonstrated in

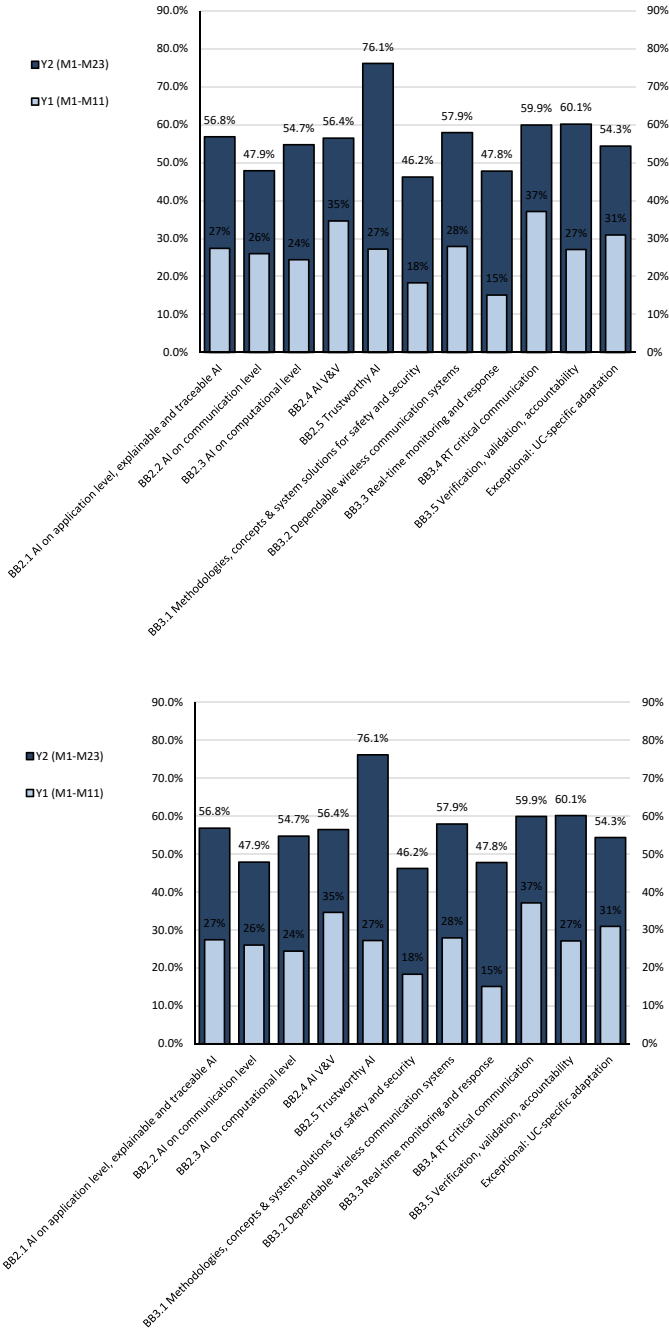


Fig. 6 Evaluation of requirement assessments on BB and on UC level

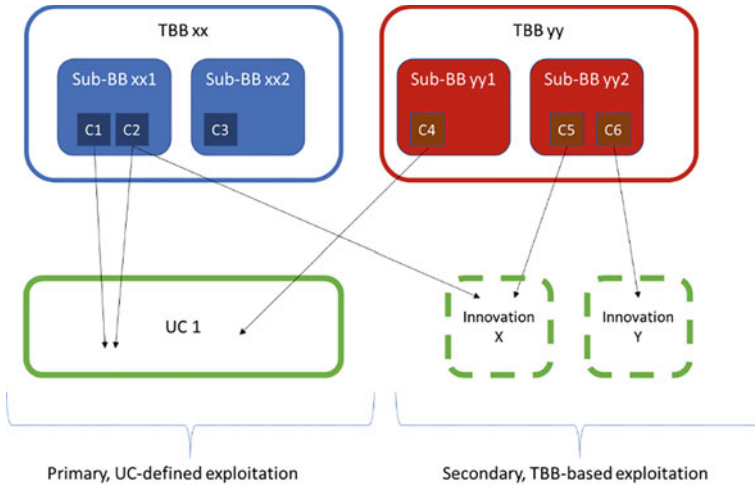


Fig. 7 Primary and secondary paths of exploiting research results; TBB denotes “technical” BB

relevant environment; TRL 7: **System prototype** demonstration in *operational* environment). Demonstrators therefore serve as inevitable check point for the partners involved. But there is more: demonstrators are also used to prepare markets as well as stakeholders like potential users and customers. Project InSecTT is structured in three distinct 12 months long iterations, having partners providing at least yearly generations of demonstrators. Demonstrators can be on UC (primary) or BB level (secondary), see also Fig. 7. Besides used as checkpoint and for validating the results, demonstrators are also used to introduce and present technologies, to foster generating new ideas and therefore opening (additional) paths of exploitation, e.g. at the Use Case Marketplace (Fig. 8).

10 Preparing for Market: Exploitation

The explicit goal of all research and development done in this project is to apply its results towards progressing European commercial success and to raise the quality of living, especially regarding safety and security, for Europe’s citizens, and in general to benefit humankind.

All partners in the InSecTT consortium, strongly encouraged by the EC and national funding agencies as providers of the funding, therefore focus on supporting **exploitation of results**. This is supported in the project by a portfolio of activities, with some described below.



Fig. 8 UC Marketplace at the F2F Meeting in Gdansk, Poland (2022-06-13)

Groundwork has been laid already during writing the proposal for the project. After analysing the current level of technologies and applications and from that identifying the gaps, which led to the definition of InSecTT’s vision $AI + IOT = AIOT$, partners have specified **concrete industrial use cases** in WP5, applying the technologies developed in WP2 and WP3. These use cases have been developed from the domain knowledge of the industrial partners in the consortium and are supported by analysis of market needs. As a result, in total 16 use cases have been described, and embody the primary way of commercialization of project results.

Use cases demonstrate technologies in the context of the “real world”, typically at technical readiness levels of TRL 6 and above. The proposal writing stage also included a brief overview of market and **exploitation strategies of the industrial partners** to ensure it will match.

The second path towards commercialization is on the level of **technical building blocks** (TBB’s) derived from the BBs developed in WP2 and WP3. While the UC path typically defines markets and their access, exploitation on TBB level can be more open regarding target markets. This is both opportunity and challenge, which is why project InSecTT has set up the Exploitation Board, and organizes activities like an internal Open Innovation Contest (OIC 2022, Fig. 10) and Use Case marketplace (see Fig. 8). An overview depicting these paths of exploitation is shown in Figs. 7 and 9.

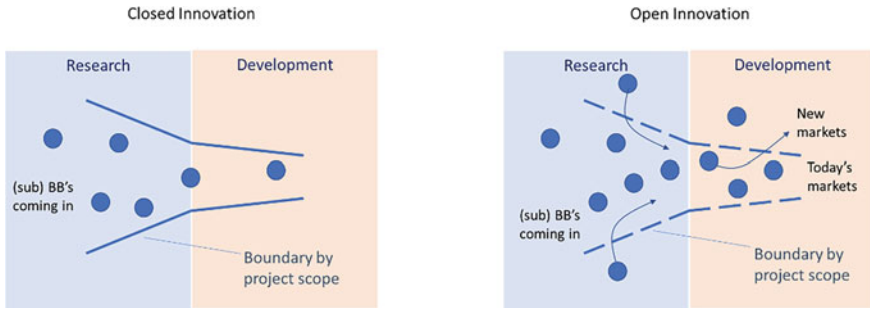


Fig. 9 Open innovation (inspired by Chesbrough (2003b): the era of open innovation. MIT Sloan management review, 2003)



Fig. 10 Trophies from the internal open innovation contest (Gdansk 2022)

11 InSecTT Exploitation Board (EB)

In order to further foster exploitation, and to support all partners, especially SME's, in the consortium, the **InSecTT Exploitation Board (EB)** has been formed. Confidentiality is ensured via the project consortium agreement (PCA).

The EB has an advisory function regarding the exploitation strategy and forms a natural interface between confidential exploitation plans created by the parties and all important stakeholders of InSecTT.

In short: Exploitation Board activities are focused on (increased) monetization of the project outcomes, aiming

- to analyse exploitation plans provided by partners, and to look for synergies within and also outside the consortium.
- to support partners in development of common exploitation strategy and corresponding activities that should be undertaken; and

- to prepare documents in coordination with the Strategic Board that can be passed to third parties, e.g. SMEs outside of the InSecTT consortium, or specialized organisations that can support InSecTT’s long-term strategy.

12 Use Case Marketplace

Use cases provide the primary path towards exploitation of technologies in the project. InSecTT has defined and organized so called Use Case Marketplaces (see Fig. 8), in order to

- demonstrate results in use cases (and naturally of the TBB’s used in the UC),
- gain in-time feedback for technical work from first implementations,
- allow to discuss results to fertilize interoperability and cross-domain use,
- support discussions to open additional exploitation opportunities for TBB’s used in the use case.

The project is designed into three iterations, therefore allowing typically three (in some cases more) generations of UC demonstrators, with increasing maturity.

13 Open Innovation

Open innovation applies know-how from “outside” during research and development, and allows to engage stakeholder already during R&D, see Fig. 9.

InSecTT developed its Open Innovation process in three steps, starting with a small and confined test run with just a few partners from the consortium and with a limited time window. The 2nd iteration was advertised in the full consortium, but still participants were only allowed from within the consortium.

A final event handing out in total 10 awards at the F2F Meeting in Gdansk (June 2022) provided enough motivation to solicit many interesting ideas already. As this was still done within the consortium only, participants could exchange ideas without being restricted by any NDA considerations.

The main concept, shown in Fig. 11, is based on

1. a **description of a component** which is already available or under development in the project, e.g.: hardware solution, (AL/ML) algorithm, methodology, software, embedded device). The description is drafted by the lead implementer (called Partner A) in Fig. 11
2. **ideas**, e.g. suggested by partner Z, bringing this component into a new use case, e.g., a new application area, or exploiting new synergy with an existing product, to provide additional benefits, address new market or users

The expected effects can be summarized as

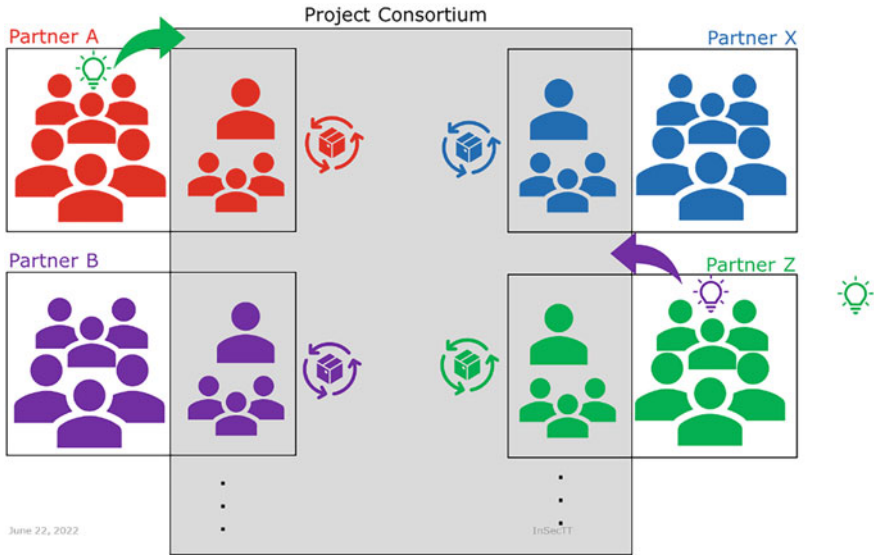


Fig. 11 The main concept of the 2nd iteration of an open innovation contest in InSecTT

- New ideas for InSecTT technology development (small adaptations for accessing new potentials in new use cases and new innovations)
- Alternative use cases opportunities and possible B2B collaboration
- New project proposals.

14 Publications to Prepare Markets

Quite naturally as in most research projects, presenting papers at conferences or putting publications in journals and other platforms are used extensively to promote results and to solicitate feedback from the research and engineering community.

The addressed stakeholders of such channels typically include academic communities, researchers and experts from the field of policy, science, and industry, students (Ph.D./Master) and applied researchers in industry, and are therefore probably more focused on results with lower TRL.

This is supported by organizing workshops and special sessions as part of established conferences, again to provide a platform for discussion and networking in the research community.

In 2021, the InSecTT consortium applied for and was granted to organize a full-day workshop at the IEEE WF-IoT 2021 conference, title: Wireless Intelligent Secure Trustable Things: bringing IoT and AI together (<https://wfiot2021.iot.ieee.org/wst-rack3/>).

In 2022, the InSecTT consortium applied for and was granted to organize a workshop at the MIKON 2022 conference, title: Focused session 1—Intelligent, Secure and Reliable Wireless Systems (<https://mrw2022.org/mikon-focused-sessions/>).

15 Website and Social Networks

A key factor to reach stakeholders and build networks is an informative, easy-to-find project website (<https://www.insectt.eu/>). Frequent updates provide current project news to the public, and allow a low-barrier introduction of project objectives, activities and partners and contacts. It also acts as repository for (public) deliverables.

Quite important is to accompany this web presence by appropriate activities using social media. InSecTT uses multiple channels of social networks, to provide fast, low-latency updates about InSecTT activities, and to reach the broadest spectrum possible of potential stakeholders and interested citizens (Tables 1 and 2).

16 Industrial Conferences, Trade Fairs and Podcasts

The project started in June 2020 and finished in August 2023. Unfortunately, it has therefore experienced constraints and challenges coming from a world-wide COVID-19 pandemic and a war between Russia and Ukraine, severely impacting both world economy and typical research project dissemination and exploitation support actions (travel, conferences, trade fairs etc.). Some remedies have been initiated by the InSecTT consortium, like releasing Podcasts (<https://podcasts.apple.com/at/podcast/project-insectt/id1605747720>) as a low-barrier channel to reach potential stakeholders, see Table 1.

Table 1 InSecTT’s podcast channel

Release date	Title	Content	Duration
6.3.2023	Venkatesha Prasad from TU Delft, Netherlands: is the revolution of the Internet of Things already over, or are we still at the beginning?	Today, Venkatesha Prasad (known as 'VP') tells Anamarija about how he came from small-town India to TU Delft in the Netherlands. VP explains the research he and his team at TU Delft do in InSecTT, and what he expects to see in IoT for the near future coming up. So ... is the revolution of the Internet of Things already over, or are we still at the beginning? Tune in to find out	27 min
11.11.2022	Lukasz Kulas from University of Gdansk on smart ideas and open innovation	In this episode, Anamarija talks with Prof. Lukasz Kulas from Gdansk University about smart ideas for using secure connected things in real life. For example, about retrofitting ships and harbors, or localizing medical devices in a hospital. Lukasz has also organized Open Innovation and Student Contests, and talks about how creativity can lead to cool innovations. These are good examples for bringing students, scholars and industry together to spark new ideas (and have fun)	22 min
22.9.2022	Do you trust AI? How to make things “trustworthy” with Peter Mörtl	Do you trust AI? How to make things “trustworthy”? Anamarija interviews Peter Mörtl from VIF about what “trust” really is, how to make things trustworthy and what the research project InSecTT contributes through its “Trustworthiness Framework”. s InSecTT is also about developing AI methods and technologies for future applications, the question needs to be asked: can we trust AI?	14 min
29.8.2022	Michael Karner from Virtual Vehicle	Today’s guest is Michael Karner, the coordinator of project InSecTT. Michael talks about the challenges of coordinating such a large project as InSecTT, especially during the Covid-19 pandemic. What is the “secret sauce” from managing a successful project?	18 min

(continued)

Table 1 (continued)

Release date	Title	Content	Duration
1.7.2022	Johannes Peltola from VTT	Johannes Peltola from VTT talks about developing AI building blocks in InSecTT. Is data the new oil? What applications are there being researched in InSecTT? And how hard is it to lead the work package #2 in such a large project?	26 min
5.5.2022	Markus Pistauer from CISC on trustworthy IoT	In this episode, Anamarija talks with Markus Pistauer, CEO of CISC Semiconductors. Markus explains his vision for trustworthy, intelligent IoT, and how this will shape our future. We also hear some thoughts on how collaboration in such a large project works	18 min
23.4.2023	Ramiro Robles from ISEP Portugal	Today, Peter (substituting for Anamarija) interviews Ramiro Robles from ISEP. What is a “high level architecture”? How is it used in such a large project like InSecTT? And why is trustworthy, intelligent IoT important for airplanes? Ramiro has the answers...	20 min
8.3.2022	Michael Jerne from NXP	Anamarija interviews Michael Jerne from NXP about the complexities and benefits of managing a large technical work package in InSecTT	28 min
9.2.2022	InSecTT: From narrow to Ultra-Wide Band	In this podcast, Anamarija interviews Andreas Springer, head of the Institute for Communications Engineering and RF-Systems at JKU Linz. Today he will talk about JKU’s involvement in InSecTT, and what a trustworthiness indicator has to do with ultra-wide band (UWB) radios. This work is part of a joined European research project InSecTT: Intelligent Secure Trustable Things, https://www.insectt.eu/ . The project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876038. The JU receives support from the European Union’s Horizon 2020 research and innovation programme and Austria, Sweden, Spain, Italy, France, Portugal, Ireland, Finland, Slovenia, Poland, Netherlands, Turkey. The podcast reflects only the interviewee’s view and the Commission is not responsible for any use that may be made of the information it contains	21 min

(continued)

Table 1 (continued)

Release date	Title	Content	Duration
17.1.2022	InSecTT: What about Automotive Security?	<p>In this podcast, Anamarija interviews Stefan Marksteiner from AVL. Stefan will explain "What to do if you don't want your car to be hacked"</p> <p>This work is part of a joined European research project InSecTT: Intelligent Secure Trustable Things, https://www.insectt.eu/. The project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876038. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Sweden, Spain, Italy, France, Portugal, Ireland, Finland, Slovenia, Poland, Netherlands, Turkey. The podcast reflects only the interviewee's view and the Commission is not responsible for any use that may be made of the information it contains</p>	12 min

Table 2 Social networking channels used by InSecTT

Channel	Link
LinkedIn	https://www.linkedin.com/groups/12467950
Facebook	https://www.facebook.com/Insectt-Project-105974577981213
Twitter	https://twitter.com/InsecttProject
YouTube	https://www.youtube.com/channel/UC27HebrTM0MBHKS8yDvwucA
Instagram	https://www.instagram.com/insecttproject

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Technology Development

InSecTT Technologies for the Enhancement of Industrial Security and Safety



Sasikumar Punnekkat, Tijana Markovic, Miguel León, Björn Leander,
Alireza Dehlaghi-Ghadim, and Per Erik Strandberg

Abstract The recent advances in digitalization, improved connectivity and cloud based services are making a huge revolution in manufacturing domain. In spite of the huge potential benefits in productivity, these trends also bring in some concerns related to safety and security to the traditionally closed industrial operation scenarios. This paper presents a high-level view of some of the research results and technological contributions of the InSecTT Project for meeting safety/security goals. These technology contributions are expected to support both the design and operational phases in the production life cycle. Specifically, our contributions spans (a) enforcing stricter but flexible access control, (b) evaluation of machine learning techniques for intrusion detection, (c) generation of realistic process control and network oriented datasets with injected anomalies and (d) performing safety and security analysis on automated guided vehicle platoons.

1 Introduction

Industry 4.0 is aiming towards convergence between industrial systems and IT infrastructures to enable higher levels of productivity through information sharing among all stakeholders. Digitalisation, automation, autonomy, artificial intelligence (AI), cloud computing, higher connectivity are regarded as key drivers the next industrial revolution. Breaking the traditional 5-level automation pyramid (“Purdue”) architecture to enable interoperability, autonomy and seamless data transfers however comes

S. Punnekkat (✉) · T. Markovic · M. León
Mälardalen University, Västerås, Sweden
e-mail: sasikumar.punnekkat@mdu.se

B. Leander
ABB AB, Västerås, Sweden

A. Dehlaghi-Ghadim
RISE, Västerås, Sweden

P. E. Strandberg
Westermo Network Technologies AB, Västerås, Sweden

with some concerns, especially with respect to safety and security in the traditionally rigid industrial segments.

As the future factories are envisioned to be flexible, adaptable and collaborative endeavours involving man and machines (“autonomous robots”) forming complex system of systems, their emergent behaviours are quite hard to fully characterise at design time. Majority of the factories also can be termed as safety critical systems since failures can often lead to adverse impacts not only on productivity but also on humans, infrastructure and environment. Here comes the mandatory and often legal safety requirements set forth by various generic/domain specific standards and machine directives.

Security is one of the major focus aspects of the EU-funded InSecTT project (<https://www.insectt.eu/>). The ever-increasing landscape of cyber security threats, together with higher levels of connectivity and opening up of traditionally closed factories into Internet, pose many potential risks to productivity, safety of products and processes, as well as industrial repute. It becomes paramount to perform detailed hazard and risk analysis and careful planning of mitigation mechanisms to meet security requirements applicable to the targeted industrial domains.

2 Background

One of the key building blocks of the EU-funded InSecTT project is denoted as “BB3.1: Methodologies, concepts system solutions for enabling safety and security”. It focuses on cross-layer security analysis, concepts, and system solutions including the essential steps of revealing security requirements and performing a threat analysis, defining suitable security methodologies, planning mitigation and resilience strategies (on system level), and finally looking at defining (cyber-)security concepts and solutions. The aim was to provide applied solutions as well more generic schemes, which can be used for a wider range of applications (e.g., cybersecurity in IoT system). Enabling safe system operation is of utmost priority for this task.

The task participants were an excellent mix of industrial partners (ABB, INDRA, ISS RFID, Kaitotek, LDO, LCC, Nurd, Philips Research, TietoEvry and Westermo) and academic partners (Mälardalen University (MDU), CINI, RISE, UCC, UTwente, UPM), who together provided 50+ specific requirements. These requirements can broadly be classified into the following four themes addressing cross-layer system level concepts and solutions as indicated in Fig. 1:

- Security requirements and threat analysis: High level requirement analysis and analysis of various vulnerabilities and effects of cyber-attacks on them.
- Security methodologies, concepts and system solutions: Focus on design of system/ application-level approaches and methods for identification of various security vulnerabilities as well as proactive measures for avoiding them.

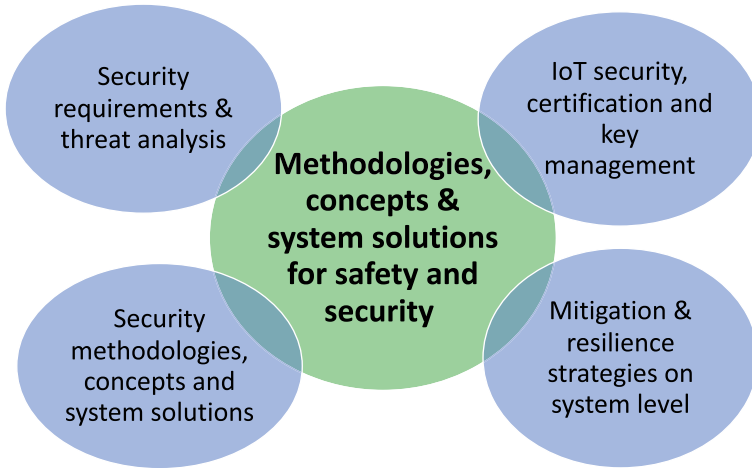


Fig. 1 Cross-layer security analysis, concepts and system solutions

- IoT security, certification and key management: Device/edge level mechanisms for enabling secure infrastructures and architectures access control and for enforcement of privacy.
- Mitigation and resilience strategies: Identification and implementation of adequate mitigation strategies to assure required levels of systems' reliance and robustness.

During the first half of the InSecTT project, a deep discussion on the individual partner contributions took place, elaborating on the various use cases, their alignments and possible synergies, resulting in the following six sub building blocks:

- (A) Access control and authentication infrastructure;
- (B) Intrusion detection systems;
- (C) IoT privacy and security mechanisms;
- (D) Secure IoT applications;
- (E) Security guidelines;
- (F) Tools and simulators.

The grouping is mainly done to find synergies and collaboration than on unification. In this chapter, we mainly present some of the research/technologies, relating to A, B, E and F, developed by a set of Swedish partners working closely in the realisation of two use cases related to smart collaborative manufacturing and secure network communications. Main critical aspects we focused on were ensuring safety/security of the industrial automation and control systems and security and privacy of the network infrastructure.

2.1 Industrial Automation and Control Systems

Industrial Automation and Control Systems (IACS) are used for operating a wide range of industrial applications, including critical infrastructure, such as power plants and clean water supplies [62]. The safe and secure operations of these systems are of utmost importance, for system owners from a business perspective, for private persons relying on reliable services and safe products, and for the society as a whole for supply of critical resources and as a basis of economical stability.

There is a trend in Industrial Control System (ICS) architectures, transitioning from a hierarchical controller-centric model as described by the Purdue Enterprise Reference Architecture (PERA) [67], towards a network-centric design strategy using a common network back-bone [2, 32]. Driving forces behind this trend are technical advances as well as novel business models and market expectations related to flexibility and customization, etc. The trend is connected to the Industry 4.0 [21, 31, 42] paradigm which is currently shaping the future of IACS, implying huge changes both from a business and technological perspective.

The named developments have a fundamental impact on the technical level for how IACS are constructed, implying increased connectivity, higher diversity and complexity, and more classes of stakeholders taking part in the system. This makes cybersecurity a major concern.

3 Selected InSecTT Technologies Targeting Security and Safety

3.1 Access Control and Authentication Infrastructure

Access control [58] is a crucial aspect of enhancing security in industrial systems [22], but it is still relatively underdeveloped compared to modern IT systems.

At the beginning of the InSecTT project, a survey was conducted among cybersecurity practitioners within industrial organizations and companies in Sweden. The study was designed to sample which techniques and principles are used related to access control, what was the foreseen challenges in the area, and, possibly, to see if the technical maturity in access control usage was related to used cybersecurity standards. The study [36] focuses on two essential cybersecurity requirements: identification and authentication control.

The purpose of performing the study was to increase knowledge on state-of-the-practice and establish a starting point for further exploration, thereby bridging the gap between the current state and the requirements of emerging systems with regards to access control.

Cybersecurity is an area where practitioners typically are reluctant in sharing potentially sensitive information, which made recruiting respondents to the study difficult. The surveying activity received enough responses to analyze and give a broad picture

of “what is out there”, but not enough to claim any statistical significance. Questionnaire was sent out to 350 organizations, which resulted in 40 respondents, some of which dropped out before completing the survey.

In the part of the survey related to challenges, the respondents could answer in a free-text form. The answers were then analyzed, resulting in seven different themes:

T1 Cost related to inclusion of secure HW components.

T2 Cost of account management.

T3 Increasing system complexity.

T4 Lack of technical support and standardization.

T5 Improper use of methods.

T6 Regulations related to open market making implemented methods ineffective.

T7 Increasing amount of cyber-attacks.

Analyzing the perceived challenges indicated by the respondents, it becomes clear that they see increasing costs related to components (theme **T1**) as well as management effort (**T2**, **T3**) in relation to identification and account management. This may be an effect of increased system complexity driven by the Industry 4.0 evolution, but also requirements related to evolving best practices. As an example, the cost for changing from shared user accounts to unique user accounts puts a significant additional burden on the account management process.

The heterogeneity of the future industrial systems is seen as a big challenge (theme **T4**), with different component manufacturers choosing incompatible technical solutions. A lack of standardization is mentioned by several respondents as an issue hampering effective account management in industrial systems.

Three themes imply direct threats to the integrity of the industrial systems. Theme **T5** indicates a lack of technical maturity leading to improper usage of the available methods. Theme **T6** indicates that “right to repair”-regulations may force manufacturers to include mechanisms which could make authentication less secure. The theme **T7** related to cybersecurity attacks on industrial systems are possibly worsened by the previous two, as the likelihood of a successful attack will increase with improperly configured systems or inherently vulnerable mechanisms. Cybersecurity attacks and information leakages in other seemingly unrelated systems may have collateral impact also on industrial systems using unique user identifications, as password re-use over several platforms is a common issue.

The perceived challenges illustrate the on-going technical shift from isolated to increasingly interconnected systems, with a resulting complexity and heterogeneity that currently used solutions cannot handle, requiring investments both related to technical components and system solutions for account management. The fear is that lack of standardization and improper usage of technical solutions may lead to more vulnerable systems, consequently increasing the likelihood of successful cybersecurity attacks.

It is clear that detailed access control used correctly will improve security characteristics of a system, but there are risks of complexity and heterogeneity making management efforts too costly and difficult. An advanced security mechanism which

is poorly configured can be worse than a very simple one used correctly. In the light of these challenges, we wanted to develop approaches and methods for handling access control in support of the emerging IACS characteristics, which are practically useful with regards to management effort and adherence available industrial standards.

Dynamic manufacturing is a well-established development of industrial automation and control systems. Manufacturing environments have, to a large extent, been optimized for high-volume production to a low per-item cost. This has led to highly specialized and optimized factories with a high complexity. These factories are prone of being difficult and expensive to retro-fit for changing demands or requirements.

Smart manufacturing [12, 47] and modular automation [30, 70] are design strategies optimized for being adaptable and customizable, in order to easily ramp up or down production, adapt to new innovations or specific customer requirements, etc. The resulting systems are dynamic manufacturing environments, which exhibits different levels of dynamicity, e.g., for modular automation, as follows.

1. Dynamic system composition—available processing modules and how they are interconnected change over time, due to changing high-level requirements.
2. Dynamic production schemes—available and active recipes describing the production workflow change on a daily basis, based on business requirements.
3. Dynamic operations—during recipe execution, different steps of the recipe-workflow are activated, implying different processing operations being executed.

In order to follow the principle of least privilege, being one of the fundamental practices within the access control theory [56], the rules of access control should adapt to the current system state. One difficult challenge arises on how to formulate access control policies to be sufficiently close to the least-privilege principle, while keeping the engineering effort related to policy formulation on a manageable level.

To investigate this challenge for dynamic manufacturing systems, we performed a study looking at five access control strategies, where three are currently being used, and two progressively aim towards a thought ideal.

All the strategies were implemented and evaluated in a simulation experiment with a number of attack scenarios, clearly showing each strategies relative effectiveness toward different attacks. As part of the study we also developed a method to automatically generate access control policies based on already available engineering data, targeting the challenge of minimizing the management effort of upholding policies close to the least-privilege principle. Details on the strategy evaluation and developed methods are available in [35].

Policy models [57] are focusing on the primitives and logic used for describing access control rules. As important are policy enforcement models, which describe the components needed, and their interactions, in order to ensure that the formulated policies are followed.

Our study [37] looked into how an access control enforcement architecture apt for dynamically changing access control scenarios of dynamic manufacturing systems could be constructed. Dynamic access control is not widely used in IACS, but it is highly relevant for the evolving system types which are inherently dynamic. Four different enforcement architecture models are investigated and evaluated based on three

important metrics: resource server workload, network, load, and flexibility. The two most promising models required policy delegation mechanisms using access tokens. Four different variations on how to encode policy decisions into access tokens are provided and discussed with regards to available support in the Open Process Communication Unified Automation (OPC UA) standard [49]. Finally an implementation is performed, using a combination of one of enforcement models and delegation mechanisms, e.g., detailing the authorization protocol, access token encoding logic, and policy decision logic in the resource server. Separate studies [38, 52] are performed evaluating quality metrics on different aspects of the proposed OPC UA authorization protocol.

3.2 *Intrusion Detection Systems*

One of the biggest challenges in the network security research area is identifying malicious activities on time and mitigating them promptly. The process of analyzing network traffic to identify signs of malicious activity is called intrusion detection [41] and a system that automates this process is called the Intrusion Detection System (IDS) [8]. There are two common methodologies that IDSs use to identify threats: signature-based and anomaly-based [59]. A signature-based IDS monitors network packets and searches for patterns that correspond to known network attack types. Anomaly-based IDS learns the general behavior of normal network traffic and raises an alarm when significant deviations are detected.

In recent years, Machine Learning (ML) has become a popular and effective method for developing new anomaly-based IDS [9, 17, 46, 60, 65].

ML is the part of AI where algorithms learn patterns from datasets without explicit instructions [55]. It can be divided into the following areas:

- Supervised Learning (SL): algorithms within the SL category use input-output pairs to learn a function that maps from inputs to outputs;
- Unsupervised Learning (UL): algorithms within the UL category learn patterns within the input data without any output information given in the training phase;
- Reinforcement Learning (RL): algorithms within the RL category learn by trial and error. A specific “reward” or “punishment” is given depending on their actions and consequences.

Various ML algorithms were applied to existing datasets either to separate normal traffic for the malicious one (binary classification problems) or to detect specific attack types (multiclass classification problems). Buczak et al. [10] did a focused literature survey of ML methods used in IDSs and recognized some of the most commonly used methods such as Random Forest (RF), Decision Trees, density-based clustering algorithms (e.g., DBSCAN), Support Vector Machine (SVM), Artificial Neural Networks (ANN), Naive Bayes (NB), association rules, etc. Many studies in this area analyze the accuracy of different ML algorithms on different benchmark

datasets. Revathi et al. [53] presented an evaluation of supervised ML algorithms (RF, J48, SVM, Classification and regression trees and NB) for multiclass classification on one dataset (NSL-KDD) and derived the conclusion that RF has the highest accuracy compared to all other algorithms. Abedin et al. [4] worked on the same problem by applying NB, J48, NBTree, Multilayer Perceptron (MLP), and RF and their findings were that J48 and RF had the best performance. Tuan et al. [66] evaluated SVM, ANN, NB, Decision Tree, and unsupervised ML on the UNBS-NB 15 and KDD99 datasets. This paper considered only the Distributed Denial of Service (DDoS) attacks and unsupervised ML was the best at differentiating between DDoS and normal network traffic, but it was not specified which unsupervised ML algorithms were used. There are several papers that evaluate a single ML algorithm on one or more benchmark datasets, such as different types of neural networks [9, 24, 28, 54, 68], RF [16], SVM [50], K-means [29], etc.

Most of existing works focus on evaluating ML algorithms on a single dataset or on evaluating single ML algorithm on multiple datasets. Also, most of the papers focused only on binary or multiclass classification.

Our research efforts in the EU-funded InSecTT project with respect to intrusion detection had the following overall objectives:

- Apply multiple methods on multiple datasets and compare their performances.
- Extend the SOTA anomaly classification problem.
- Extend the SOTA and SOP for the realization of federated learning in industrial contexts with resource constraints.
- Study the pros and cons of existing datasets and design new more realistic datasets and simulators to suit the manufacturing and networking domains.

In this section, we briefly present our results on the first three objectives, while the fourth one is addressed in Sect. 3.3.

As previously mentioned, AI is used as an IDS by many authors on different datasets, but the test results are usually limited in terms of algorithms, datasets or problem that was solved (anomaly detection or anomaly classification).

In [39], we made a more complete comparison including a total of 5 supervised learning algorithms (ANN, SVM, KNN, LDA, RF) and 3 unsupervised learning algorithms (K-means, mean-shift and DBSCAN) tested for anomaly detection and anomaly classification on 4 datasets (KDD99, NSL-KDD, UNSW-NB15 and CIC-IDS-2017). The results showed that RF, KNN, and SVM were the algorithms that performed the best in terms of accuracy. If in addition training and testing time are considered, RF emerges as the best option.

Now we know which ML method is more suitable for the desired problem. However, we believe that more layers of security are needed. For this reason, we proposed a Federated Learning (FL) framework that increases the security of data [44]. The framework is used on different clients (i.e., routers) that receive packages. On each client, a different RF is implemented and trained on the edge. RF is selected because of the various reasons that are proved by the experiments presented in the previous paragraph: the best performance and a reasonable time to be implemented in a real-time scenario. On top of that, RF is the algorithm with a high degree of explainability.

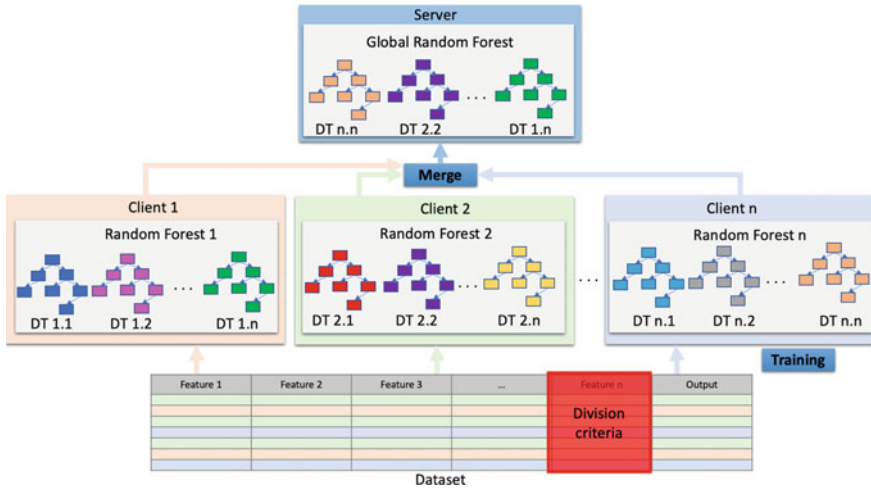


Fig. 2 Architecture of the Federated Learning Framework based on RF [44]

After training the models on different clients, these are sent to the server where a new super RF is created as a combination of them. Then, this final model is sent back to the clients to perform anomaly detection and classification. An overview of the framework is presented in Fig. 2. The results of the experiments clearly showed that combining the different RF algorithms is beneficial for the algorithm performance, increasing the detection rate. With this framework, we avoid distributing the data through the network to create the model in the server with all the data available, which protects the data in a sense, since after training the models, these data can be deleted and the risk of an intruder accessing the data is eliminated.

Subsequently, we proposed a second framework in which data, in this case, could be shared between different entities, because the data are encoded [40]. This framework uses an autoencoder (type of ANN [19]) to encode the data used by the ML algorithm to detect or classify anomalies. The novelty lies in the use of an optimization algorithm called Differential Evolution (DE) [61] to train the autoencoder. It uses two objectives to find the best model: (a) the error of the autoencoder when trying to decode the encoded features, and (b) the accuracy of a ML algorithm. The results show that for the algorithms that obtained the best performance in our first comparison [39], the performance is reduced by a small amount. On the other hand for the algorithms that could not perform high-quality anomaly detection or classification, performance increased significantly. Furthermore, the method is compared to the principal component analysis [3] obtaining better results for anomaly detection. An overview of the presented framework is given in Fig. 3. With this method, we can send the encoded data through the network with the certainty that no intruder will be able to decode the data since the model is needed for decoding. An additional benefit of this method is that by using autoencoder, we are able to recover the original data, which is not possible by the majority of the well-known encoding methods.

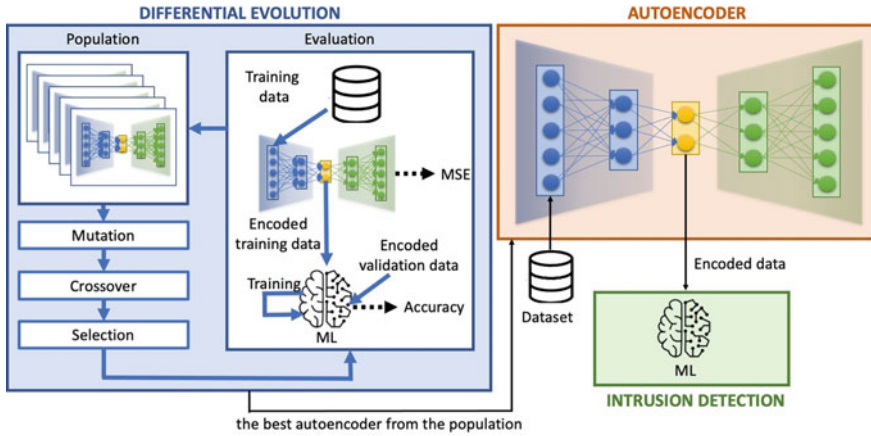


Fig. 3 Proposed framework for feature encoding [40]

3.3 Tools, Simulators and Datasets

The majority of experiments in the area of intrusion detection were conducted using one or more benchmark datasets [18]. Some of the well-known IDS datasets are: KDD99, NSL-KDD, UNSW-NB 15, CIDDS-001, CICIDS2017, CSE-CIC-IDS2018, etc. All of those datasets consist of a combination of normal and malicious traffic. Data about network packets were preprocessed to create the features, and every entry was labeled as normal activity, or as some type of network attack.

The impossibility of testing in operational industrial systems due to security issues leaves a gap in the environment for testing and measuring the impact of cyber threats and the development of defense systems. There are some real ICS testbeds in the world, such as the national SCADA testbed or Swat, a small-scale water treatment center. However, these testbeds are not accessible by all researchers. Building such an environment is also a pretty time-consuming and expensive process. Besides, scientists should deal with a vast range of unrelated technical problems that needs HW knowledge. These barriers have led many security researchers, especially those who want to use AI and ML methods for attack detection, to use available datasets for their experiments. Intrusion detection using prepared datasets prevents researchers to define customized test conditions or change the type of attack on the industrial systems. Therefore, a tool to create a virtual industrial control system cable to perform cybersecurity research will be a great asset for researchers.

3.3.1 ICSSIM–A Framework for Building Industrial Control Systems Security Testbeds

The importance of studying cyberattacks, testing ICSs, and creating defense mechanisms cannot be overstated. However, due to safety concerns, conducting these

studies on operational ICSs is often not allowed. One solution is to use a small-scale pilot ICS as a test environment, but these testbeds are not widely accessible, can be time-consuming and expensive to build, and require hardware knowledge to overcome various technical issues [20], as mentioned in Sect. 3.3. As a result, many security researchers, particularly those using machine learning methods for attack detection, resort to using existing datasets for their experiments. But using these datasets limits the ability to customize test conditions or change the attack type on industrial systems. Thus, a tool to create a test environment for ICS security would be valuable for researchers and practitioners alike.

After thorough analysis, we have reviewed numerous publications introducing various testbeds and simulation tools for ICS [6, 11]. Based on this review, we have compiled a list of essential features for a testbed. As a significant contribution to the EU-funded InSecTT project, we introduce ICSSIM [13], a framework designed to facilitate the creation of virtual testbeds for in-depth exploration of diverse cyber threats and network attacks within ICSs. ICSSIM has a set of base classes for modeling ICS components and communication. Notably, this framework allows deploying its simulated components onto hardware such as Raspberry Pi and containerized platforms like Docker. Furthermore, ICSSIM offers comprehensive support for physical process modeling, incorporating both software and hardware-in-the-loop simulation techniques.

The primary objective of ICSSIM is to expedite the development of ICS components, resulting in the creation of versatile, reproducible, and cost-effective ICS testbeds that capture real-world details. The efficacy of ICSSIM becomes readily apparent through the practical demonstration of its capabilities. In this context, we have leveraged ICSSIM to construct a testbed, showcasing its versatility in simulating various cyberattacks. We have implemented several attack scenarios within this environment, including Man in the Middle attack (MITM), DDoS attack, reconnaissance attack, false data injection using MITM, replay attack, and command injection by considering various attack scenarios.

We also published the ‘ICS-Flow’ dataset [15], created through sample security experiments in this environment. We presented the ICS-Flow dataset for ML-based IDS evaluation through supervised and unsupervised methods. The dataset was generated using the ICSSIM simulator, which emulates the ICS of a bottle-filling factory in a ‘Hardware in the Loop’ simulation and utilizes realistic industrial protocols such as Modbus. Network data and process state variable logs were recorded during normal operations and during four common cyberattacks. The ICS-Flow dataset includes raw network data, network flow data, process variable logs, and attack logs for ML-based anomalous record detection and sequence detection. We demonstrate the effectiveness of the ICS-Flow dataset by applying decision tree, random forest, and artificial neural network models for anomaly and attack detection, showing that the dataset can be effectively utilized for training ML models for intrusion detection.

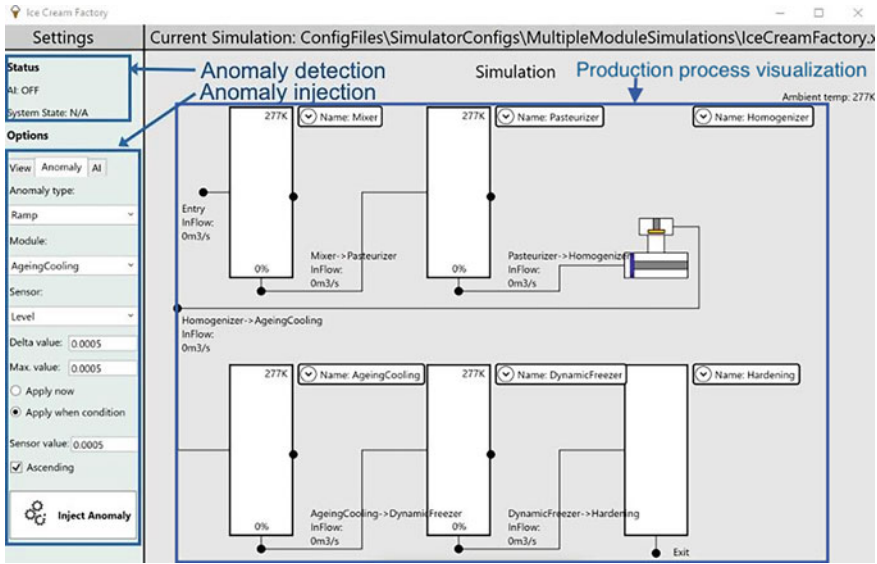


Fig. 4 User interface of simulation environment for modular ice cream factory example, including: production process visualisation, anomaly injection and anomaly detection [33]

3.3.2 Modular Ice Cream Factory Simulator and Anomaly Injections

ABB and MDU developed a simulation environment to represent a modular manufacturing system [33, 34]. This environment is composed of simulated sensors and actuators and was built using the modular automation design strategy [30, 48, 69]. It allows easy configuration and combination of simple modules into complex production processes. Sensor and actuator signals are exchanged with controllers using the Message Queue Telemetry Transfer (MQTT) protocol [1]. Synchronization of the overall process is performed using high-level recipe orchestration, utilizing OPC UA [49] client/server communication. The simulation environment is presented in detail in [34]. Visualisation of the current state of the simulated process is provided as a simple Graphical User Interface (GUI) that contains visual representations of modules, their interconnections, and current values of the parameters. Additionally, there is functionality that enables users to manually inject different types of anomalies into analog sensors during the production process.

An example of the use of the simulation environment is a modular ice cream factory, in which, as shown in Fig. 4, the simulation engine is configured to simulate the behavior of six separate modules: a mixer, a pasteurizer, a homogenizer, an ageing and cooling module, a dynamic freezer, and a packaging module.

This setup was used to create an open dataset named Modular ice cream factory Dataset on Anomalies in Sensors (MIDAS) [45] that contains various anomalies in analog sensors and can be used for ML research in modular manufacturing systems. The anomalies were injected using a script that automatizes the anomaly injection pro-

cess, injecting anomalies to modify values of different sensors during different stages of the simulated process, with different values of parameters for a specific anomaly that is injected. The anomaly injection occurs in a randomly selected moment, either with an increasing or a decreasing trend, when the sensor value changes. The dataset contains a separate CSV file for each of 1000 runs, where 258 runs represent normal behaviour and 742 runs contains anomalies, with three different types of anomalies (Freeze, Step, and Ramp). It has 36,124,859 instances, where 49.67% instances that represent normal behaviour, and 50.33% instances contain anomalies. The distribution of instances between normal behaviour and each anomaly were Normal (50%), Freeze (15%), Ramp (17%), and Step (18%).

The generated dataset was used to evaluate different supervised ML algorithms (Logistic Regression, Decision Tree, Random Forest, and MLP, as well as a time-series ML algorithm (Long-Short Term Memory–LSTM)) for two different problems: anomaly detection and anomaly classification. Experiments showed that using the temporal information into LSTM network performed better than the non-temporal ML algorithms [43]. We decided to integrate the LSTM model into the demonstrator to provide reliable anomaly detection functionality.

3.3.3 Virtualization and Emulation of Industrial Network Topology–Westermo

Many of Westermo’s products—switches and routers for harsh industrial settings—run the Westermo operating system (WeOS). In order to verify that WeOS is operating as expected after code changes and extensions, a significant effort has been invested in automated software testing of the devices [63].

In the ideal case, software has a low fail rate and high reliability once it is developed, see blue curve in Fig. 5. However, in reality, there are typically updates during the useful life of the software (red curve) [5, 51]. There is thus a strong need for quality assurance, software testing and preferably automated software testing.

When testing embedded systems, at some point one has to run it on physical hardware to verify timing and other non-functional characteristics related to hardware and the software-hardware integration. For this purpose, several physical test systems have been constructed at Westermo. There is a significant amount of physical test equipment, it weighs more than a tonne, requires redundant air conditioning and fills several large rooms. Since some years, many of the pure software parts of WeOS can run in a virtual environment, which enables testing of significant parts of WeOS without any hardware. There are thus a number of test systems constructed where the physical devices have been replaced with purely virtual digital twins, QEMU has been an important enabler.

A challenge that may occur when developing new software is to get access to hardware that supports it, in particular when new hardware models are developed in parallel with the software development. In Fig. 6 the timing of one software development sub-project from Westermo is illustrated. In blue, we see the trend of failing tests when running WeOS on virtual test systems, and in red the same trend on

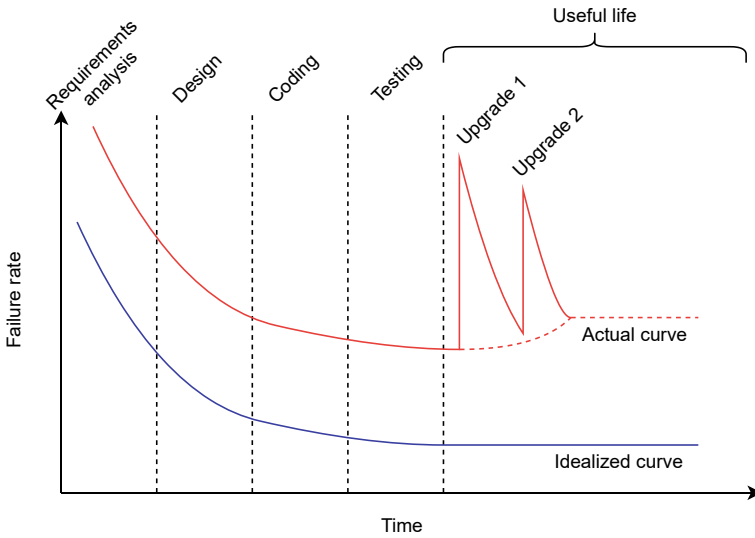


Fig. 5 Typical curve for software reliability (image from [5], used with permission)

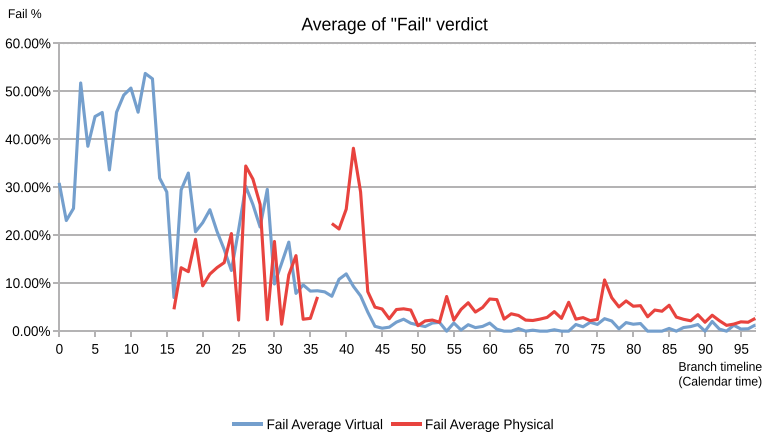


Fig. 6 Ratio of tests that fail, from an experimental development branch, over more than three months. Blue curve illustrates virtual test systems, and red physical test systems (image from [5], used with permission)

test systems with physical devices. There are two relevant observations to be made. First, testing on physical hardware is postponed, in this case only by two weeks— anecdotally, we know that this can be much more. Second, some bugs in the software are only visible on physical hardware (red peak at about 40 days). For this reason, we wished to explore hybrid test systems, where some if not most parts of a test system were virtualized whereas one or a few devices were physical. An overview of how this could be implemented is illustrated in Fig. 7.

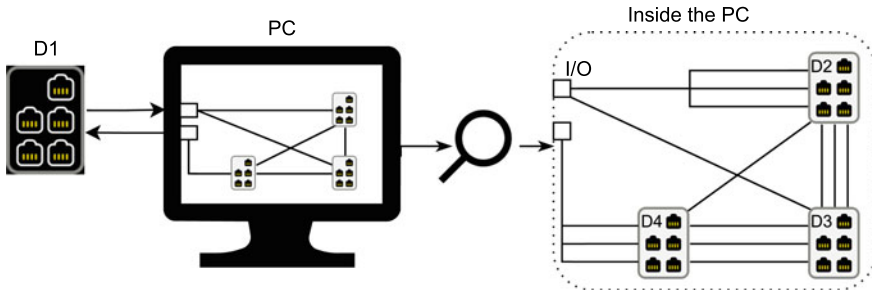


Fig. 7 Example of how a hybrid test system could be created (image from [5], used with permission)

During the InSecTT project, such hybrid test systems were explored with different activities followed by a standardisation and refinement phase. The results indicate that testing software modifications could start earlier, that certain software issues could be reproduced more reliably, when compared with testing in a purely virtual environment, in particular with respect to the reality gap and timing issues. Furthermore, some of the challenges with pure hardware environments are reduced (e.g., the need for hardware is reduced, which has become a problem due to the chip crisis). On the other hand, one can expect test systems with only hardware to more reliably mimic customer settings, therefore, hybrid systems can be expected to be less reliable. Furthermore, industry practitioners expressed concern that virtual-only test systems may lead to false positives when testing, but how more reliable hybrid test systems are when compared to virtual test systems remains to be explored [5].

A second technology Westermo have worked on during the EU-funded InSecTT project, is to run an AI inside a container in a Westermo router, see Fig. 8. For this work, Westermo developed a container feature in some WeOS versions on some hardware products (see A in Fig. 8). Thanks to InSecTT project activities, Westermo now has a rather mature container support based on cgroups which is a common container technology (though not as well known as Docker). Towards the end of InSecTT, we started implementing an AI for our containers (B in Fig. 8). The first step was to explore how well this worked in practice, and the limitations of the resources in the hardware. Preliminary results were mixed, hardware restrictions were acceptable, and many but not all anomalies were detected [23]. In future work, we could explore distributed or federated AI (C), or if a fog or cloud-based AI is better for this industrial context (D).

A third track of work from Westermo in InSecTT is collecting a realistic dataset for supporting AI research. To achieve this, Westermo teamed up with partners (MDU, RISE and TietoEVRY), to define and implement a data collection scenario. In our previous experience, when releasing a dataset [64] from the parallel research project AIDOaRt, we set up information security risk workshops. This practice was also used in InSecTT, and the network traffic dataset has now been published for the general public on GitHub [64].

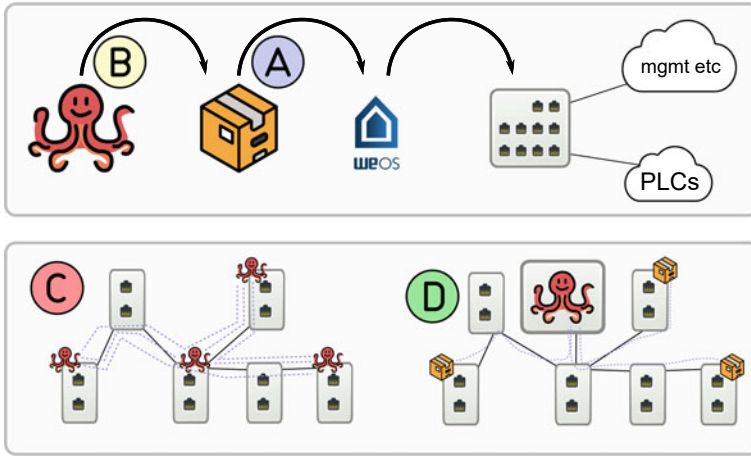


Fig. 8 Overview of AI in container implementation (top row), and possible extensions (bottom)

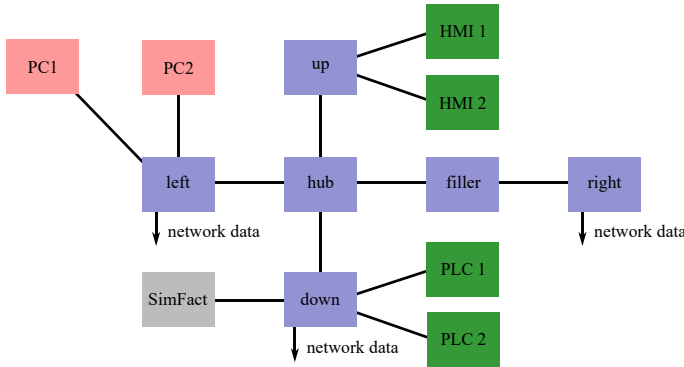


Fig. 9 Network topology used during data collection [64]

For the data collection, a test system with six Westermo devices, five Raspberry Pi devices and two laptops has been built. On the Raspberry Pi devices, we ran ICSSIM (see Sect. 3.3.1), two had the role of HMIs, two were PLCs, and the fifth simulated the physical world, see Fig. 9. Using port forwarding we redirected all network traffic from three Westermo devices to a PC where it was collected with tcpdump. During the data collection we conducted seven types of changes, misconfigurations or attacks, intermixed with periods where the network was not disturbed. The disturbances were:

1. Misconfigured IP: a random WeOS device has its IP address changed. Instead of a correct one, like 192.168.0.1, we would swap the second and third octet, into 192.0.168.1. After some time, the address would be corrected.
2. Duplicated IP: a random WeOS device is given the same IP address as another WeOS device. After some time, the address is corrected.

3. Acceptable SSH traffic: Using SSH with a correct username and password, we log in and check the contents of a log file, and log out.
4. Password guessing: By using usernames and passwords based on the Mirai Malware¹ we generate many parallel attempts to log in over SSH.
5. Port scan: Using nmap, we scan the ports of one or several devices in the network.
6. MITM: Using the attack toolbox of ICSSIM, we launch a MITM attack and rewrite values in modbus packets.

By including both human misconfiguration, attacks, as well as benign disturbances, we aim at supporting work on distributed AI and anomaly detection with this dataset. We speculate that it can also support work on where an anomaly detection system ought to be placed in the network (e.g., close to PLCs).

The Westermo network traffic dataset was used to evaluate ML algorithms with centralized, local, and federated approach for anomaly detection in network data [14]. We used different supervised ML algorithms in local and centralized approach, including: Logistic Regression, SVM, ANN, KNN, Decision Tree and RF. RF and ANN exhibited superior performance and were implemented in a federated setup. The experiments showed that federated version outperforms the local models, and achieves comparable or even superior results compared to the centralized model, while ensuring data privacy and confidentiality of sensitive information.

3.4 Safety and Security Analysis for AGV Platooning

There is an increasing trend of Automated Guided Vehicles (AGVs) and platooning. AGVs are an integral part of the Industry 4.0 [26]. Their platooning tends to improve overall safety, security and operational efficiency of production site. The published studies on platooning focus mainly on the design of technical solutions of automotive domain, but not considered the AGV platooning in production sites and Industry 4.0. We presented a platooning strategy which not only provides a means to control overall traffic flow at production site and reduce resource usage but also manage transportation risks in a dynamic manner. However, the automation, digitalisation and connectivity of AGVs with each other or with the infrastructure significantly pose the safety and security issues [25, 26]. A security attack or a single failure in one AGV could lead to unsafe behaviour of whole platoon that can potentially harm humans (injuries or even deaths) or create damages to machines, property or the environment. The safety-critical systems can only be regarded as safe if they are also secure. The literature highlights a dearth of comprehensive research on different aspects of vehicle platooning including safety and security analyses [7]. There is a need for comprehensive studies to deal with the situations such as joining and leaving platoon in production sites, connectivity with fog and cloud servers, system or component failures, security attacks, and influencing environmental factors.

¹ <https://github.com/jgamblin/Mirai-Source-Code>.

Established safety and security analyses methods such as the hazard and threat analyses are performed during design and development phase by using the Hazard and Operability (HAZOP) and Threat and Operability (THROP) techniques, respectively. We considered the interactions of collaborative autonomous systems with one another, and to the fog controller that, in-turn, interact with the cloud infrastructure. To perform the HAZOP and THROP analyses we establish a list of guide words (e.g., no/not, false/fake, incorrect, increase/exceed, unavailable, unintended, exploit, other than, etc.) and systems parameters/functions, such as sensors, actuators, communication and connectivity (e.g., WiFi, 4G/5G, IoT devices, fog) and type of messages (e.g., request, response, and command). As the collaborative autonomous systems underlie the need for dynamic risk management, the data is gathered to monitor systems operations, identify unexpected or incorrect behaviour, evaluate the potential implications and trigger control actions to resolve them.

We presented the overall approach for a fault- and threat tolerant platooning for materials transportation in production environments with detailed analysis in [27].

4 Novelty and Applicability of Proposed Technologies

Adaptable access control rule inference and enforcement are based on industrial standards. The technologies used for the enforcement architecture uses available standardized components, e.g., OPC UA for the communication stack, and JWT for access tokens, which makes the suggested solution applicable in any domain utilizing these standards. The publications both present novel material, and some of the enforcement architecture mechanisms developed are currently being evaluated for potential IP protection.

We have proposed a structured approach for generation of datasets on sensor anomalies in manufacturing context (both manual and automatic injection of anomalies supported). The architecture of the use case and the various simulation modules are following the modular automation principles, thus allowing easy evolution and adaption of the systems and related validation efforts. This also helps in quick security analysis through focused testing efforts. Our ML algorithms comparisons are based on a larger set of algorithms applied on multiple well-known anomaly datasets.

The ICSSIM is built based on container technology, which means that ICS components run on isolated operating system kernels. Moreover, simulated containerized components such as PLCs, HMIs, or HW in a loop (HIL) processes could be run on simulation engines such as GNS3 or emulation environments such as Docker containers, or they could totally be replaced with physical entities. It also has a stub for SW simulation of HIL to simulate the control process. Moreover, ICSSIM has interfaces to communicate with the HW through the file to use real HW for the process. ICSSIM can be used for simulation of any ICS.

This creation of a virtual network topology by Westermo is enhanced by addition of their test framework that can be used to test any functionality in an industrial network topology, not limited to ICSs. This can also run automatically with follow

up of test results. A true first step of a digital twin (DT) that can be valuable for ensuring network security in diverse manufacturing setups is to enable even online detections and mitigations.

5 Conclusions and Future Perspectives

In this chapter we have presented some of the research outputs and technology contributions realised as part of the EU-funded InSecTT project. The presented works show fruits of extended industry-academia collaboration to solve important challenges related to safety and security in manufacturing environments. The industrial partners are keen on exploiting the above results as evidenced by patent applications and tools being inducted. We are currently continuing the work in several directions such as studies to include wider coverage of ML models, integrating federated learning model into resource-constrained network switches and further demonstrations in other contexts such as smart-cities.

References

1. MQTT Version 5.0. OASIS Standard, March 2019. Edited by Andrew Banks, Ed Briggs, Ken Borgendale, and Rahul Gupta
2. O-PAS Standard, Version 2.0: Part 1-Technical Architecture Overview. Open Group Preliminary Standard (P201-1), The Open Group (Feb. 2020)
3. Abdi, H., Williams, L.J.: Principal component analysis. In: Wiley Interdisciplinary Reviews: Computational Statistics, vol. 2, no. 4, pp. 433–459 (2010)
4. Abedin, M., Alam Siddiquee, K.N.E., Bhuyan, M.S., Karim, R., Hossain, M.S., Andersson, K., et al.: Performance analysis of anomaly based network intrusion detection systems. In: 43rd IEEE Conference on Local Computer Networks Workshops (LCN Workshops), Chicago, 1–4 Oct. 2018, pp. 1–7. IEEE Computer Society (2018)
5. Alhasan, W.: Evaluating Challenges, Benefits, and Dependability of Virtual and Physical Testing of Embedded Systems Software. Master's thesis, Mälardalen University (2022)
6. Ani, U.P.D., Watson, J.M., Green, B., Craggs, B., Nurse, J.R.C.: Design considerations for building credible security testbeds: perspectives from industrial control system use cases. *J. Cyber Secur. Technol.* **5**(2) (2021)
7. Axelsson, J.: Safety in vehicle platooning: a systematic literature review. *IEEE Trans. Intell. Transp. Syst.* **18**(5), 1033–1045 (2017)
8. Bace, R., Mell, P.: Intrusion detection systems. National Institute of Standards and Technology (NIST), Technical Report 800-31 (2001)
9. Behera, S., Pradhan, A., Dash, R.: Deep neural network architecture for anomaly based intrusion detection system. In: 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), pp. 270–274. IEEE (2018)
10. Buczak, A.L., Guven, E.: A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Commun. Surv. Tutor.* **18**(2) (2015)
11. Conti, M., Donadel, D., Turrin, F.: A survey on industrial control system testbeds and datasets for security research (2021). [arXiv:2102.05631](https://arxiv.org/abs/2102.05631)
12. Davis, J., Edgar, T., Porter, J., Bernaden, J., Sarli, M.: Smart manufacturing, manufacturing intelligence and demand-dynamic performance. *Comput. Chem. Eng.* **47**, 145–156 (2012)

13. Dehlaghi-Ghadim, A., Balador, A., Moghadam, M.H., Hansson, H., Conti, M.: Icssim-a framework for building industrial control systems security testbeds. *Comput. Ind.* **148**, 103906 (2023)
14. Dehlaghi-Ghadim, A., Markovic, T., Leon, M., Söderman, D., Strandberg, P.E.: Federated learning for network anomaly detection in a distributed industrial environment. In: 2023 22nd IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE (2023)
15. Dehlaghi-Ghadim, A., Moghadam, M.H., Balador, A., Hansson, H.: Anomaly detection dataset for industrial control systems (2023). [arXiv:2305.09678](https://arxiv.org/abs/2305.09678)
16. Farnaaz, N., Jabbar, M.A.: Random forest modeling for network intrusion detection system. *Proc. Comput. Sci.* **89**, 213–217 (2016)
17. Fu, Y., Lou, F., Meng, F., Tian, Z., Zhang, H., Jiang, F.: An intelligent network attack detection method based on rnn. In: 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC), pp. 483–489. IEEE (2018)
18. Ghurab, M., Gaphari, G., Alshami, F., Alshamy, R., Othman, S.: A detailed analysis of benchmark datasets for network intrusion detection system. *Asian J. Res. Comput. Sci.* **7**(4), 14–33 (2021)
19. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*. MIT Press (2016). <http://www.deeplearningbook.org>
20. Green, B., Lee, A., Antrobus, R., Roedig, U., Hutchison, D., Rashid, A.: Pains, gains and PLCs: ten lessons from building an industrial control systems testbed for security research. In: 10th {USENIX} Workshop on Cyber Security Experimentation and Test {CSET}, vol. 17 (2017)
21. Hermann, M., Pentek, T., Otto, B.: Design principles for industrie 4.0 scenarios. In: Proceedings of the Hawaii International Conference on System Sciences, vol. 2016, pp. 3928–3937. IEEE (Mar. 2016)
22. IEC 62443 security for industrial automation and control systems. Standard, International Electrotechnical Commission, Geneva, CH, 2009-2018
23. Ingletto, G., Lidholm, P.: Anomaly Detection for Network Traffic in a Resource Constrained Environment. Master's thesis, Mälardalen University (2023)
24. Ingre, B., Yadav, A.: Performance analysis of NSL-KDD dataset using ANN. In 2015 International Conference on Signal Processing and Communication Engineering Systems, pp. 92–96. IEEE (2015)
25. Jaradat, O., Sljivo, I., Habli, I., Hawkins, R.: Challenges of safety assurance for industry 4.0. In: 13th European Dependable Computing Conference, EDCC Geneva, Switzerland (2017)
26. Javed, M.A., Muram, F.U., Hansson, H., Punnekkat, S., Thane, H.: Towards dynamic safety assurance for Industry 4.0. *J. Syst. Archit.* **114**, 101914 (2021)
27. Javed, M.A., Muram, F.U., Hansson, H., Punnekkat, S., Hansson, H.: Safe and secure platooning of automated guided vehicles in industry 4.0. *J. Syst. Archit.* **121**, 102309 (2021)
28. Kim, J., Kim, J., Kim, H., Shim, M., Choi, E.: CNN-based network intrusion detection against denial-of-service attacks. *Electronics* **9**(6), 916 (2020)
29. Kumar, V., Chauhan, H., Panwar, D.: K-means clustering approach to analyze NSL-KDD intrusion detection dataset. *Int. J. Soft Comput. Eng. (IJSCE)* ISSN, 2231–2307 (2013)
30. Ladiges, J., et al.: Integration of modular process units into process control systems. *IEEE Trans. Ind. Appl.* **54**(2), 1870–1880 (2018)
31. Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., Hoffmann, M.: Industry 4.0. *Bus. Inf. Syst. Eng.* **6**(4), 239–242 (2014)
32. Leander, B., Johansson, B., Lindström, T., Holmström, O., Nolte, T., Papadopoulos, A.V.: Dependability and security aspects of network-centric control. In: 28th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE (2023)
33. Leander, B., Markovic, T., Leon, M.: Enhanced simulation environment to support research in modular manufacturing systems. In: IECON, pp. 1–6. IEEE (2023)
34. Leander, B., Marković, T., Čaušević, A., Lindström, T., Hansson, H., Punnekkat, S.: Simulation environment for modular automation systems. In: IECON (2022)
35. Leander, B., Čaušević, A., Hansson, H., Lindström, T.: Toward an ideal access control strategy for industry 4.0 manufacturing systems. *IEEE Access* **9** (2021)

36. Leander, B., Čaušević, A., Lindström, T., Hansson, H.: A questionnaire study on the use of access control in industrial systems. In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA) (2021)
37. Leander, B., Čaušević, A., Lindström, T., Hansson, H.: Access control enforcement architectures for dynamic manufacturing systems. In: 2023 IEEE 20th International Conference on Software Architecture (ICSA), pp. 82–92 (2023)
38. Leander, B., Čaušević, A., Lindström, T., Hansson, H.: Evaluation of an OPC UA-based access control enforcement architecture. In: ESORICS 2023 International Workshops: CyberICPS (2023)
39. Leon, M., Markovic, T., Punnekkat, S.: Comparative evaluation of machine learning algorithms for network intrusion detection and attack classification. In: 2022 International Joint Conference on Neural Networks (IJCNN), pp. 01–08. IEEE (2022)
40. Leon, M., Markovic, T., Punnekkat, S.: Feature encoding with autoencoder and differential evolution for network intrusion detection using machine learning. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion (2022)
41. Liao, H.-J., Richard Lin, C.-H., Lin, Y.-C., Tung, K.-Y.: Intrusion detection system: a comprehensive review. *J. Netw. Comput. Appl.* **36**(1), 16–24 (2013)
42. Lu, Y.: Industry 4.0: a survey on technologies, applications and open research issues. *J. Ind. Inf. Integr.* **6**, 1–10 (2017)
43. Markovic, T., Dehlaghi-Ghadim, A., Leon, M., Balador, A., Punnekkat, S.: Time-series anomaly detection and classification with long short-term memory network on industrial manufacturing systems. In: 18th Conference on Computer Science and Intelligence Systems FedCSIS. IEEE (2023)
44. Markovic, T., Leon, M., Buffoni, D., Punnekkat, S.: Random forest based on federated learning for intrusion detection. In: Artificial Intelligence Applications and Innovations: 18th IFIP WG 12.5 International Conference, AIAI 2022, Hersonissos, Crete, Greece, June 17–20, 2022, Proceedings, Part I, pp. 132–144. Springer (2022)
45. Markovic, T., Leon, M., Leander, B., Punnekkat, S.: A modular ice cream factory dataset on anomalies in sensors to support machine learning research in manufacturing systems. *IEEE Access* **11**, 29744–29758 (2023)
46. Mazhar Rathore, M., Ahmad, A., Paul, A.: Real time intrusion detection system for ultra-high-speed big data environments. *J. Supercomput.* **72**(9) (2016)
47. Mittal, S., Khan, M.A., Wuest, T.: Smart manufacturing: characteristics and technologies. In: Harik, R., Rivest, L., Bernard, A., Eynard, B., Bouras, A. (eds.) *Product Lifecycle Management for Digital Transformation of Industries*, pp. 539–548, Cham, 2016. Springer International Publishing (2016)
48. NAMUR Working Group 1.12. NE 148 Automation Requirements relating to Modularisation of Process Plants. NAMUR-recommendation (2013)
49. OPC unified architecture: Standard, IEC, Geneva, CH (2016)
50. Pervez, M.S., Farid, D.M.: Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs. In: International Conference on Software, Knowledge, Information Management and Applications (SKIMA), pp. 1–6. IEEE (2014)
51. Quyoum, A., Dar, M.-U.-D., Quadri, S.M.K.: Improving software reliability using software engineering approach-a review. *Int. J. Comput. Appl.* **10**(5), 41–47 (2010)
52. Radonjić, I., Bašić, E., Leander, B., Marković, T.: An authorization service supporting dynamic access control in manufacturing systems. In: IEEE 9th World Forum on Internet of Things (2023)
53. Revathi, S., Malathi, A.: A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. *Int. J. Eng. Res. Technol. (IJERT)* **2**(12), 1848–1853 (2013)
54. Roy, B., Cheung, H.: A deep learning approach for intrusion detection in internet of things using bi-directional long short-term memory recurrent neural network. In: International Telecommunication Networks and Applications Conference (ITNAC), pp. 1–6. IEEE (2018)

55. Russell, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*. Pearson Education Limited, Malaysia (2016)
56. Saltzer, J., Schroeder, M.: The protection of information in computer systems. *Proc. IEEE* **63**, 1278–1308 (1975)
57. Sandhu, R., Ranganathan, K., Zhang, X.: Secure information sharing enabled by trusted computing and PEI models. In: *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, ASIACCS'06*, vol. 2006, pp. 2–12 (2006)
58. Sandhu, R.S., Samarati, P.: Access control: principle and practice. *IEEE Commun. Mag.* **32**(9), 40–48 (1994)
59. Scarfone, K., Mell, P., et al.: *Guide to intrusion detection and prevention systems (idps)*. NIST Special Publication, (800-94) (2007)
60. Shrivastava, A.K., Dewangan, A.K.: An ensemble model for classification of attacks with feature selection based on KDD99 and NSL-KDD data set. *Int. J. Comput. Appl.* **99**(15), 8–13 (2014)
61. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Global Optim.* **11**(4) (1997)
62. Stouffer, K., Pillitteri, V., Lightman, S., Abrams, M., Hahn, A.: *Guide to Industrial Control Systems (ICS) Security* NIST Special Publication 800-82 Revision 2. NIST Special Publication 800-82 rev 2, pp. 1–157 (2015)
63. Strandberg, P.E.: *Automated System-Level Software Testing of Industrial Networked Embedded Systems*. Ph.D. thesis, Mälardalen University (2021)
64. Strandberg, P.E., Söderman, D., Dehlaghi-Ghadim, A., Leon, M., Markovic, T., Punnekkat, S., Moghadam, M.H., Buffoni, D.: The Westermo network traffic data set. *Data in Brief* **50**, 109512 (2023)
65. Survey, A., Wang, S., Fernando Balarezo, J., Kandeepan, S., Al-Hourani, A., Gomez Chavez, K., Rubinstein, B.: Machine learning in network anomaly detection. *IEEE Access* **9**, 152379–152396 (2021)
66. Tuan, T.A., Long, H.V., Son, L.H., Kumar, R., Priyadarshini, I., Son, N.T.K.: Performance evaluation of Botnet DDoS attack detection using machine learning. *Evolut. Intell.* **13**(2), 283–294 (2020)
67. Williams, T.J.: The Purdue enterprise reference architecture. *Comput. Ind.* **24**(2), 141–158 (1994)
68. Yin, C., Zhu, Y., Fei, J., He, X.: A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access* **5**, 21954–21961 (2017)
69. ZVEI-German Electrical and Electronic Manufacturers' Association. *Module-based production in the process industry-Effects on automation in the "Industrie 4.0" environment*. White Paper (Mar. 2015)
70. ZVEI-German Electrical and Electronic Manufacturers' Association. *Process INDUSTRIE 4.0: The Age of Modular Production*. White Paper, Frankfurt (2019)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Algorithmic and Implementation-Based Threats for the Security of Embedded Machine Learning Models



Pierre-Alain Moëllic, Mathieu Dumont, Kevin Hector, Christine Hennebert, Raphaël Joud, and Dylan Paulin

Abstract The large-scale deployment of machine learning models in a wide variety of AI-based systems raises major security concerns related to their integrity, confidentiality and availability. These security issues encompass the overall traditional machine learning pipeline, including the training and the inference processes. In the case of embedded models deployed in physically accessible devices, the attack surface is particularly complex because of additional attack vectors exploiting implementation-based flaws. This chapter aims at describing the most important attacks that threaten state-of-the-art embedded machine learning models (especially deep neural networks) widely deployed in IoT applications (e.g., health, industry, transport) and highlighting new critical attack vectors that rely on side-channel and fault injection analysis and significantly extend the attack surface of AIoT systems (Artificial Intelligence of Things). More particularly, we focus on two advanced threats against models deployed in 32-bit microcontrollers: model extraction and weight-based adversarial attacks.

1 Introduction

The development of AI-based systems in many IoT domains comes across security challenges that are especially important given that these systems are usually performing critical tasks with the help of sensitive data. Since almost a decade, the machine learning (ML) security community (mainly structured on *Adversarial Machine Learning* [1] and *Privacy-Preserving Machine Learning* [2]) works unceasingly with a threefold objective: (1) turn the spotlights on attacks that target every step of the machine learning pipeline with an impressive diversity of attack vectors, (2) propose defense schemes to improve the robustness of the models or the systems, (3) build sound evaluation methodologies to properly assess the intrinsic robustness of models or the real impact of protections.

P.-A. Moëllic (✉) · M. Dumont · K. Hector · C. Hennebert · R. Joud · D. Paulin
CEA-LETI, Grenoble, France
e-mail: Pierre-Alain.MOELLIC@cea.fr

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_6

105

However, most of these works are focused on demonstrating (or defending against) attacks that exploit the inputs and the outputs of a white or black-box target model seen as a pure algorithmic abstraction. Obviously, these studies are compulsory since they enable to reveal theoretical flaws, but the attack surface needs to encompass attack vectors related to the *physical* implementation of the models on specific hardware platforms. Interestingly, one can draw a parallel with cryptography-based systems for which international standardization and certification are well established. For example, if we consider the actual standard for symmetric encryption (AES—Advanced Encryption Standard), this algorithm is known to be secure as no cryptanalysis-based attack has been proven (unless the brute-force strategy). However, several physical attacks (typically, *side-channel* and *fault injection analysis*) have been successfully demonstrated on many platforms (ASIC, FPGA, microcontrollers...) to recover the secret key. Then, to claim a certain level of security, an AES-based system must be evaluated against a set of state-of-the-art physical attacks to obtain the related certification level as it is the case for the Common Criteria.¹

The purpose of this chapter is twofold. First, we outline a panorama of the threats against embedded machine learning models (especially state-of-the-art deep neural networks) as well as the available defense strategies. Second, we highlight advanced physical attacks against the confidentiality and integrity of models since these threats are still insufficiently concerned, despite a recent amplification of these topics in the hardware security community.

2 Threat Models

The definition of a threat model enables to precisely set the goal, knowledge and ability of an adversary as well as the most important features of the system to defend [3]. First, we define the formalism used in this chapter, then we briefly describe the different features composing a threat model.

2.1 Formalism

First, we distinguish a ML model, \mathbb{M} , as an *abstract algorithm* and its *implementations* \mathcal{M} after its deployment in different hardware and software environments. In some cases, these deployed models may be functionally different because of optimization techniques that modify their architecture or some parameters (e.g., quantization, pruning).

A supervised neural network model \mathbb{M}_W is a parametric model that maps an input space $\mathcal{X} = \mathbb{R}^d$ to the output space \mathcal{Y} . W is the set of parameters to be learned and $\mathcal{D}_{\mathcal{X}}$ is the data distribution. Typically, for a classification task, \mathcal{Y} is a finite set of labels

¹ <https://www.commoncriteriaportal.org/>.

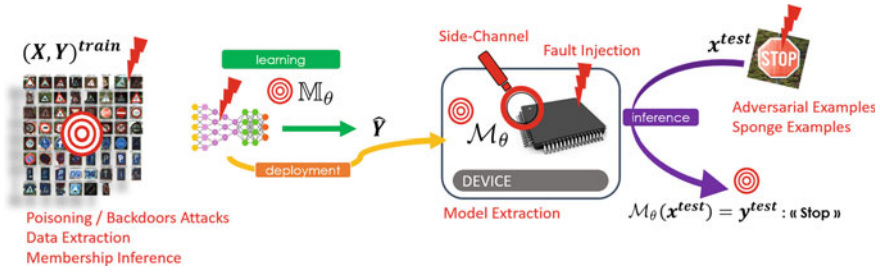


Fig. 1 The (supervised) ML pipeline for an embedded ML model deployed on a mobile device. State-of-the-art threats cover every stage of the pipeline and can target both data and the model

$\{0, \dots, C - 1\}$. According to the *Empirical Risk Minimization* framework, \mathbb{M}_W is trained by minimizing a loss function \mathcal{L} (e.g., the cross-entropy loss) that quantifies the error between the prediction $\hat{y} = M_W(x)$ and the groundtruth label y as defined in Eq. 1:

$$W^* = \arg \min_W \left(\mathbb{E}_{x, y \sim \mathcal{D}} [\mathcal{L}(M_W(x); y)] \right) \tag{1}$$

Figure 1 illustrates the traditional (supervised) *machine learning pipeline*: the data used for training are samples of \mathcal{X} and compose the *training data set*: X^{train} . A fraction of that data set, X^{val} , is used to intrinsically validate the training in order to evaluate the behavior of \mathbb{M}_W on unseen data and therefore measures the so-called *generalization gap*. At inference time, the learned model is used on new samples and outputs the probability vector that x^{test} belongs to each class of \mathcal{Y} .

2.2 Adversarial Objectives

Classically, the goal of an adversary is defined thanks to the confidentiality/integrity/availability triad that we develop in Sect. 3.

Confidentiality and privacy concern both the data and the model. First, extracting data—even partial information—captured or memorized within a model is a critical threat for many domains such as medical applications. Second, another major adversarial objective is to reverse-engineer a protected model by extracting information about its architecture or parameters. *Model extraction* is a growing concern in the security community with both API-based and physical attacks.

Integrity-based attacks aim to deflect the nominal behavior of a model. Classically, the objective may be to fool the prediction of a model on a global scale (i.e., to drop the average accuracy on a test set) or only for very specific inference inputs. The most popular threats are adversarial examples and poisoning attacks respectively at inference and training times.

Threaten the **availability** means that the adversary targets the overall system hosting the ML algorithms so that it will become unavailable. That also encompasses adversaries that significantly deteriorate the performance (quality or system-based such as processing times) so that that it becomes useless.

2.3 *The System Under Attack*

Distinction between *sensor* and *API*-based systems is fundamental because both types induce different ways of interacting with the model. For example, most of the works related to inference input-based attacks (such as adversarial examples explained in Sect. 3.2.2 and Fig. 3) deal with *API*-based systems (more particularly cloud-based ML-as-a-service systems) for which an adversary may have a full control over the inputs contrary to a sensor-based system that brings with additional physical (and software) layers.

Another essential point is the position of the system within the ML pipeline, more essentially its ability to perform both training and inference steps. For many IoT applications, ML models are previously trained on high-computing platforms (essentially GPU-based servers) and then deployed to connected devices only for pure inference purposes. This paradigm is no longer exclusive because of the fast development of powerful ML-compliant hardware architectures, the improvement of model optimization techniques as well as advanced training strategies dedicated to IoT (e.g., federated learning). Even platforms with memory and power constraints will likely be able to support training tasks. In that case, the attack surface is significantly wider and an attacker may exploit potential flaws at training time (e.g., poisoning or backdoor attacks) to weaken the resulting learned model and perform efficient attacks at inference time.

2.4 *Knowledge and Capacity of an Adversary*

As previously mentioned, an important criterion is the ability of an adversary to perform an attack at training or/and at inference time. At training time, an adversary may have a full access to the training set as well as the training process itself. At inference time, the attacker generally exploits (at least) the inputs and outputs by querying the model according to potential restrictions. However, the most critical point is the level of knowledge of the adversary about the target model.

An adversary that performs a **white-box** attack has a perfect knowledge of the parameters and architecture of the target model. This setting may be widened to the knowledge and the access to the training set. On the contrary, with **black-box** attacks, the attackers have no (or partial) information about the target model. They need to guess some information thanks to his expertise and his knowledge about the task or by querying the model.

Efforts on adversarial and privacy-preserving machine learning have highlighted (and still do) many traps in the evaluation of the robustness of ML systems such as the underestimation of the attackers. An important evaluation standard is the definition of a so-called *worst-case scenario* that relies on the assumption of an advanced adversary who has enough expertise and knowledge about protections to perform *adaptive attacks* that completely thwart these defenses with only minor changes in state-of-the-art attacks as exposed in [4].

2.5 Attack Surface

As mentioned in introduction, the complexity of defending embedded machine learning models mainly relies on the extent of the attack surface. This complex attack surface must encompass the algorithmic threats *and* the implementation-based threats as illustrated in Fig. 2. The first ones will exploit the theoretical flaws of the models and the second ones include powerful physical attacks such as side-channel analysis (SCA) and fault injection analysis (FIA) that both leverage some characteristics of the (software or hardware) implementation of a model as well as the hardware platforms (e.g., memory types, instruction sets...).

To illustrate that point, let's focus on the Rectified Linear Unit (ReLU) activation function, widely used in many deep neural network models, a piecewise linear function defined as: $ReLU(x) = \max(0, x)$. From a mathematical standpoint, this function has several strong properties. For example, ReLU maps input values to $[0, +\infty]$ and then *cancel*s out negative inputs. Moreover, its second derivative is zero everywhere except at its critical point ($x = 0$). These properties are exploited in some attacks, for example a cryptanalysis-based attack from Carlini et al. [5] to reverse-engineer the parameter values of a multilayer perceptron (MLP) model or to attack the training process by altering the initialization process of the parameters [6]. These attacks are algorithmic ones since they directly exploit the definition of ReLU,

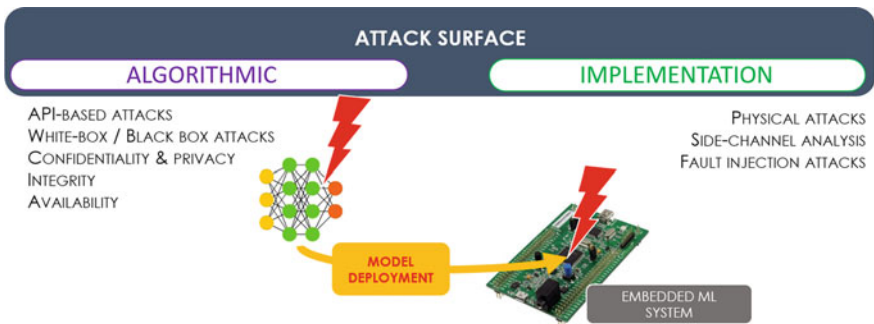


Fig. 2 For embedded ML-based systems, the attack surface must encompass algorithmic and physical threats including physical attacks

whatever the way this function has been implemented and deployed on a device. On the contrary, other attack vectors rely on the implementation flaws of ReLU, such as timing attacks that will exploit the fact that most ReLU implementations are non-constant time, which can leak some information about the architecture of a protected model.

3 A Panorama of Algorithmic Attacks

3.1 Confidentiality and Privacy Threats

3.1.1 Data Leakages

In many critical domains, the training data is exclusively or partially composed of private information that may be captured by the model during the training process. It is obviously the case for models suffering from overfitting, but Carlini et al. [7] also demonstrated that large language models tend to memorize training data at the early steps of training. Note that privacy breaches may concern even low levels of information. The best example is membership inference attack [8]: the adversary aims at guessing if an inference input fed to the model belongs to the training dataset. This *membership* knowledge about a data can be a critical information in some cases, for example in medical prediction tasks or biometric-based access control systems.

3.1.2 Model Theft

The confidentiality of models is also an important issue that drives stakeholders to protect models against reverse-engineering attacks. There are many goals underlying a model theft attack as detailed by Jagielski et al. [9] that highlight the concepts of *fidelity* and *accuracy*. In a *fidelity* context, an adversary aims to precisely extract the model's characteristics in order to obtain a *clone model*. Additionally to model theft, the adversary may aim to steal a model to shift from a black-box to a white-box context in order to craft more efficient attacks. On the contrary, an *accuracy* objective refers to performing well over the underlying learning task of the original model. The attacker aims at stealing the performance of the model and, effortlessly, reach equal or even superior performance. In such a case, the exact extraction of the architecture nor the parameters values are compulsory.

API-based approaches for model extraction exploit input/output pairs and potential information about the target model. We highlight a milestone work from Carlini et al. who cleverly consider the extraction of parameters as a *cryptanalytic* problem [5] and demonstrate significant improvements from [9]. The threat model sets an adversary that knows the architecture of the target model but not the internal parameters. The attack relies on the ReLU properties, more precisely the fact that the second derivative is null everywhere except at a *critical point*. The authors demonstrate

a complete extraction of a 100,000 parameters MLP (one hidden layer) with $2^{21.5}$ queries with a worst-case extraction error of 2^{-25} .

In Sect. 4.1 we focus on implementation-based approaches for model extraction.

3.2 Integrity-Based Attacks

3.2.1 At Training Time

Attacking the integrity of a model at training time relies on a strong assumption related to the ability of the adversary. Poisoning attacks aim to control the behavior of a model by manipulating its training data [10–12]. By injecting, modifying or removing training samples, an adversary aims at altering the decision boundary. Therefore, several objectives can be linked to such attacks. For example, attackers want to degrade the overall accuracy of the model or they want to target a very specific behavior when the model is fed with a *trigger* signal.

A first approach is to alter the training labels as proposed in [13] in a healthcare context with an additional model that learns the influence of a data and therefore enables to highlight samples of higher interest to poison.

However, most of the state-of-the-art poisoning attacks are based on the inputs by altering or injecting new training samples. *Trigger-based* data poisoning (also referred to as *backdoor poisoning attack*) aims at fooling a model at inference time with inputs containing a specific *trigger* (e.g., an image patch or a specific word sequence) that has been learned by the model thanks to poisoned data [14]. On the contrary, *trigger-less* poisoning attacks (or *features-based*) are only focused on the training process. Shafahi et al. [10] propose altering a clean training data sample so that some features extracted from the model are very close to ones from a target training sample that belongs to another class. The authors add an adversarial watermark in order to blend features between the clean and target samples.

3.2.2 At Inference Time

An adversarial example is an input x^* , crafted from an initial sample x , that is misclassified by a model even though it is the result of small (even imperceptible) perturbations (see Fig. 3). Formally, Szegedy et al. [15], define x^* as in Eq. 2:

$$\arg \min_{\delta} \mathbb{M}_{\Theta}(x + \delta) = l \quad (l \neq \mathbb{M}_{\Theta}(x)) \quad \text{with } x^* = x + \delta \in \mathcal{X} \quad (2)$$

With δ a perturbation applied to an input x so that x^* is still in \mathcal{X} , and l the (mis)classification output. A classical setting bounds the perturbation with a l_p norm: $\|x - x^*\|_p \leq \epsilon$ with ϵ the *adversarial budget*. Classically, the l_2 and l_{∞} norms are used, which lead to an alteration of every dimension of the input. State-of-the-art



Fig. 3 From [16]: an illustration of a successful adversarial example that fools a model. The adversarial perturbation is magnified for visualisation

crafting methods are mainly based on the gradients of the loss w.r.t. the inputs such as the PGD (projected gradient descent) attack, proposed in [17], an iterative approach defined as:

$$\begin{aligned}
 x_0^* &\sim \mathcal{B}(x, \epsilon) \\
 x_{t+1}^* &= x_t^* + \lambda \cdot \text{sign}\left(\nabla_x \mathcal{L}(\mathbb{M}_\Theta(x_t^*), y)\right)
 \end{aligned}
 \tag{3}$$

Where $\mathcal{B}(x, \epsilon)$ is an ϵ -ball around x according to the l_p norm. PGD is performed several times and, for each attempt, the initial state x_0 is picked randomly in \mathcal{B} .

Interestingly, l_0 -based attacks aim to minimize the number of perturbed dimensions such as the so-called One-Pixel Attack [18] or [19] that reaches more than 95% success rate on CIFAR10 by perturbing at most 10 pixels by combining sparsity and imperceptibility (a mix of l_0 and l_∞ constraints).

In a black-box setting, the adversary cannot directly compute the gradients $\nabla_x \mathcal{L}(x, y)$ but can take benefit from two types of approaches. First, the attacker can leverage the transferability property of adversarial examples [20]. Indeed, adversarial examples crafted on a model \mathbb{M} are likely to be successful on a model \mathbb{M}' that performs the same task. Therefore, an adversary can design and train a *substitute model* as close as possible from the target model with which he can compute gradients. Second, an adversary may approximate the gradients by exploiting a set of input/output pairs gathered by massively querying the model [21]. The complexity of these query-based attacks depends on the nature of the available outputs (logits, scores or labels only).

Some works shift the digital adversarial examples into the physical world [22] such as Eykholt et al. with road sign classifiers fooled with posters or stickers [23] or face recognition systems fooled with handcrafted glasses in [24]. Moreover, several works highlight that the classical gradient-based attacks (such as PGD) usually demonstrated on computer vision tasks are efficient without any adaptation on multivariate time series. Adversarial examples have also been successfully crafted for

other applications like speech-to-text [25], Q&A systems, malware detection [26] and even for reinforcement learning [27, 28].

3.3 Availability

First, the availability threats encompass all the typical network-based attacks that aim at weakening a computing or communication infrastructure such as Denial-of-Service (DoS) attacks. However, some works focus on specific availability-based attacks that target the training process. In [6], Grosse et al. propose exploiting the initialization of the weights as a way to fool the training. The attack aims to severely degrade the model's performance and increase training time. The basic principle of the attack is straightforward and relies on the property of the ReLU activation function to map negative inputs to zero. By controlling even a small proportion of the initial value of the weight matrix, the attacker leverages the ReLU property to set to zero activation values with a cascade effect with deeper models. In some cases, the training is impossible since too many neurons are shutdown.

Another concerning attack vector is the batch-feeding process. To work properly the standard *mini-batch* stochastic gradient descent method (SGD) used in deep learning relies on the assumption of a uniform random sampling of the training data. In [29], Shumailov et al. consider an adversary that sets in a strong black-box threat model, without knowledge about the model nor any prior knowledge of the training data. The basic principle of the attack is to interact on the batching part of the ML pipeline in order to thwart the randomness assumption of the *mini-batch* SGD leading to strong convergence issues.

4 A Focus on Physical Attacks

The attacks described in the previous section are algorithmic threats with models seen as pure mathematical abstractions. In this section, we focus on implementation-based attack vectors by highlighting the use of *side-channel* (SCA) and *fault injection analysis* (FIA) for model extraction and integrity-based attacks. We highlight recent results concerning 32-bit MCUs, typically used for IoT applications. For a broader survey on hardware security of deep neural networks, interested readers may refer to [30].

4.1 Model Extraction Based on Side-Channel Analysis

As previously detailed in Sect. 3.1.2, model extraction is becoming a major threat with different adversarial objectives (model cloning, functionality theft). Interestingly,

even if several API-based strategies have been proposed, model extraction is also a threat significantly studied by the hardware security community because of the well-known efficiency of side-channel analysis in extracting critical information on an embedded program (both the data and the instructions).

4.1.1 Side-Channel Analysis

Side-channel analysis (SCA) are physical attacks relying on the *observations* of some physical signals that depend on both the algorithm and the processed data. A classical attack targets the software or hardware implementations of cryptographic primitives to recover a secret information (e.g., ciphering key) [31]. A typical setup is to feed the system with known inputs and capture the electromagnetic emanations of the target chip. The information is stored for further statistical processing and is traditionally referred as *traces*. A *leakage model* links these traces with known algorithmic steps that process the (known) inputs and the secret information (e.g., the *SubBytes* operation for the AES). Classically, a *leakage model* enables to bridge the physical observations and the target algorithm and data. Classical choices for leakage models are the Hamming weight and Hamming distance. Then, an adversary can make some hypothesis on the secret values and correlate these hypothesis to the recorded traces thanks to the leakage model. It is possible to extract the most likely hypothesis because this value will better explain what has been physically observed. This technique is known as Correlation Power Analysis (CPA) but simpler analysis are possible, for example by simply *reading* the traces (Simple Power Analysis–SPA) that may emphasize patterns as it is the case with data-dependent constant-time algorithms. Figure 4 illustrates a power trace of two inferences of a MobileNet (v2) neural network deployed on a Cortex-M7 platform with clear distinctions separating the different structural blocks of the network.



Fig. 4 An electromagnetic trace (red) of two consecutive inferences of a MobileNet (v2) model running on a Cortex M7 microcontroller. We can observe the clear separations between the convolutional basic blocks of the model. The blue line is a trigger signal showing the start and stop time sample of each inference

4.1.2 Timing Analysis

Because the inference process of an embedded model is hardly time-constant, *timing analysis* is a classic way to infer information about the architecture and (in some cases) the parameter values of a model. For example, Gongye et al. exploit extra CPU cycles (on a $\times 86$ processor) for IEEE-754 multiplications or additions with *subnormal* values [32] in order to precisely recover the weights and bias of a 4-layer neural network model. Maji et al. [33] also demonstrate parameter extraction with timing analysis by exploiting the ReLU activation function and the multiplication operation with floating-point, fixed-point and binary models deployed on three platforms² without floating-point unit (FPU).

4.1.3 SCA-Based Extractions

Several works [34, 35] highlight the use of side-channel analysis to extract the value of the internal parameters of a deep neural network model. Here, we focus on the software implementation of neural network models on typical IoT platforms based on microcontroller such as Cortex-M 32-bit microcontrollers.

Exploitable leakages are related to the basic operation used in a neural network, that is the multiplication operation between a secret parameter (also called weight) and an input value (i.e., the input data or the output of the previous layer). For a full-precision implementation of a neural network, this multiplication operation handles two IEEE-754 32 bit floating-point values. We remind that a 32-bit single-precision floating-point value a is composed of a sign, exponent and mantissa parts as in Eq. 4:

$$\begin{aligned} a &= (-1)^{b_{31}} \times 2^{(b_{30} \dots b_{23})_2 - 127} \times \left(1.b_{22} \dots b_0\right)_2 \\ &= (-1)^{S_a} \times 2^{E_a - 127} \times \left(1 + 2^{-23} \times M_a\right) \end{aligned} \quad (4)$$

With S_a , E_a and M_a are respectively the sign, exponent and mantissa values. Then, the result of the multiplication operation $o = x \times w$ with x and w the input and parameter value of a neuron, leads to the sign (S_o), exponent (E_o) and mantissa (M_o) detailed in Eq. 5:

$$S_o = S_x \oplus S_w, E_o = E_x + E_w - 127, M_o = M_x + M_w + 2^{-23} \times M_x \times M_w \quad (5)$$

Joud et al. [35] demonstrate a coarse-to-fine strategy to exploit side-channel leakage in order to precisely extract the 32 bits of a parameter. The basic idea is to first focus on the exponent part and keep (with CPA) the most likely hypothesis and, then, progressively extend the correlation analysis to other bits of the mantissa. With

² ATmega-328P, ARM Cortex-M0+, RISC-V RV32IM.

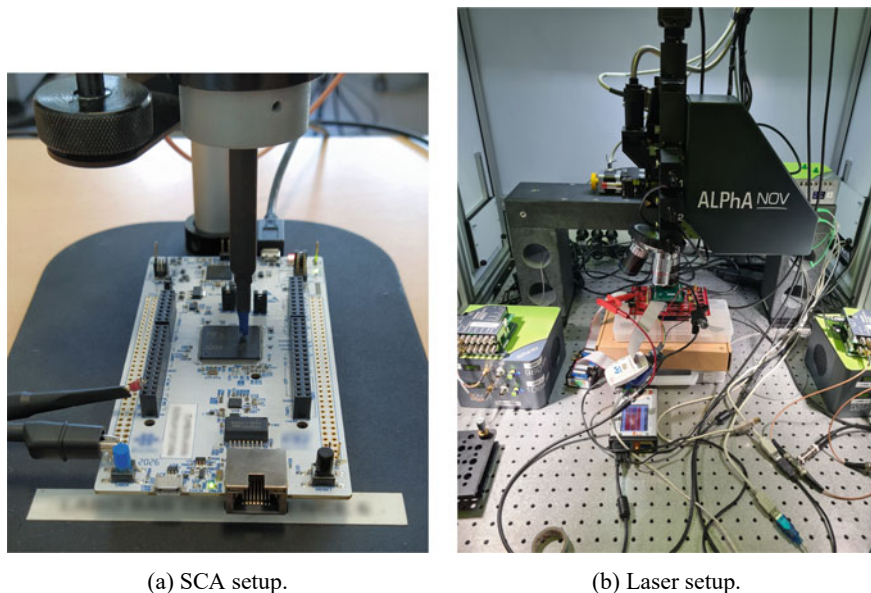


Fig. 5 (Left) Experimental setup used in [35] for the high-precision extraction of 32-bit parameters of a shallow MLP model on a Cortex-M7 microcontroller. A electromagnetic probe is precisely position on the chip the capture EM traces. (Right) Laser bench used for laser fault injection against an embedded neural network model in a Cortex-M 32-bit microcontroller

this approach, experiments on a Cortex M7 target using FPU (floating-point unit) demonstrate an extraction precision close to 10^{-7} for the absolute value of a set of parameters of a shallow MLP model. Figure 5 shows the setup for collecting the traces for these experiments.

However, some challenges remain open for a full extraction of internal parameters [35]. First, the use of ReLU as activation function, by mapping the output of a layer into positive or null values, significantly increases the complexity of the extraction of the bit sign. Second, a single error extraction in a fully-connected network automatically and dramatically leads to the impracticality of the extraction for all the weights of the next layers. Moreover, the exploitable leakage relies on the multiplication operation that does not concern bias values and the addition operation is more complex to exploit than the multiplication with classical CPA methods.

4.2 Weight-Based Adversarial Attacks

Input-based integrity attacks are not the only attack vectors available for an adversary that aims to fool a model at inference time. In the past few years, several implementation-based attacks have been proposed and demonstrated that directly

target the parameters stored in memory (e.g., DRAM or Flash memory). Alongside safety-related efforts that evaluate the robustness of ML models against random faults, these works highlight the lack of robustness of deep neural network models to fault injection attacks that alter the data, the parameters as well as the instruction flow [36, 37].

4.2.1 Target the Parameters Stored in Memory

As formalized in [38] or [39], a parameter-based attack aims at maximizing the loss (i.e., increasing mispredictions) on a small set of test inputs, as represented in the first part of Eq. 6. As for the *imperceptibility* criterion of adversarial examples, the attacker may add a constraint over the perturbation by bounding the bit-level Hamming distance (HD) between the initial (W) and faulted parameters (W'), corresponding to an *adversarial budget* S (second part of Eq. 6).

$$\max_{W'} \sum_{i=0}^{N-1} \mathcal{L}(M(x_i; W'), y_i) \text{ s.t. } HD(W', W) \leq S \quad (6)$$

A state-of-the-art parameter-based attack is the (white-box) Bit-Flip Attack (hereafter BFA) initially proposed by Rakin et al. [38]. The goal is to decrease the performance of a model by selecting the most sensitive bits of the stored parameters and progressively flipping these bits until reaching an adversarial goal. Typically, the objective of an attacker is to ultimately degrade the model so that its accuracy corresponds to a random-guess level (as in [38] or [40]). The selection of the bits is based on the ranking of the gradients of the loss w.r.t. to each bit $\nabla_b \mathcal{L}$, computed thanks to a small set of inputs. The most sensitive bit is permanently flipped according to the gradient ascendant as defined in [38] with Eq. 7 (with \hat{b} , the bit after bit-flip) and the process is repeated until the adversarial objective is met.

$$\hat{b} = b \oplus (\text{sign}(\nabla_b \mathcal{L})/2 + 0.5) \quad (7)$$

Many simulation-based works have been proposed since the BFA presentation [39–44] but fewer efforts have been made to practically implement the attacks. Among these works, an important milestone is [37] that demonstrates the BFA with RowHammer on an Intel i7-3770 CPU platform. RowHammer [45] is a powerful attack vector that can only target DRAM memory cells. It relies on the interaction between DRAM cells, more particularly between the nearby memory rows of a target cell stressed by an adversary that can lead to bit flips in those adjacent rows.

4.2.2 Practical Experiments on Microcontrollers

In this section, we present experiments on a Cortex-M 32-bit microcontroller that embeds an 8-bit quantized neural network thanks to NNoM an open-source deployment library (*Neural Network on Microcontrollers* [46]). Our attack vector (BFA-like attack) and fault model enable to evaluate the robustness of a model against an advanced adversary who aims at significantly altering the performance of a model with a very limited number of faults.

Working on a 32-bit microcontroller with SRAM and Flash memory, we cannot exploit RowHammer technique. A state-of-the-art means of fault injection, used in most of the certification and security testing labs is laser fault injection (LFI). We considered an accurate fault model relevant for laser injections previously explained and demonstrated for Flash memory of Cortex-M MCU by Colombier and Menu [47, 48]: the bit-set fault model that consists of setting a targeted bit to a logical 1. When targeting a Flash memory at read time, the induced bit-set is transient, it affects the data being read at that time while the stored value is left unmodified. LFI is a very accurate technique, both temporally and spatially. Depending on the laser spot diameter, up to two adjacent bits can be faulted simultaneously [48].

Figure 5b shows the laser setup we used for these experiments. The laser platform gathers two near-infrared independent laser sources focused through the same lens (spot diameter is ranging from 1.5 to 15 μm) with a maximum power of 1,700 mW. An infrared camera is used to observe the spot location on the target and a XY stage enables moving the objective above the entire surface of the target device.

Since faulting every bit of every parameter stored in memory may be impractical we leverage the BFA principle to select the most sensitive bits to shoot with the laser and adapt it to our laser fault model (i.e., bit-set). As recommended in [42], we also adapted the adversarial objective by introducing an *adversarial budget* (20 bit-sets) representing a maximum of faults we are able to process.

We implement a MLP model trained on the standard digit recognition dataset (MNIST [49]). We reduced the complexity of our model by compressing the input data (784 pixels grayscale images) from \mathbb{R}^{784} to \mathbb{R}^{50} with principal component analysis. The model has one intermediate layer of 10 neurons and ReLU as activation function. The resulting model has 620 trainable parameters (including bias). After training, the model reaches 92% of accuracy on the test set.

We ran a BFA simulation over all the weight columns and bit lines of the Flash memory that pointed out the most significant bits of the second column weight as the most sensitive. The model accuracy was evaluated over 100 inferences. The blue curve in Fig. 6 represents our experimental results while the red one is the BFA simulations for the most significant bits. First, we can notice that experimental and simulation results are quite similar, meaning that we can guide our LFI evaluation with high reliability and confidence.

The fact that experimental results are slightly more powerful than simulations may be explained by the impact of the width of the laser spot on nearby memory cells. We observe that for an adversarial budget of only 5 bit-sets (0.1% faulted bits) the embedded model accuracy drops to 39% which represents a significant loss

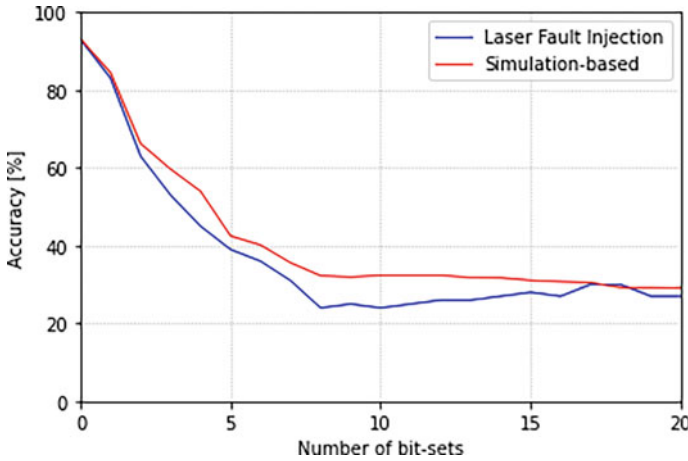


Fig. 6 Experimental and simulation results of a laser fault injection (LFI) attack on a MLP model trained on MNIST targeting the 20 most sensitive MSB of the 2nd weight column

and a strong integrity impact compared to the nominal performance of 92.5%. After 10 bit-sets (accuracy to 25%), the most effective faults have been injected and the accuracy does not decrease anymore: this positions the level of robustness of the model according to the adversarial budget.

5 Protecting ML System

In this section we discuss the ways of defending a ML-based systems in view of the different threats we presented in the previous sections. Before focusing on the specific protections and countermeasures available in the literature to thwart algorithmic and physical attacks, we first examine the protection of a ML-system at a wider level. Indeed, in many IoT applications, a ML inference program is a critical part of an overall system that strongly interacts with other subsystems. Therefore, as in many critical information systems, strong integrity-requirements apply to an ML inference to guarantee the authenticity of the inference process as well as its inputs and outputs.

5.1 Embedded Authentication Mechanism

The traditional security model of a cyber-physical system is composed of several successive layers. The first layer consists in integrating in-depth physical countermeasures in the electronic design. The second layer is dedicated to reducing the attack surface to protect the ML system against cyber-security attacks. However,

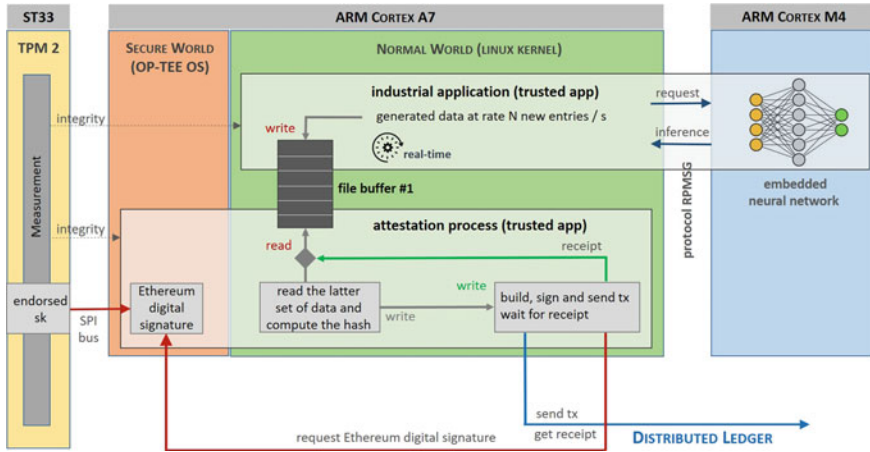


Fig. 7 Design of HistoTrust System-on-Module includes a TPM (ST33), a Cortex-A7 and a Cortex-M4 in a STM32MP1 platform [50]

vulnerabilities still exist that may be exploited by advanced attackers. This is why the third layer consists in integrating anomaly detection mechanisms, and the fourth layer aims at tracing the events and the behavior of the system.

In [50], Paulin et al. propose dealing with detection and traceability needs for an ML inference process by staying at the device-level. The platform named HistoTrust integrates the ML system on a System-on-Module (SoM) composed of two micro-controllers and hardware security components such as a secure enclave for Trusted Environment Execution (TEE) and a Trusted Platform Module (TPM), as illustrated and detailed in the Fig. 7. This enables the integration of embedded security mechanisms as close as possible to the ML system.

HistoTrust embeds an attestation scheme, based on the TPM2 attestation principle, that provides evidences of events and inferences issued by the embedded neural network. These are integrated into transactions sent to an ethereum blockchain [51].

This enables tracing the inference outputs by the embedded AI, the events that may modify the behaviour of the AI and to monitor the integrity of the embedded neural network model. The transaction authenticates the issuer device and the embedded AI in an history known to be immutable. By this way, decisions or choices relevant from the usage of an embedded neural network are authenticated through the use of secure hardware components, and can be justified. This is useful in the event of a failure to understand the behavior of the AI and to attribute accountability to those who trained, configured, embedded or used the AI.

A typical scenario for such a device-centered authentication scheme is for integrity audit of a system composed of several ML embedded systems in devices belonging to different stakeholders. In such a scenario, IoT devices perform a task thanks to ML-based algorithm (in [50], inferences of a classical convolutional neural network model). A detected anomaly is linked to the detection timestamp that corresponds

with a record in the ledger. Thus, we can consider an history of transactions starting from the detection timestamp. Each transaction/attestation embeds the authentication code (account address) of the device as well as the hashing of the data produced by this device at the transaction timestamp. An independent auditor may ask each stakeholder to provide the raw data, certified, produced by their devices. The auditors check that this data is authentic and not corrupted thanks to the transactions/attestations recorded in the ledger. This data can be safely used and exploited to further investigate the source of the anomaly and potentially unveil integrity or authenticity breaches.

5.2 Main Defenses Against Algorithmic Attacks

A first class of defenses encompasses all the input *pre-processing* techniques that basically aim at monitoring the inputs that feed the models, for example by excluding statistical outliers. Another approach is to purify or, on the contrary, drown potential alterations of the inputs.

A second class of approaches gathers so-called *hiding* techniques. For many of API-based attacks, adversaries take benefit from information provided by the system that may not be necessary to correctly achieve a task (such as all the prediction scores whereas the predicted label).

Another important defense scheme is based on *model hardening* at training time. For example, *differential privacy* [52] or *adversarial training* [17] are standards to make models more robust against privacy and integrity-based attacks. Moreover, if it is possible according to the characteristics and the requirements of the system, the use of ensemble methods is known as a good strategy to weaken the impact of attacks.

5.3 Countermeasures Against Physical Attacks

To thwart timing analysis, some simple countermeasures are available, such as the *flush-to-zero* mode offered in many embedded platforms (e.g., ARM Cortex-M cores) which turns subnormal values into zeros or to favor constant-time implementation of neural network primitives like the ReLU activation function.

For SCA-based parameters extraction, randomization may significantly enhance the complexity of an attack and may take place at different levels. Traditional *hiding* techniques encompass the use of random dummy instructions that desynchronize the traces. Similarly, at the neuron-level, the weighted sum between the parameters and the inputs as well as the addition of the bias can be processed in a randomized order (at each inference process). Moreover, additional noise can be efficiently applied (i.e., with minor influence on the accuracy of the model) to bring uncertainty in

the input layer values that are important knowledge for the adversary when making hypothesis on the secret values (parameters).

Additionally to traditional countermeasures against fault injection [53], specific defense schemes against BFA encompass weight clipping, model pruning [54], clustering-based quantization [36, 55], code-based detectors [56] or adversarial training [39]. The practical evaluation of these defenses against fault attack means such as RowHammer, glitching or LFI is an important direction for future research efforts. As for adversarial examples (with so many defenses regularly *broken* afterwards) the definition of proper and sound evaluations of defenses against parameter-based attacks against embedded ML models is a research action of the highest importance.

6 Conclusion

The large-scale development and deployment of IoT systems, relying on Artificial Intelligence solutions, raises critical security issues that highlight two urgent needs. First, ML practitioners have to define security requirements at the early stages of their design and development processes: with so many demonstrated attacks at every stage of the ML pipeline, ML security breaches are unlikely to be *simply patched afterwards* as in the standard security fairy tales. Second, despite major efforts from the ML Security community, evaluation methodologies to properly assess the impact of attacks, robustness of models as well as defense benefits have to be strengthened and disseminated. The upcoming challenges for the security of machine learning models and systems are essentially focused on the design and development of robust defenses including certifications or guarantees as well as taking into consideration more realistic attacks and threat models. Indeed, for now, most of the state-of-the-art research efforts are *model*-centered whereas real-world applications need to consider security at a global *system*-scale. These challenges could be overcome by joint efforts between the adversarial machine learning community and AI-system stakeholders (designers, developers, end-users).

References

1. Biggio, B., Roli, F.: Wild patterns: ten years after the rise of adversarial machine learning. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, pp. 2154–2156 (2018)
2. Al-Rubaie, M., Chang, J.M.: Privacy-preserving machine learning: threats and solutions. *IEEE Secur. Priv.* **17**(2), 49–58 (2019)
3. Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Berkay Celik, Z., Swami, A.: The limitations of deep learning in adversarial settings. In: IEEE European Symposium on Security and Privacy, pp. 399–414. IEEE (2016)
4. Tramer, Florian, Carlini, Nicholas, Brendel, Wieland, Madry, Aleksander: On adaptive attacks to adversarial example defenses. *Adv. Neural Inf. Process. Syst.* **33**, 1633–1645 (2020)

5. Carlini, N., Jagielski, M., Mironov, I.: Cryptanalytic extraction of neural network models. In: Annual International Cryptology Conference, pp. 189–218. Springer (2020)
6. Grosse, K., Trost, T.A., Mosbach, M., Backes, M., Klakow, D.: On the security relevance of initial weights in deep neural networks. In: International Conference on Artificial Neural Networks, pp. 3–14. Springer (2020)
7. Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., et al.: Extracting training data from large language models. In: 30th USENIX Security Symposium (USENIX Security 21), pp. 2633–2650 (2021)
8. Shokri, R., Stronati, M., Song, C., Shmatikov, V.: Membership inference attacks against machine learning models. In: Security and Privacy (SP), 2017 IEEE Symposium on, pp. 3–18. IEEE (2017)
9. Jagielski, M., Carlini, N., Berthelot, D., Kurakin, A., Papernot, N.: High accuracy and high fidelity extraction of neural networks. In: 29th {USENIX} Security Symposium ({USENIX} Security, 20), pp. 1345–1362 (2020)
10. Shafahi, A., Huang, W.R., Najibi, M., Suci, O., Studer, C., Dumitras, T., Goldstein, T.: Poison frogs! targeted clean-label poisoning attacks on neural networks. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, pp. 6106–6116 (2018)
11. Gu, T., Dolan-Gavitt, B., Garg, S.: Badnets: Identifying vulnerabilities in the machine learning model supply chain (2017). [arXiv:1708.06733](https://arxiv.org/abs/1708.06733)
12. Salem, A., Wen, R., Backes, M., Ma, S., Zhang, Y.: Dynamic backdoor attacks against machine learning models. In: 2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P), pp. 703–718. IEEE (2022)
13. Mozaffari-Kermani, M., Sur-Kolay, S., Raghunathan, A., Jha, N.K.: Systematic poisoning attacks on and defenses for machine learning in healthcare. *IEEE J. Biomed. Health Inf.* **19**(6), 1893–1905 (2015)
14. Saha, A., Subramanya, A., Pirsiavash, Hamed: Hidden trigger backdoor attacks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 11957–11965 (2020)
15. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks (2013). [arXiv:1312.6199](https://arxiv.org/abs/1312.6199)
16. Goodfellow, I., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: International Conference on Learning Representations (2015)
17. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: International Conference on Learning Representations (2018)
18. Su, J., Vargas, D.V., Sakurai, K.: One pixel attack for fooling deep neural networks. *Trans. Evol. Comput.* **23**(5), 828–841 (2019)
19. Croce, F., Hein, M.: Sparse and imperceptible adversarial attacks. In: The IEEE International Conference on Computer Vision (ICCV) (2019)
20. Papernot, N., et al.: Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. In: CoRR, pp. 1605.07277 (2016)
21. Chen, J., Jordan, M.I., Wainwright, M.J.: Hopskipjumpattack: a query-efficient decision-based attack. In: 2020 IEEE Symposium on Security and Privacy (SP) (2020)
22. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial examples in the physical world. In: International Conference on Learning Representations (2016)
23. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., Song, D.: Robust physical-world attacks on deep learning visual classification. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
24. Sharif, M., Bhagavatula, S., Bauer, L., Reiter, M.K.: Accessorize to a crime: real and stealthy attacks on state-of-the-art face recognition. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1528–1540 (2016)
25. Carlini, N., Wagner, D.: Audio adversarial examples: targeted attacks on speech-to-text. In: 2018 IEEE Security and Privacy Workshops (SPW), pp. 1–7. IEEE (2018)
26. Grosse, K., Papernot, N., Manoharan, P., Backes, M., McDaniel, P.: Adversarial examples for malware detection. In: Computer Security–ESORICS 2017 (2017)

27. Huang, S.H., Papernot, N., Goodfellow, I., Duan, Y., Abbeel, P.: Adversarial attacks on neural network policies. In: CoRR (2017). [arXiv:abs/1702.02284](https://arxiv.org/abs/1702.02284)
28. Behzadan, V., Munir, A.: Vulnerability of deep reinforcement learning to policy induction attacks. In: Machine Learning and Data Mining in Pattern Recognition: 13th International Conference, MLDM 2017, New York, NY, USA, July 15–20, 2017, Proceedings 13, pp. 262–275. Springer International Publishing (2017)
29. Shumailov, I., Shumaylov, Z., Kazhdan, D., Zhao, Y., Papernot, N., Erdogdu, M.A., Anderson, R.J.: Manipulating SGD with data ordering attacks. In: Advances in Neural Information Processing Systems, vol. 34, pp. 18021–18032 (2021)
30. Mittal, S., Gupta, H., Srivastava, S.: A survey on hardware security of DNN models and accelerators. *J. Syst. Archit.* **117**, 102163 (2021)
31. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks: Revealing the Secrets of Smart Cards, vol. 31. Springer Science & Business Media (2008)
32. Gongye, C., Fei, Y., Wahl, T.: Reverse-engineering deep neural networks using floating-point timing side-channels. In: 2020 57th ACM/IEEE Design Automation Conference (DAC), pp. 1–6 (July 2020). ISSN: 0738-100X
33. Maji, S., Banerjee, U., Chandrakasan, A.P.: Leaky nets: recovering embedded neural network models and inputs through simple power and timing side-channels—attacks and defenses. *IEEE Internet of Things J.* (2021)
34. Batina, L., Bhasin, S., Jap, D., Picek, S.: CSI NN: reverse engineering of neural network architectures through electromagnetic side channel. In: 28th USENIX Security Symposium, pp. 515–532, (2019)
35. Joud, R., Moëllic, P.-A., Pontie, S., Rigaud, J.-B.: A practical introduction to side-channel extraction of deep neural network parameters. In: Smart Card Research and Advanced Applications (CARDIS) (2022)
36. Hou, X., Breier, J., Jap, D., Ma, L., Bhasin, S., Liu, Y.: Security evaluation of deep neural network resistance against laser fault injection. In: Proceedings of the International Symposium on the Physical and Failure Analysis of Integrated Circuits, IPFA, 2020-July, pp. 1–6 (2020)
37. Yao, F., Rakin, A.S., Fan, D.: DeepHammer: depleting the intelligence of deep neural networks through targeted chain of bit flips. In: 29th USENIX Security Symposium, pp. 1463–1480. USENIX Association (Aug. 2020)
38. Rakin, A.S., He, Z., Fan, D.: Bit-flip attack: crushing neural network with progressive bit search. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (Oct. 2019)
39. Stutz, D., Chandramoorthy, N., Hein, M., Schiele, B.: Random and adversarial bit error robustness: energy-efficient and secure DNN accelerators. *IEEE Trans. Pattern Anal. Mach. Intell.* (2022)
40. He, Z., Rakin, A.S., Li, J., Chakrabarti, C., Fan, D.: Defending and harnessing the bit-flip based adversarial weight attack. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14095–14103 (2020)
41. Liu, L., Guo, Y., Cheng, Y., Zhang, Y., Yang, J.: Generating robust DNN with resistance to bit-flip based adversarial weight attack. *IEEE Trans. Comput.* (2022)
42. Hector, K., Moëllic, P.-A., Dumont, M., Dutertre, J.-M.: A closer look at evaluating the bit-flip attack against deep neural networks. In: IEEE 28th International Symposium on On-Line Testing and Robust System Design (IOLTS), pp. 1–5 (2022)
43. Rakin, A.S., He, Z., Li, J., Yao, F., Chakrabarti, C., Fan, D.: T-bfa: targeted bit-flip adversarial weight attack. *IEEE Trans. Pattern Anal. Mach. Intell.* 1–1 (2021)
44. Lee, K., Chandrakasan, A.P.: Sparsebfa: attacking sparse deep neural networks with the worst-case bit flips on coordinates. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4208–4212. IEEE (2022)
45. Kim, Y., Daly, R., Kim, J., Fallin, C., Lee, J.H., Lee, D., Wilkerson, C., Lai, K., Mutlu, O.: Flipping bits in memory without accessing them: an experimental study of dram disturbance errors. *ACM SIGARCH Comput. Archit. News* **42**(3), 361–372 (2014)
46. Ma, J.: A higher-level Neural Network library on Microcontrollers (NNoM) (Oct. 2020)

47. Colombier, B., Menu, A., Dutertre, J.M., Moellic, P.A., Rigaud, J.B., Danger, J.L.: Laser-induced single-bit faults in flash memory: instructions corruption on a 32-bit microcontroller. In: Proceedings of the 2019 IEEE International Symposium on Hardware Oriented Security and Trust, HOST 2019, pp. 1–10 (2019)
48. Menu, A., Dutertre, J.M., Rigaud, J.B., Colombier, B., Moellic, P.A., Danger, J.L.: Single-bit laser fault model in NOR flash memories: analysis and exploitation. In: Workshop on Fault Detection and Tolerance in Cryptography, FDTC, pp. 41–48 (2020)
49. Deng, L.: The mnist database of handwritten digit images for machine learning research. *IEEE Signal Process. Mag.* **29**(6), 141–142 (2012)
50. Paulin, D., Joud, R., Hennebert, C., Moëllic, P.-A., Franco-Rondisson, T., Jayles, R.: Histotrust: Tracing AI Behavior with Secure Hardware and Blockchain Technology
51. Vujičić, D., Jagodić, D., Ranić, S.: Blockchain technology, bitcoin, and ethereum: a brief overview. In: 2018 17th International Symposium Infoteh-Jahorina (Infoteh), pp. 1–6. IEEE (2018)
52. Truex, S., Liu, L., Gursoy, M.E., Wei, W., Yu, L.: Effects of differential privacy and data skewness on membership inference vulnerability. In: 2019 First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA) (Dec. 2019)
53. Barenghi, A., Breveglieri, L., Koren, I., Pelosi, G., Regazzoni, F.: Countermeasures against fault attacks on software implemented AES: effectiveness and cost. In: Proceedings of the 5th Workshop on Embedded Systems Security, WESS'10, pp. 7:1–7:10. New York, NY, USA, (2010). (ACM)
54. Gao, Z., Wei, X., Zhang, H., Li, W., Ge, G., Wang, Y., Reviriego, P.: Reliability evaluation of pruned neural networks against errors on parameters. In: 2020 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), pp. 1–6 (2020)
55. Libano, F., Wilson, B., Wirthlin, M., Rech, P., Brunhaver, J.: Understanding the impact of quantization, accuracy, and radiation on the reliability of convolutional neural networks on fpgas. *IEEE Trans. Nucl. Sci.* **67**(7), 1478–1484 (2020)
56. Javaheripi, M., Koushanfar, F.: Hashtag: hash signatures for online detection of fault-injection attacks on deep neural networks. In: IEEE/ACM International Conference On Computer Aided Design (ICCAD), pp. 1–9 (2021)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Explainable Anomaly Detection of 12-Lead ECG Signals Using Denoising Autoencoder



Rok Hribar and Drago Torkar

Abstract Anomaly detection is an important task in the field of medical diagnostics, especially when it comes to ECG signals. Anomalies in ECG signals can be indicative of various cardiac conditions, such as arrhythmias, atrial fibrillation, ventricular tachycardia, and other, possibly life threatening, cardiac diseases. In medicine AI-assisted anomaly detection is favorable compared to diagnoses prediction because it can identify all out-of-the-ordinary patterns and not only those that are well represented in the available data bases. Also, AI-assisted anomaly detection can help reduce the risk of medical errors due to its ability to detect subtle abnormalities that may not be easily detected by humans. However, traditional anomaly detection methods are often limited in their ability to explain why an anomaly was detected. We propose an approach to an explainable anomaly detection and denoising of 12-lead ECG signals using a denoising autoencoder. The proposed approach is based on the idea of reconstructing the original signal from a noisy version of it, and then using the reconstruction error to detect anomalies, and also pinpoint them on the ECG in a visual way. We evaluate the proposed approach on a publicly available data sets and show that it is able to detect anomalies with high accuracy and explain why they were detected. The developed framework was also implemented as a cloud-based service that enables user-friendly ECG anomaly detection with minimal software and hardware capabilities.

1 Introduction

In recent years, a great potential for utilizing artificial intelligence (AI) in medicine has been recognized that stem from the immense advances made in the field including signal processing [1], image analysis [2] and drug discovery [3]. AI algorithms are able to quickly analyze large amounts of data, such as medical images, patient

R. Hribar (✉) · D. Torkar
Jožef Stefan Institute, Ljubljana, Slovenia
e-mail: rok.hribar@ijs.si

D. Torkar
e-mail: drago.torkar@ijs.si

© The Author(s) 2024
M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_7

records, and lab results, and provide insights that would otherwise take a human doctor much longer to identify. AI can also help with predictive analytics, allowing doctors to better anticipate potential health issues before they arise. Additionally, AI-driven tools can assist with decision support, providing doctors with evidence-based recommendations on the best course of action for each individual patient. Finally, AI can automate mundane tasks, freeing up time for doctors to focus more on their patients' needs. However, trust in AI models is a key factor in determining how widely they are used in medicine. If clinicians and patients trust the accuracy, reliability, and safety of AI models, then they will be more likely to use them as part of their medical decision-making process. To increase trust, explainable AI (XAI) offers a way for physicians and other medical personnel to understand the reasoning behind AI models. If one knows how AI came to some conclusion one is more likely to trust it which in effect brings possibilities to fully benefit from AI in medicine.

XAI is a concept that focuses on making the predictions of an AI model more transparent, understandable, and interpretable for both clinicians and patients. In addition to building trust, XAI is especially important in medicine because it helps to ensure that the decisions being made by AI-based systems are ethically sound, medically accurate, and consistent with patient safety. XAI allows medical professionals to understand how the system has come to its conclusions and gives them more confidence in using the technology for diagnosis, treatment planning, and other clinical tasks. It also provides a layer of transparency so that patients can better comprehend the decisions being made on their behalf. For example, XAI techniques can be used to explain why a certain diagnosis was made or why a particular treatment plan was recommended by the AI system. Additionally, it could potentially reduce medical errors due to increased transparency and accountability of the AI system. Overall, XAI has great potential to increase trust in AI models and enable them to be more widely used in medicine. By providing explanations for the decisions made by AI models, not only will this help to improve the accuracy, reliability, and safety of these models but also foster better communication between clinicians and patients. With increased trust, AI models are more likely to be used more widely as stakeholders become more comfortable with its capabilities.

In this work we focus on building an XAI pipeline, for processing electrocardiogram (ECG) recordings, able to detect anomalies, explain the results and provide visualizations. ECG is an important diagnostic tool in medicine as it provides a non-invasive method of assessing the electrical activity of the heart. The ECG records the electrical signals generated by the heart, which can be used to detect abnormal rhythms and other cardiac abnormalities such as arrhythmias, conduction delays, and myocardial infarction. By analyzing these signals, physicians are able to diagnose various cardiovascular diseases. Additionally, the ECG can provide valuable information about the patient's overall health, such as their risk factors for developing heart disease or stroke. The 12-lead version of ECG recording has the ability to provide a more comprehensive view of the electrical activity of the heart. By recording twelve different leads, it can detect abnormalities that may not be visible on other tests such as single lead ECGs or chest X-rays [4]. It also allows for comparison between multiple recordings over time, which can help identify changes in

cardiac function and diagnose arrhythmias. The 12-lead ECG is especially useful in detecting myocardial infarction (heart attack) because it can show areas of decreased blood flow to the heart muscle [5]. Additionally, it can provide valuable information about the size and shape of the heart chambers, helping to diagnose certain types of cardiomyopathy. As such, the 12-lead ECG is an invaluable tool in diagnosing cardiovascular conditions.

In our work, we chose to build an anomaly detection model as opposed to a diagnoses prediction model because it can identify all out-of-the-ordinary patterns and not only those that are well represented in the available data bases which is more prudent in critical applications such as medicine and especially cardiology. Anomaly detection is an unsupervised learning approach that attempts to detect unusual behavior or outliers in data. Unlike diagnosis prediction, this technique does not require labeled data as it looks for anomalies within existing data sets. Even though the training of anomaly detection model can be more computationally expensive and sensitive as opposed to to diagnoses prediction models, it can be used in a more general setting and requires less domain knowledge to correctly present explanations in the scope of XAI.

2 Anomaly Detection and Explainability in Deep Learning

ECG recordings involve extremely complicated patterns and are somewhat unique from person to person. This is why a powerful, high capacity model needs to be selected for processing them. A model that is able to work with very large data bases, noise and perform very heterogeneous pattern recognition. Deep learning models are an excellent choice for such a task since they scale well and are very flexible [6].

To perform anomaly detection an autoencoder (AE) is the most common class of deep learning architectures which works by compressing the input data into a low-dimensional latent space, then reconstructing the output from that latent space [7]. In this sense it is an unsupervised learning model that learns the multidimensional manifold on which our data is distributed and can be easily modified to assess whether a given data instance falls to that manifold or whether it is an outlier/anomaly. In this methodology, the AE is trained on normal data instances and learns to reconstruct only the normal patterns. When applied to a general data instance, the reconstructed output can be very similar to the input (data instance is normal) or significantly different from the input (data instance is likely to be anomalous). Such way of detecting anomalies is especially suitable for XAI because it does not only tell us whether an anomaly is present but also which part of the signal is anomalous which offers a way to visualize the anomaly in human-understandable way. See Fig. 1 for a depiction of AE concept for anomaly detection.

However, there are many flavors of AEs. The most basic version (shown in Fig. 1) consists of an encoder and a decoder, both with multiple layers of neurons. The input data is compressed by the encoder into a low-dimensional latent representation which is then reconstructed by the decoder. This vanilla AE sometimes suffers from

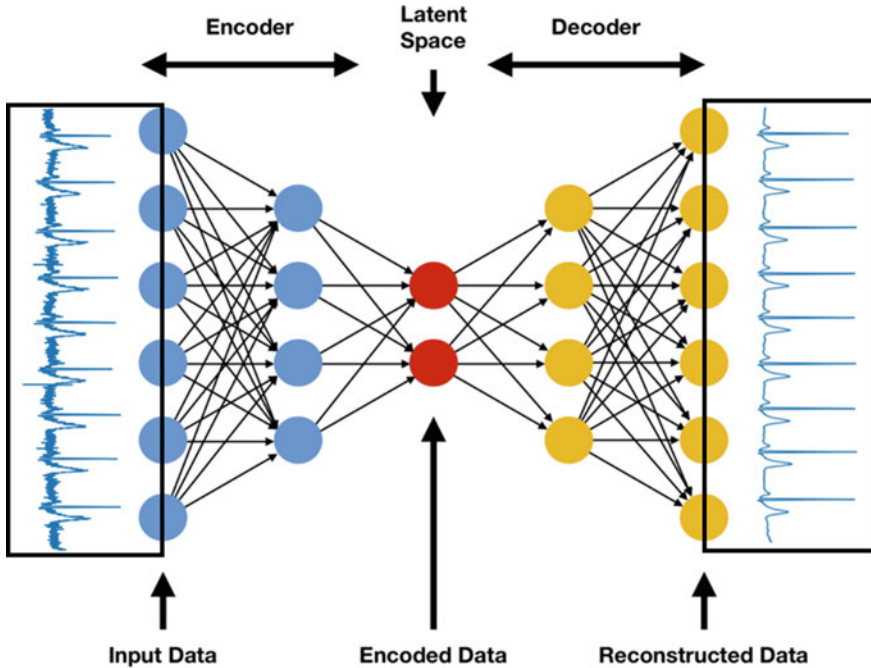


Fig. 1 Conceptual depiction of AE model when applied to ECG data. On the left AE receives a real-world ECG recording as an input and processes it so that each layer of the encoder decreases its dimensionality until a bottleneck is reached known as the latent space. If AE is trained properly small-dimensional version of the data instance encoded in the latent space includes enough relevant information to meaningfully reconstruct the ECG recording with a decoder part of AE using only the information from the latent space so that the output on the right is approximately equal to the input on the left

overfitting problems. These problems can be alleviated by using a sparse AE [8] where additional constraints are imposed on the weights in order to induce sparsity on the activations of the hidden layers during training. This helps ensure that only a few important features are encoded. Variational AE is another type which upgrades the AE into a generative model [9], meaning that it can generate new data samples that resemble the training data. It does this by learning a probability distribution over the inputs. Another type is a contractive AE [10]. This type of AE adds a regularization term to the loss function to enforce a contractive behavior on the hidden units. This helps make the model more robust to small changes in the input. Finally, there is a denoising AE (DAE) [11] which was found most useful in our work. Unlike other AEs, this type attempts to reconstruct the original input from a corrupted version of it. It turns out to be more robust to noise and have superior manifold learning capabilities. It can also be used for tasks such as removing noise from data as complex as images or text [12].

As stated before, AE can provide explanations for detecting anomalies by pinpointing regions of the ECG signal that was poorly reconstructed by the AE. Those regions are the reason that the signal was flagged as anomalous and we can also build a visual depiction for the locations of anomalies. This method of explainability, however, is not the only known method in deep learning. In recent years XAI advanced substantially, not only in deep learning but also in general machine learning. Widely applicable methods include Local Interpretable Model-Agnostic Explanations (LIME) [13] (which works by approximating model behavior locally around a prediction by creating simplified surrogate models), SHapley Additive exPlanation (SHAP) [14] (which uses game theory to explain the contribution of each feature to the model's predictions) and Anchors [15] (instance-level explanations that identify key features that lead to specific predictions). Some methods were, on the other hand, specifically designed for neural networks, such as Layer-wise relevance propagation (LRP) [16], and specifically for convolutional neural networks such as Grad-Cam method [17]. Despite the wide assortment of methods none of them was explicitly designed for time series data. Therefore, explainability for ECG processing models has been in literature performed by either adding additional explainable features designed by physicians [18], which is costly and error prone [19], or by employing methods originally designed for image data [20] which did not fully align with time series nature of ECGs [21].

3 Denoising Autoencoder as an Explainable Anomaly Detection Model for ECGs

DAE is a type of neural network used for unsupervised learning with strong connections to manifold learning. Suppose that the data set on which DAE is trained on lies on a low-dimensional manifold embedded in a full feature space of the data set. This embedded manifold can be learned by DAE by training an AE so that it can reconstruct input data that has been corrupted by noise, it denoises the instances. This means that DAE learns to project noisy data instance to a closest point on a manifold on which the full data set resides. Simply put, DAE is an AE where input data instances have added noise. The type of noise injected is typically Gaussian or salt and pepper noise (randomly sets some of the input values to one of its extreme values). In the specific case of ECG data Gaussian random walk noise was found to be useful as well. For illustration, Fig. 2 shows all three stages of DAE for ECGs, a real-world ECG input, the same ECG with injected noise, and the output ECG (i.e., the reconstructed, denoised ECG).

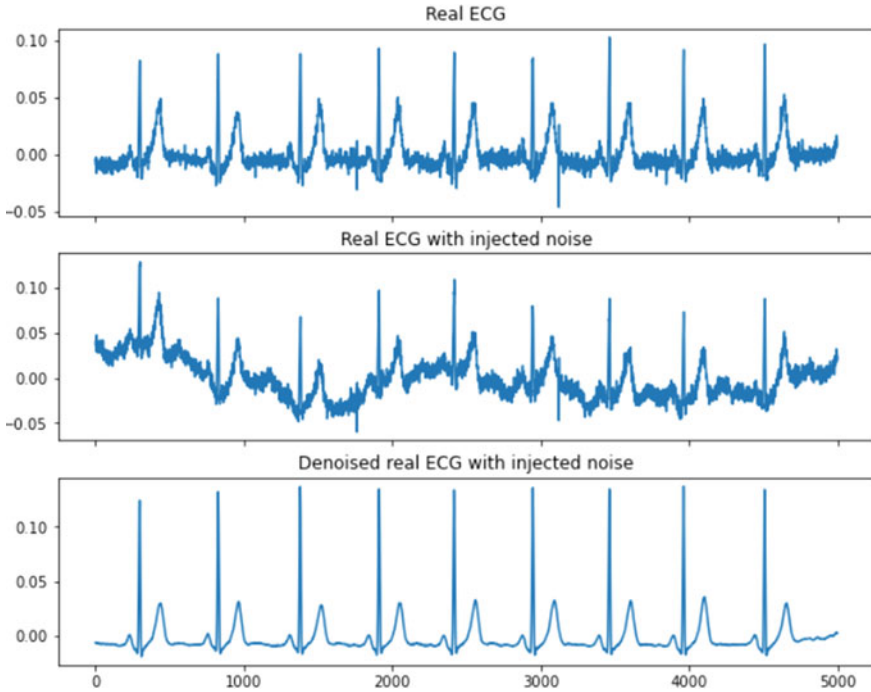


Fig. 2 How a real-world ECG recording changes when passed through a trained DAE. On top is a real-world ECG given as an input to DAE, in the middle is the same ECG but with injected noise and on the bottom is the output of DAE, a denoised ECG which is reconstructed from a low-dimensional latent space

3.1 ECG Data Sets

The data set collection used in this work was compiled for the purpose of The PhysioNet/Computing in Cardiology Challenge 2021 and includes the following data sets: CPSC Database and CPSC-Extra Database, INCART Database, PTB and PTB-XL Database, The Georgia 12-lead ECG Challenge (G12EC) Database, Augmented Undisclosed Database, Chapman-Shaoxing and Ningbo Database, The University of Michigan (UMich) Database. The most prevalent form of ECG recording in this collection is a 12-lead, 500Hz recording that lasts for 10s. There are minute amount of other types of ECG recordings present in the collection, e.g. with other frequencies, however, we decided to discard those in order to generate a data set with uniform properties. Each ECG recording is equipped with supplementary data including patient age, gender and diagnoses. The resulting data set we use includes 81,100 data instances of which 14,419 are pure sinus rhythm without any anomalies. There are 132 possible diagnoses in the collection and a given patient can have multiple of them.

3.2 Model Architecture and Training

To build an efficient DAE model we now need to select appropriate layers that will process ECG recordings to increasingly smaller dimension and back again. The choice of layers need to reflect the properties of the data. A single ECG data instance in our collection is a time series of 5,000 steps with each step containing a vector of 12 numerical values corresponding to the 12 ECG leads. Without considering any of the data properties we would build a DAE that gets 60,000 input numbers and transform them using dense layers (general matrix multiplication) to ever smaller dimension. However, such a model would be extremely large and computationally expensive to both use and train. It is, therefore, prudent to consider the fact that the ECG signal is a time series which means that our recordings are “continuous” in time. This means that we can reduce the dimensionality of our ECGs by contracting the number of steps in them and at the same time obtaining as much information as possible. In other words, in any given time point it is only important how the 12 leads crudely change in time and not what are the exact values that surround a given point. This leads us to use layers that act locally which means to use multiplication with a band matrix instead of the general dense matrix. If we also consider the fact that the way to process ECG should not depend on time, we are left with convolutional layers. It is important to note that recurrent and transformer layers are also well suited for time series processing, however, it is known that they are difficult to train when sequences are long which is especially true for our ECG recordings. Another drawback is that they offer less parallelism compared to convolutional layers which results to less opportunities for extensive training usually needed for (D)AE models.

We performed manual fine tuning of a DAE with convolutional layers with various properties such as kernel and filter sizes, number of layers, type of activations and so on. The most efficient model was DAE with encoder composed of 9 layers with filter and kernel sizes dropping from 64 to 4 and 16 to 4, respectively. To reduce the time dimensionality of our ECGs one can use either max-pooling or striding, however, we found that both perform similarly on our data. Because striding is more computationally efficient we used convolutional layers with stride equal to 2. To ensure well defined striding we zero-padded the input ECGs to the number of time steps equal to $5120 = 10 \cdot 2^9$. By applying 9 layers that reduce the number of time steps by factor of 2 we produced a latent space of shape (10, 4). Zero-padding also resolved edge artefacts that were present in former models.

Each convolutional layer is followed by a batch normalization and than a leaky rectified linear unit (ReLU) layer. We also tested models with skip connection with various topologies, however, contrary to the findings in literature, they did not improve the training in any way. Since skipping produces additional computational load and does not bring any benefits in our case we omit it from our end model. The decoder used is simply a mirror image of the encoder with convolution layers substituted for transposed convolution layers and zero-padding for cropping. For training we used absolute value metric and Adam algorithm. The model was trained for 20 epochs with batch size of 32 on a set of 11,535 (80% for training and 20% for testing) recordings

with pure sinus rhythm normalized with a constant factor of $1/2^{12}$. It is important to note that model accuracy was quite sensitive to the type and level of noise injected before the input layer. In our case the following noise was found the most useful:

$$\text{noise}_t = \frac{1}{\sqrt{t + 3000}} \left[\sum_{i=-3000}^t U(-0.1, 0.1) \right] + N(0, 0.003), \quad (1)$$

where $t \in \{1, \dots, 5000\}$ is a time step index, $U(a, b)$ is a variate drawn from uniform distribution on interval $[a, b]$ and $N(\mu, \sigma)$ a variate drawn from normal distribution with mean μ and standard deviation σ . The random walk part of the noise was found to be important, we suspect that this is due to its similarity to the noise actually observed in ECGs. The shift of 3000 steps has a role of burn-in that guarantees statistical independence.

3.3 Results of Denoising and the Exploration of the Latent Space

The trained DAE model is able to process a real-world ECG recording that may include noise (e.g., due to random body movements and respiration, power line and external electromagnetic interference) or medical anomalies and returns the closest approximation of the same ECG as if it would not have any noise or anomalies. The denoised ECG is reconstructed from only 40 floating point numbers, however, comparing it to the original ECG shows that it holds enough information and captures positions and shape of peaks really well together with patient specific artefacts. In Fig. 2, we show three examples of real-world ECG alongside the denoised, reconstructed ECG given as the output of our DEA. The height of the peaks does not align completely with the real-world ECG, however, we expect to improve that in future work. If medical anomalies are present in the ECG our DAE is unable to reconstruct them because DAE was not trained on such ECGs, it can only reconstruct normal ECGs which makes it a suitable anomaly detection model. In Fig. 2, we show two anomalous ECGs and how our DAE model attempts to reconstruct them as if they are normal ECGs. One anomaly type in Fig. 2 is localized while the other is not. We can observe in what way the differences between original and the reconstructed ECG manifest and how the explainability model put on top of our DAE can be implemented (Fig. 3).

We see that the 40-dimensional latent space can encode all the relevant features of the ECG which means that it can be used further as a compact encoding for ECGs. As far as we know from literature [22], latent space possesses special structure that is semantically meaningful, therefore, it is interesting to explore how ECGs in our data base are distributed in this 40-dimensional space. DAE reduces the raw ECG recording of shape (5, 000, 12) to an encoding of shape (10, 4) in the latent space.

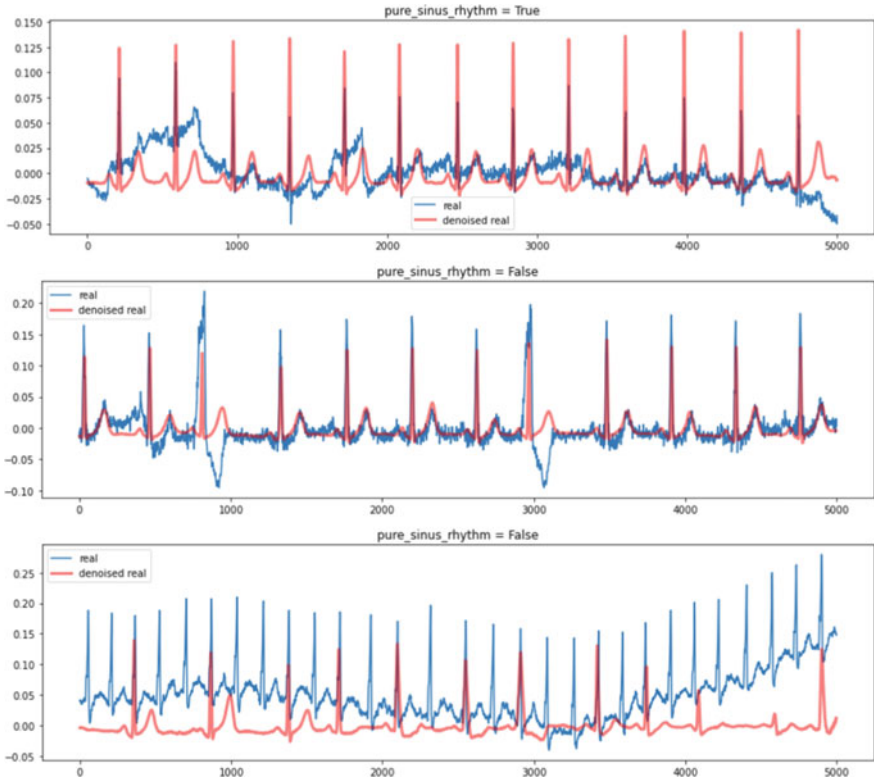


Fig. 3 Three examples of application of our DAE on real-world test ECGs. On top there is an ECG without anomalies with clearly observable natural noise, in the middle there is an ECG with two localized visually observable anomalies, and at the bottom an ECG with non-localized anomaly. All plots show only the first ECG lead

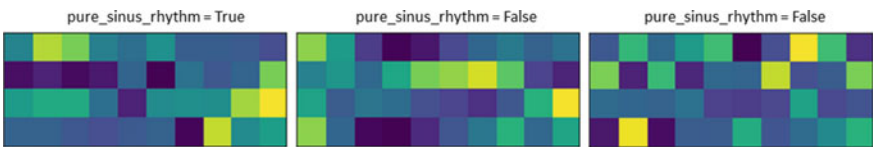


Fig. 4 Visual depiction of the same three examples as in Fig. 2 but as encoded in our 40-dimensional latent space in DAE. Instead of using original ECG of shape (5000, 12) we can represent them with a reduced shape of (10, 4). Even though the reduction is enormous this encoding holds all the relevant information to reasonably reconstruct the original ECG (minus the noise)

Figure 4 shows three examples of ECGs encoded in this space, however, it is difficult to see any evident structure.

What is informative is to study the distribution of the ECGs in our data base. Figure 5 shows a 2-dimensional projection of this distribution using UMAP method [23] which generates a scatter plot in such a way that the distances between the

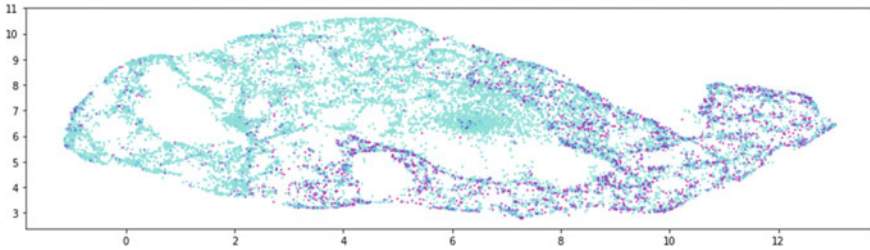


Fig. 5 2-dimensional depiction of the distribution of both anomalous (cyan) and non-anomalous (magenta) test ECGs as encoded in the 40-dimensional latent space of our DAE. We can observe partial separation between the two classes

points are as close as possible to the ones in full 40-dimensional latent space. What we can see is that normal ECGs are distributed in a special subregion of the space and partially overlaps with anomalous ECGs. In other words normal ECGs are a special kind of anomalous ECG, such as whiteout or an anomaly.

Given that the latent space encoding of an ECG holds enough information to reasonably reconstruct it, it can also be used for other ECG processing tasks. Instead of using raw ECG we could use this compact 40-dimensional encoding to perform, for example, parameter extraction and diagnosis prediction. To demonstrate this strength of the latent space encoding produced in our DAE we built a very simple classifier model that predicts whether an ECG is a pure sinus rhythm or not. The input to this simple model is solely the encoding as seen in Fig. 4. The simple classification model is a neural network with two dense hidden layers and returns a probability that the ECG is a pure sinus rhythm. The structure of the model can be seen in Fig. 6 alongside with the model performance.

4 Cloud-Based Service and Visualization of Explainable Anomaly Detection on ECGs

The denoising autoencoder model can now be used on any standard 12-lead 500 Hz ECG recording to denoise and detect anomalies. In order to maximise the widest possible use of this methodology we aspire to make it as user-friendly as possible. To this end we provide all the necessary software inside a docker image [24] and is ready to use without any other software requirements. This image includes several trained models, functions to apply the model and visualize the result alongside with all libraries that are required. We also provide a cloud-based service to perform ECG anomaly detection on a server which can be used without any programming knowledge [25]. This service uses FastAPI and applies our model to a desired ECG recording and returns a visualization of the ECG alongside with the annotations of regions where the model has detected anomalies. Currently the service operates on

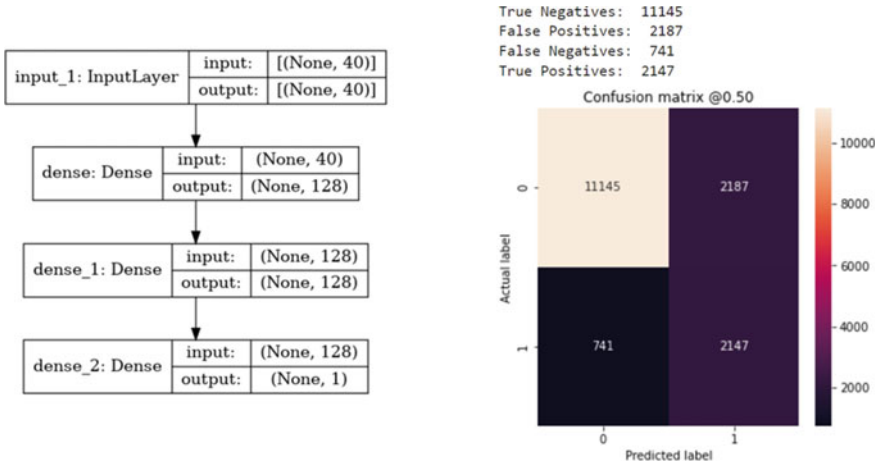


Fig. 6 The structure of a simple prediction model that takes the latent space encoding of ECG as input and returns the probability that the ECG is non-anomalous (left). A confusion matrix showing different classification metrics for such a simple model (right)

an ECG data base that is stored on the server, however, we intend to expand this to allow uploading of ECGs in FHIR compatible format in safe and secure way.

The visualization of the explainable anomaly detection is performed by first observing deviation between the real and the reconstructed ECG recording. We empirically found that the relevant anomalies exceed a threshold of 820mV. We color code the degree of deviation that exceeds this threshold and superimpose it alongside the original ECG recording. An example of such visualization is shown in Fig. 7. To reduce the noisiness of the visualization, which results from the noise present in the original ECG recording, we convolve the colors with the time window of 60ms which results to visually pleasing representation of the positions of anomalies present in the ECG.

5 Conclusion

In this work, we show that anomaly detection can be done using pure data-driven methodology without using expert knowledge and even on signals as complicated as ECG recordings. We found denoising autoencoder model to be the most effective for this task and it is able to compress 60,000-dimensional ECG to a mere 40-dimensional vector that holds all the relevant information including heart rate, respiratory rate, PR and QT interval, PR and ST segments, shape of QRS complex and other, even distinct features seen in individual patients. What is interesting is that this catalog of shapes that are statistically common in ECGs were extracted directly from the data. We found that this 40-dimensional encoding holds even an information about the

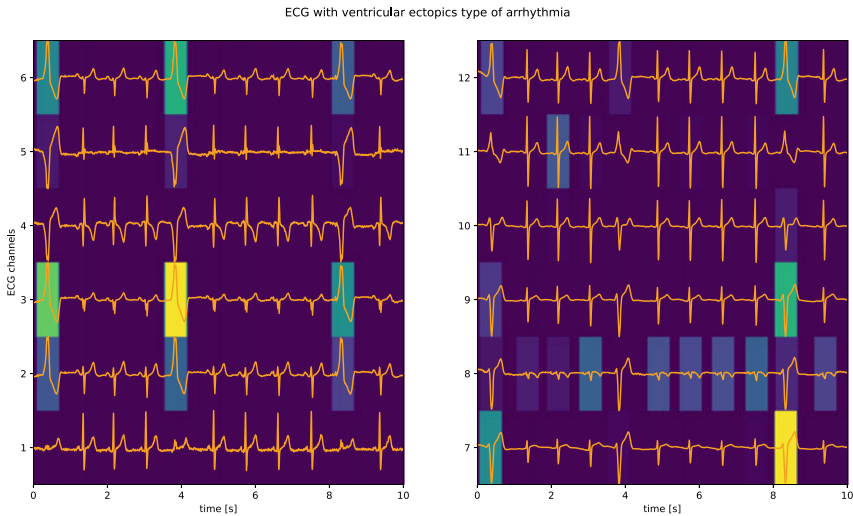


Fig. 7 A visualization of the result of explainable anomaly detection with denoising autoencoder on an example of a ECG recording with ventricular ectopics type of arrhythmia. The sections of ECG where the deviation between original and reconstructed ECG is large is color coded with yellow tones indicating large discrepancy and with blue tones low discrepancy. By this we can show the positions of medical anomalies in the ECG recording and provide not only whether the anomaly is present but also where to look for it in the recording

diagnoses and can be used to construct a simple prediction model. The signal can be reconstructed back from this 40 numbers, however, in a way that the reconstructed ECG does not include neither noise or medical anomalies present in the original ECG recording. We use this fact to construct an explainable anomaly detection model that can both tell if ECG includes anomalies and where exactly in the ECG they are positioned. To maximize the use of our methodology we provide a way to visualize the ECG with anomalies annotated using color codes and provide a user-friendly cloud-based service to perform it using the simplest possible hardware and software.

For future work it is important to further explore the possibilities for disentanglement of medical anomalies and anomalies that result from noise. One way to find more understanding is to stack two denoising autoencoders where each of them is trained in a way so that it can only reconstruct one type of anomaly. So that we have one model for removing noise and one for removing medical anomalies. Another possibility is to use semi-supervised learning and train a model that not only removes anomalies but at the same time predicts the diagnoses. This couples the representation in the latent space more tightly to the medical interpretation and possibly result to disentanglement of noise anomalies and medical anomalies in the latent space itself which can be studied due to its low dimension.

References

1. Rim, B., Sung, N.-J., Sedong, M., Hong, M.: Deep learning in physiological signal data: A survey. *Sensors* **20**(4), 969 (2020)
2. Singh, A., Sengupta, S., Lakshminarayanan, V.: Explainable deep learning models in medical image analysis. *J. Imaging* **6**(6), 52 (2020)
3. Gupta, R., Srivastava, D., Sahu, M., Tiwari, S., Ambasta, R.K., Kumar, P.: Artificial intelligence to deep learning: machine intelligence approach for drug discovery. *Mol. Divers.* **25**(3), 1315–1360 (2021)
4. Tomašić, I., Trobec, R.: Electrocardiographic systems with reduced numbers of leads-synthesis of the 12-lead ecg. *IEEE Rev. Biomed. Eng.* **7**, 126–142 (2013)
5. Miranda, D.F., Lobo, A.S., Walsh, B., Sandoval, Y., Smith, S.W.: New insights into the use of the 12-lead electrocardiogram for diagnosing acute myocardial infarction in the emergency department. *Can. J. Cardiol.* **34**(2), 132–145 (2018)
6. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*. MIT Press (2016)
7. Chen, Z., Yeo, C.K., Lee, B.S., Lau, C.T.: Autoencoder-based network anomaly detection. In: 2018 Wireless telecommunications symposium (WTS), pp. 1–5. IEEE (2018)
8. Ng, A., et al.: Sparse autoencoder. In: CS294A Lecture Notes, vol. 72, pp. 1–19 (2011)
9. An, J., Cho, S.: Variational autoencoder based anomaly detection using reconstruction probability. In: Special Lecture on IE, vol. 2, no. 1, pp. 1–18 (2015)
10. Rifai, S., Mesnil, G., Vincent, P., Muller, X., Bengio, Y., Dauphin, Y., Glorot, X.: Higher order contractive auto-encoder. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 645–660. Springer (2011)
11. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., Bottou, L.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* **11**(12) (2010)
12. Kim, J.-C., Chung, K.: Multi-modal stacked denoising autoencoder for handling missing data in healthcare big data. *IEEE Access* **8**, 104933–104943 (2020)
13. Ribeiro, M.T., Singh, S., Guestrin, C.: “Why should I trust you?” explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144 (2016)
14. Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*, vol. 30, pp. 4765–4774. Curran Associates, Inc. (2017)
15. Ribeiro, M.T., Singh, S., Guestrin, C.: Anchors: High-precision model-agnostic explanations. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
16. Montavon, G., Binder, A., Lapuschkin, S., Samek, W., Müller, K.-R.: Layer-wise relevance propagation: an overview. In: *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pp. 193–209 (2019)
17. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626 (2017)
18. Maweu, B.M., Dakshit, S., Shamsuddin, R., Prabhakaran, B.: Cefes: a cnn explainable framework for ecg signals. *Artif. Intell. Med.* **115**, 102059 (2021)
19. Jo, Y.-Y., Cho, Y., Lee, S.Y., Kwon, J., Kim, K.-H., Jeon, K.-H., Cho, S., Park, J., Oh, B.-H.: Explainable artificial intelligence to detect atrial fibrillation using electrocardiogram. *Int. J. Cardiol.* **328**, 104–110 (2021)
20. Ganeshkumar, M., Ravi, V., Sowmya, V., Gopalakrishnan, E.A., Soman, K.P.: Explainable deep learning-based approach for multilabel classification of electrocardiogram. *IEEE Trans. Eng. Manag.* (2021)
21. Raza, A., Tran, K.P., Koehl, L., Li, S.: Designing ecg monitoring healthcare system with federated transfer learning and explainable ai. *Knowl. Based Syst.* **236**, 107763 (2022)

22. Oh, M., Zhang, L.: Deepmicro: deep representation learning for disease prediction based on microbiome data. *Sci. Rep.* **10**(1), 6026 (2020)
23. McInnes, L., Healy, J., Melville, J.: Umap: uniform manifold approximation and projection for dimension reduction (2018). [arXiv:1802.03426](https://arxiv.org/abs/1802.03426)
24. Explainable ECG anomaly detection playground. <https://repo.ijs.si/hribarr/ecg-anomaly-detection>. Accessed 10 May 2023
25. Explainable ECG anomaly detection webapp. <https://repo.ijs.si/cs/insectt-webapp>. Accessed 10 May 2023

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Indoor Navigation with a Smartphone



Drago Torkar

Abstract This chapter presents a cost-effective system for indoor localization and navigation that does not require the use of satellite positioning or data communication networks. The system, implemented as a smartphone app, relies on QR codes that are pre-generated and attached to the walls inside the building. By utilizing the information from these codes and the smartphone's inertial motion unit (IMU) sensors processed by the Pedestrian Dead-Reckoning (PDR) algorithm, the user's current position can be determined. The Dijkstra navigation algorithm is then used to guide the user to the desired destination. The smartphone app can also be used as a healthcare logistics service in mass-casualty incidents for collecting and reporting georeferenced triage decisions to the cloud.

1 Introduction

The basic idea was to create a proof of concept demonstrating that the indoor navigation and localisation is possible using only passive tags. For this purpose, a smartphone navigation app was developed to be used in inner parts of the buildings and which can operate with no satellite positioning service available and no communication network present. The satellite navigation systems (GPS, GLONASS, Galileo, BeiDou, QZSS, IRNSS) inside buildings, at least on lower floors and cellars, in central parts, and away from windows, usually do not work, or their accuracy is very reduced due to a small number of visible satellites. The communication networks (WiFi, LTE, 5G...) might not be available in some circumstances such as catastrophic incidents, power reductions, or similar, which disables the localisation systems based on them.

The purpose of the smartphone app was twofold. First, to develop a reliable, simple-to-use, and cheap indoor navigation system that could be used in large buildings like hospitals, shopping malls, trade centres, fairs, etc. where no other positioning

D. Torkar (✉)

Computer Systems Department, Jožef Stefan Institute, Jamova cesta 39, SI-1000 Ljubljana, Slovenia

e-mail: drago.torkar@ijs.si

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_8

141

service is available. Second, to develop an indoor position reporting system that can be used in accidents and mass-casualty incidents for reporting triage decisions to the server. Both functionalities are based on the QR codes [1] holding all the information needed.

2 Encoding Information in QR

The QR codes must hold all the information needed for the navigation. Two types of QRs were created:

1. *Floor QR* codes placed at the entrance to each floor include:
 - A compressed floor plan presented as edge list (Fig. 1). A python script was created for automatic generation of such items from raster images of floor plans.
 - The navigation graph with coordinates of the destinations and navigation points with description labels of all destinations (Fig. 2).

The two items are separated from each other with the ‘|’ character. Since the maximal storage space within the QR is very limited (3 kB) the data were first encoded with the base64 encoder and then compressed with the gzip algorithm [2]. The generated QRs of this type are very dense thus slowing down the recognition process. In less than optimal lighting conditions, the captured image may not be robustly recognized by the decoding routine, despite being supported by the AI/ML

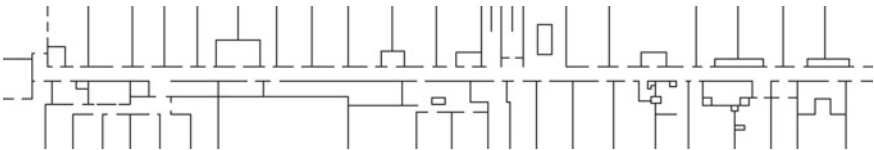


Fig. 1 A test floor plan saved in the floor QR after reconstruction

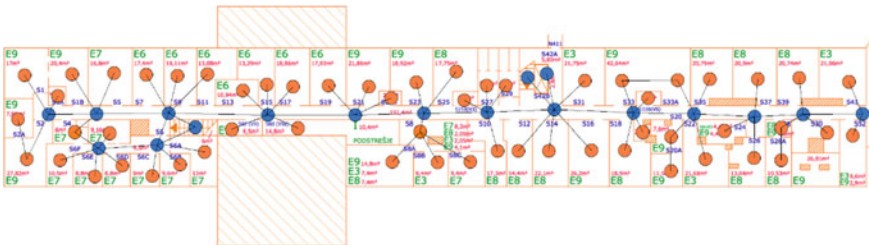


Fig. 2 Example of a navigation graph for a testing floor with all possible destinations (orange circles) and navigation points (blue circles) depicted on the original floor plan raster image

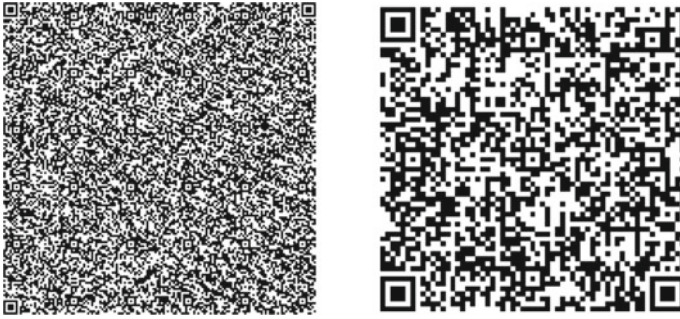


Fig. 3 Example of the floor QR (left) and location QR (right)

mechanism. Through our experiments, we gradually increased the size of the printed QR code from A5 to A3, with 2 sizes (one half and one full) for each format. We then tested each image under 3 different lighting conditions, ranging from poor to excellent artificial and natural light. Our findings suggest that using a full A3 paper size to provide a larger and clearer image for the decoding routine can significantly improve the recognition accuracy of the captured image.

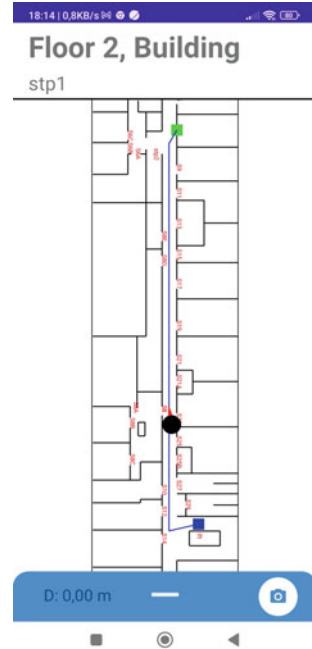
2. *Location QR* codes placed at destinations, intersections, and other navigation points which include the location in GPS (latitude, longitude) and in metric UTM (northing, easting) coordinates and some additional data about this location (room label, residents etc.). The data size is much smaller (3–4 times) compared to the floor QRs and the printed codes are not so dense. We conducted experiments following the same methodology used for the floor QR codes, and found that A4 or even A5 paper sizes are suitable for achieving robust recognition of the QR codes. Our findings suggest that smaller paper sizes can still provide accurate recognition results, making it possible to use the system in various settings where larger paper sizes may not be practical.

An example of both QR types is shown in Fig. 3.

3 Navigation

Navigation itself within the floor is performed by first scanning the floor QR, selecting the destination and then Dijkstra’s algorithm [3] calculates the shortest path to the destination which is then visualized on the floor plan (Fig. 4). The user’s current position is clearly marked and updated by the PDR algorithm [4] which determines steps converting them into distance according to chosen stride length, and direction of motion from IMU sensors. The smartphone orientation is also shown to facilitate determination of the user’s current orientation.

Fig. 4 Visualised path from the start (blue square) to the destination (green square) and user's current location and orientation (black circle with red pointer)



4 Local to Global Coordinates

The localization is done in global outdoor coordinates commonly referred to as GPS coordinates. This is desired for compatibility with publicly available maps and commercial outdoor satellite navigation systems like google maps. Since the current user's location is tracked by the smartphone IMU sensors and determined in meters by the PDR algorithm we needed to transform the local coordinates into global ones. Two approaches were considered:

- Inverse Transverse Mercator Projection (ITMP)
- Bilinear Interpolation (BI)

Transverse Mercator Projection (TMP) refers to the manner in which geographic coordinates are transformed into plane coordinates and basically projects a sphere or ellipsoid earth model to a cylinder which is then rolled out to a plane. In our case we needed to transform plane metric coordinates to geographic ones therefore the inverse TMP. This approach is known to be more accurate than other methods, but it is also more computationally demanding.

Fig. 5 Comparison of the two conversion methods from the local (metric) coordinates to GPS ones plotted on google maps. The green ones are calculated by the bilinear interpolation and the blue ones by ITMP



Bilinear interpolation is a method for interpolating functions of two variables using repeated linear interpolation. In our case, we consider latitude and longitude as functions over x and y , and we can calculate their values for each x and y from the known values in the vertices of the rectangle representing the floor. These known values can be read out from google maps. In the case of a non-rectangular floor shape, an apparent rectangle that encloses all destinations within the floor should be chosen. This approach is simple and less accurate than other methods, but it is also easier to implement.

Both approaches were tested and the results were very similar as depicted in Fig. 5. We first manually measured 52 locations on the test floor in a metric coordinate system aligned with the building orientation which we then rotated and scaled into UTM (Universal Transverse Mercator) coordinate system. We then made both conversions with this data. For the actual implementation, we then used BI since computations are much simpler compared to ITMP.

5 Triage

The app supports the following triage scenario. First responders locate each casualty, decide on the emergency level, and then using the app’s menu select the triage decision and send it together with the location in GPS coordinates to the server using REST API interface and GeoJSON data format (Fig. 6) [5].

In case the communication network is not available the data is saved to a local JSON based database and sent to the cloud later when the communication is restored which is reported to the user as shown in Fig. 7.

The data sent is encoded in GeoJSON data format for compatibility with the server-side software. An example of a red triage message (immediate, life-threatening status) is shown below.

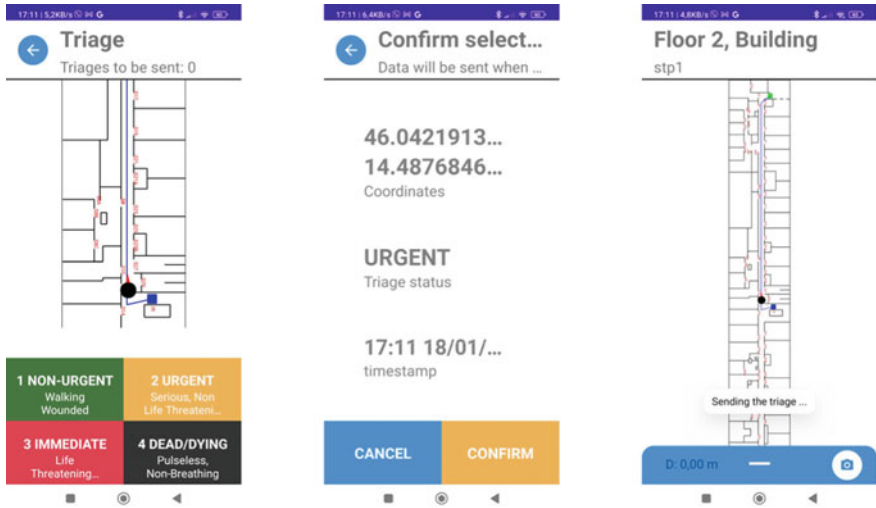


Fig. 6 The app’s screen for triage decisions (left), confirmation of the data to be sent (middle) and during sending them to the server (right)

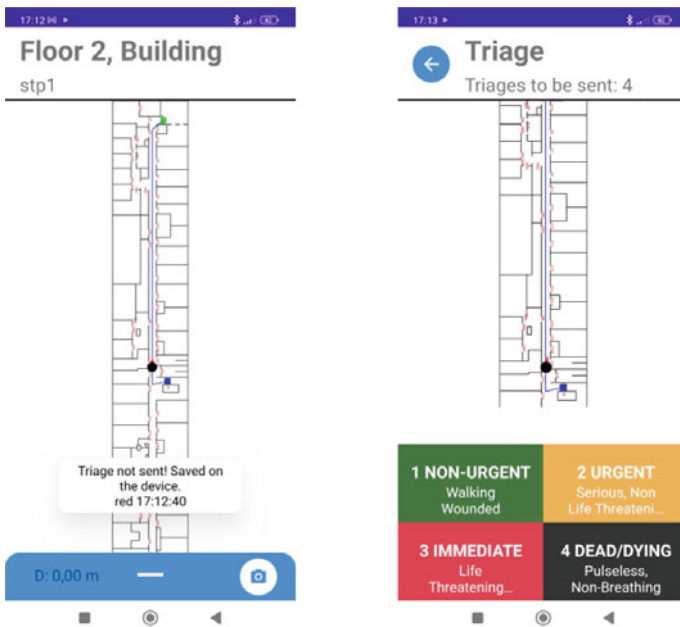


Fig. 7 If triage message was not successfully sent (left) the status of unsent messages is depicted on triage selection screen (right)

```
{
  {
    "id": 120,
    "type": "Feature",
    "geometry": {
      "type": "Point",
      "coordinates": [
        14.4876842,
        46.0421606
      ]
    },
    "properties": {
      "color": "red",
      "battery": "79%",
      "accuracy": 100,
      "client": "JSI",
      "dateTime": "20230123T205150Z"
    }
  }
}
```

6 Future Work

The described indoor navigation and localization system retrieves all the data needed from the QR codes which must be prepared and placed in appropriate locations in advance. The creation and placement of the codes demand a certain amount of time-consuming optimization due to the reduced storage space available. Also, the use of the system is not very user-friendly because users must locate the QRs without any help which is sometimes hard especially in catastrophic conditions when some QRs might be destroyed, occluded, or not accessible at all. In real life, the conditions when the communication network is destroyed or not available are very rare. The navigation itself was meant from the beginning to be used in normal daily circumstances. Therefore, it is reasonable to assume that users could access the navigation data for a certain complex of buildings from a designated server or cloud instead of from the QRs. This means that all the data needed is not size limited and can be transferred at once to a smartphone beforehand the user visits a desired building or complex. In this case, the organization of the data can be different. For example, all possible final destinations within a hospital are available in advance (not only when a particular floor is reached) and connected to a proper building, floor and wing. The navigation could be visualized and described in text even before the user reaches the entrance. Similarly, the starting point for indoor navigation (building entrance) can be determined from the smartphone's built-in satellite positioning technology which eliminates any scanning at the destination site.

7 Conclusions

We have developed an indoor navigation and localization system that is both simple and cost-effective, using QR codes and a smartphone as a proof of concept. One of the system's main advantages is that all necessary information is stored in the passive QR codes, requiring no additional information source. However, a potential drawback is that scanning and recognizing the QR codes can be cumbersome and time-consuming, particularly in emergency situations or under poor lighting conditions. To address the limitations of the system, we continue to explore potential improvements, including the integration of more advanced algorithms and technologies to enhance the scanning and recognition process. The availability of the navigation data on the cloud for use in normal daily conditions where communication networks are available is also considered. Additionally, we have developed and tested a triage support feature to be used in healthcare logistics services, demonstrating the versatility of the system beyond navigation and localization. Overall, this simple and inexpensive system shows promise for indoor navigation and localization, and with continued development, it has the potential to revolutionize indoor navigation and logistics.

References

1. QR Codes: What Are They And How Do They Work? <https://www.fastprint.co.uk/blog/quick-response-codes-what-are-they-and-how-do-they-work.html>. Accessed 22 Jan 2023
2. Umaria, M.M., Jethava, G.B.: Enhancing the data storage capacity in QR code using compression algorithm and achieving security and further data storage capacity improvement using multiplexing. In: 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, India, pp. 1094–1096 (2015). <https://doi.org/10.1109/CICN.2015.215>
3. Dijkstra, E.W.: A note on two problems in connexion with graphs. *Numerische Mathematik* **1**(1), 269–71 (1959). <https://doi.org/10.1007/BF01386390>. S2CID 123284777
4. Hou, X., Bergmann, J.: Pedestrian dead reckoning with wearable sensors: a systematic review. *IEEE Sens. J.* **21**(1), 143–152 (2021). <https://doi.org/10.1109/JSEN.2020.3014955>
5. van de Laar, F., Klabunde, K.: Location Awareness in HealthCare. InSecTT Book (2023)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Reconfigurable Antennas for Trustable Things



Mateusz Groth, Mateusz Rzymowski, Krzysztof Nyka, and Lukasz Kulas

Abstract In modern applications, the Internet of Things plays a significant role in increasing the productivity, effectiveness or safety and security of people and assets. Additionally, the reliability of Internet of Things components is crucial from the application point of view, where a resilient and low-latency network is needed. This can be achieved by utilizing reconfigurable antennas to enhance the capabilities of the wireless sensor network (WSN). Additionally, reconfigurable antennas can provide extended functionalities to the Internet of Things. One such aspect of wireless sensing is localization, where objects can be identified and positioned using radio frequency (RF) signals. For this purpose, analysis of spatial diversification of signals can be used by utilizing reconfigurable antennas. This work presents a design of a reconfigurable antenna that is applicable to Internet of Things WSNs and algorithms that utilize the antenna to provide additional localization functionalities.

1 Introduction

Many practical Internet of Things (IoT) applications rely on wireless sensor network (WSN) nodes placed within a target environment to monitor required parameters or to control devices used for specific tasks. Such nodes can work as standalone units, be integrated within different devices, or even be a part of larger embedded systems to provide communication means allowing for reliable and timely data gathering. From the technical perspective, the system's trustability will rely on the overall WSN nodes performance with a special focus on parameters like network resilience, bandwidth, throughput and latency, as well as its resistance to interferences and potential jamming attacks that contribute to general network performance [1]. To this end, WSN nodes in trustable systems should provide, compared to their predecessors, more capabilities at their physical (PHY) layer and, in consequence, also more information about the neighboring nodes [2–4]. Such extended capabilities can be provided by reconfigurable antennas that, contrary to those commonly used in

M. Groth (✉) · M. Rzymowski · K. Nyka · L. Kulas
Politechnika Gdańska, Gdansk, Poland
e-mail: mateusz.groth@pg.edu.pl

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_9

151

WSN nodes dipole or monopole antennas having omnidirectional radiation patterns, are able to modify or reshape its radiation pattern, e.g. to focus the antenna beam towards a specific direction, in order to improve the overall network performance [3–6].

To successfully integrate reconfigurable antennas within WSN nodes and apply them in real-life applications in different industrial domains, including maritime, automotive, critical infrastructure and healthcare, several following challenges, important from the industry perspective, have to be considered and taken into account in the final designs:

- **Simple steerability:** antenna radiation patterns have to be changed using simple I/O signals or other methods available in simple microcontrollers that are usually integrated within radio frequency (RF) transceivers.
- **Pattern diversity:** electronically steerable antenna should provide a wide diversity of available radiation patterns including omnidirectional, single beams pointing different directions, single beams with different gain and 3 dB beam width, as well as patterns having many beams and deep nulls to mitigate interferences and jamming attacks.
- **Available processing algorithms:** to unleash extended capabilities of WSN nodes integrated with reconfigurable antennas, dedicated algorithms able to provide, if possible, without any prior calibration, additional information about the direction-of-arrival (DoA) of incoming RF signals or localization of neighboring WSN nodes, have to be available.
- **Energy-efficiency:** antenna steering circuits have to rely on components that have low energy consumption.
- **Economic viability:** antenna design as well as components used in a steering circuit could be oriented on low costs and effortless integration with available RF transceivers or wireless devices.

In this chapter, we present electronically steerable parasitic array radiator (ESPAR) antenna, together with appropriate processing algorithms that satisfy all the above criteria.

2 Electronically Steerable Parasitic Array Radiator Antenna for Trustable Things

2.1 Concept

There are different types of reconfigurable antennas, and their complexity is directly related to their feeding and switching networks, ranging from the electromechanically steered antennas, through phased arrays and active arrays to switched parasitic antennas. In this chapter, we will focus on the last group in terms of energy efficient applications.

The simplest concept considers uniformly distributed quarter-wave wire monopoles as array elements. The elements couple with each other and their mutual dependencies are described by the following equation:

$$\mathbf{I} = (\mathbf{Z} + j\mathbf{X})^{-1}\mathbf{V} \quad (1)$$

where $\mathbf{I} = [i_0, i_1, i_2, \dots, i_N]^T$ is a vector of currents flowing through the elements of the array, \mathbf{Z} is an impedance matrix that contains the output impedance of the fed elements and impedances which characterize the level of mutual coupling between the array elements, \mathbf{X} is a diagonal matrix of the reactances applied to the elements and $\mathbf{V} = [v_0, v_1, v_2, \dots, v_N]^T$ is the excitation vector where the fed elements of the antenna are represented by values different from zero [7, 8]. The far-field radiation pattern E_{total} of the N-elements linear array can be written in the form:

$$E_{total} = E_{element}(\theta, \varphi) \times AF(\theta, \varphi) \quad (2)$$

where $E_{element}(\theta, \varphi)$ describes the radiation of a single element, while $AF(\theta, \varphi)$ is an array factor that describes the impact of all elements in the array and can be written as:

$$E_{total} = E_{element}(\theta, \varphi) \times AF(\theta, \varphi) \quad (3)$$

where $I_0, I_1, I_2, \dots, I_N$ are the current magnitudes in each element and $t_0, t_1, t_2, \dots, t_N$ are the distances between the first element and the N th one, k is the spatial frequency of a wave and u points the geometrical position of the array [7].

There are different types of electronically switched parasitic arrays that operate according to the principles described above and differ mainly in the number of active and passive elements in the array, as well as different switching approaches. More complex constructions with switched active monopoles are available [8] but the most known are ESPAR antennas with one active monopole that is surrounded by a defined number of passive elements. The reconfiguration of the radiation pattern is realized by attaching proper loads to the passive elements which can be done in different ways depending on the switching network realization. The initial realizations were based on varactor diodes [13] that require adequate voltage regulation and increase the complexity of the array. The approach discussed in detail in this work considers switching circuits that use RF switches or PIN diodes for simple ON/OFF switching where the parasitic element acts as shorted reflector or open director.

2.2 Design

The proposed ESPAR antenna (Fig. 1) design consists of 12 passive elements with one active monopole placed in the centre of the ground plane being a metalized top

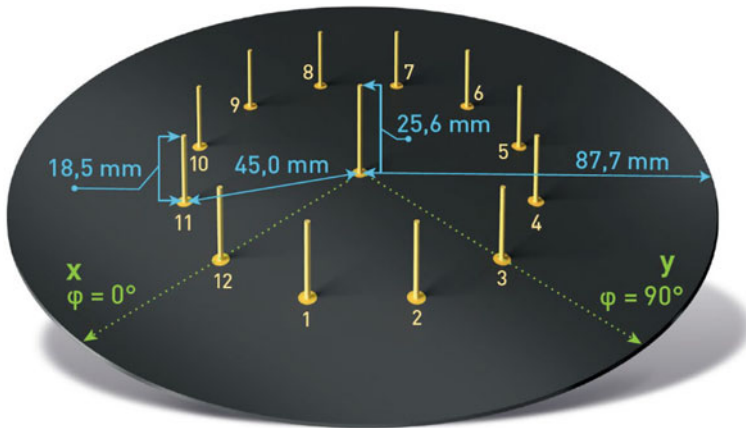


Fig. 1 ESPAR antenna design

layer of the printed circuit board (PCB) base. It was optimized to operate in the 2.4 GHz ISM band. A coaxial subminiature version A (SMA) connector is feeding the active monopole while the parasitic elements of the array are shortened to the ground or opened by the single-pole, single-throw FET switches located at the bottom layer of the antenna. This approach significantly improves energy efficiency in comparison to voltage-regulated varactors or PIN diodes [13]. When the passive monopole is shorted to the ground it functions as a reflector as it reflects energy while the open monopole is a director that passes through the electromagnetic waves. In result, a directional radiation pattern can be generated and rotated 360° with 30° discrete step by forming 12 different switches configurations.

The array design and simulations were conducted in Altair FEKO electromagnetic simulation environment. The design considers 1.55-mm-height FR4 laminate with a top-layer metallization and the main optimization goals were defined to achieve the compromise between possibly narrow main directional radiation pattern, low backward radiation level and acceptable impedance matching at the centre frequency 2.45 GHz.

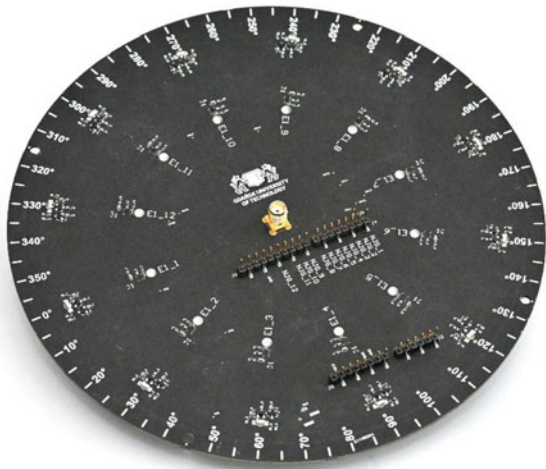
2.3 Realization

The realized ESPAR antenna is presented in Fig. 2. The array was fabricated on the 1.55 mm FR4 laminate and the monopoles were realized with 2 mm diameter silver plated copper rods. For the switching circuits realization NJG1681MD7 GaAs FET MMICs SPDT switches were selected. To provide a flexible exchange of different transceivers Arduino Shield headers have been placed on the bottom layer. Additionally, 12 LEDs are installed at the bottom of the PCB to display the status of each parasitic element.

Fig. 2 ESPAR antenna realization: **a** top view, **b** bottom view



a)



b)

Far-field radiation of the array was measured in the anechoic chamber as well as the input impedance matching. The radiation pattern in both horizontal and elevation planes is illustrated in Fig. 3. The half-power beamwidth (HPBW) is equal to 75° for horizontal and 65° for elevation planes respectively. The input impedance matching is below -10 dB in the whole considered frequency band as illustrated in Fig. 4. The results for the measured array are close to the simulated values.

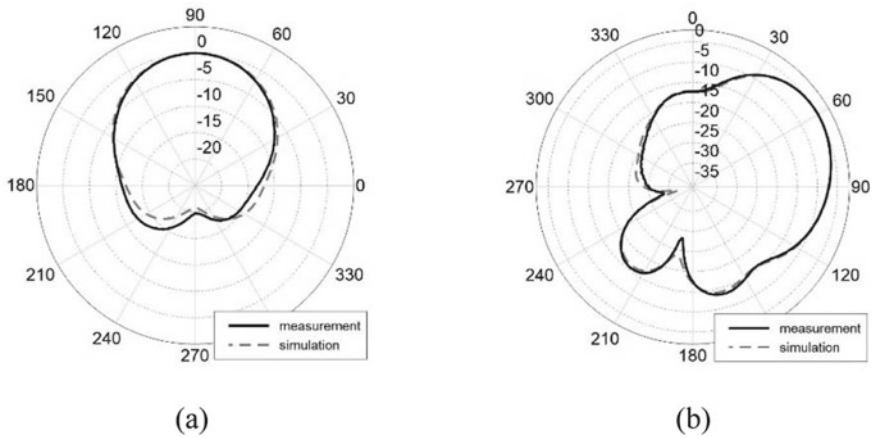


Fig. 3 Measured ESPAR antenna radiation pattern in **a** horizontal and **b** elevation plane

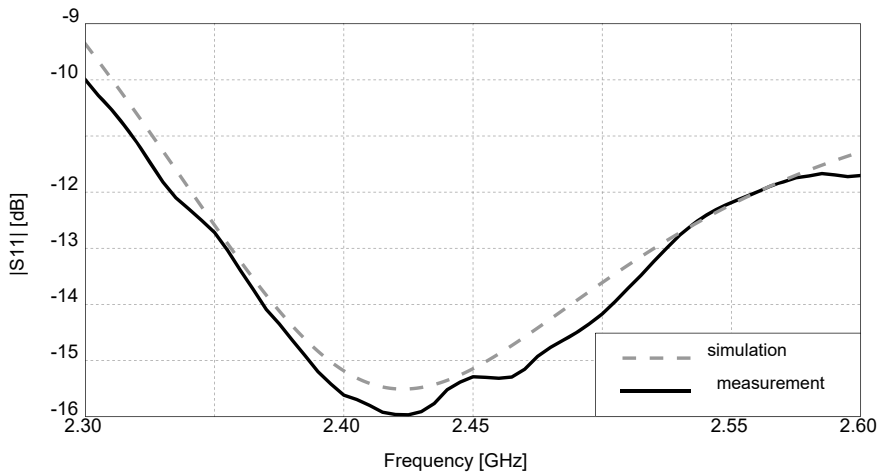


Fig. 4 ESPAR antenna input impedance matching

3 Applications

Switched beam antennas can be implemented in a number of potential applications providing useful functionalities both from the end user point of view as well as the system designer. By utilizing their directional capabilities, they can be fruitfully deployed in systems where spatial divergence of signal and its source can be utilized

either to enhance the communication or obtain and estimate additional information about the devices in range.

3.1 Direction of Arrival Estimation

One of the most crucial applications for the ESPAR antenna is the estimation of the direction of arrival of the received signal. Various methods can be used to estimate the DoA of the signal impinging the antenna [9–13]. From the IoT perspective, it is crucial for the algorithm to be implementable on cost-effective WSN sensors that are energy efficient. Thus, the estimation method should be performed with relatively non-complex operations and able to utilize parameters that can be obtained within a WSN node. Power pattern cross-correlation (PPCC) algorithm fulfils those requirements, as it uses only received signal strength and to establish the estimation, a relatively simple formula is used.

3.2 Power Pattern Cross-Correlation Algorithm

The algorithm utilizes cross-correlation coefficient between antenna radiation patterns, which are measured at the initial setup stage of the system, and the current received power measured for each antenna radiation pattern. The cross-correlation coefficient $\Gamma(\varphi)$ can be denoted as:

$$\Gamma(\varphi) = \frac{\sum_{n=1}^{12} (P(V_{max}^n, \varphi)Y(V_{max}^n))}{\sqrt{\sum_{n=1}^{12} P(V_{max}^n, \varphi)^2} \sqrt{\sum_{n=1}^{12} Y(V_{max}^n)^2}} \quad (4)$$

where φ is the azimuth plane $0^\circ \leq \varphi < 360^\circ$, $\{P(V_{max}^1, \varphi), \dots, P(V_{max}^n, \varphi), \dots, P(V_{max}^{12}, \varphi)\}$ are radiation patterns of the antenna, measured during the calibration stage for the corresponding steering vectors $\{V_{max}^1, \dots, V_{max}^n, \dots, V_{max}^{12}\}$ with the predefined angular step precision $\Delta\varphi$. Output power values measured at the antenna output for each radiation pattern are denoted by $\{Y(V_{max}^1), \dots, Y(V_{max}^n), \dots, Y(V_{max}^{12})\}$. Therefore, the estimated direction of arrival angle $\hat{\varphi}$ corresponds to the largest value of $\Gamma(\varphi)$ [13].

Since the antenna radiation patterns are measured with a discrete step $\Delta\varphi$ which determines the precision of the estimation, (usually $\Delta\varphi = 1^\circ$), the measured antenna radiation patterns $\{P(V_{max}^1, \varphi), \dots, P(V_{max}^n, \varphi), \dots, P(V_{max}^{12}, \varphi)\}$ can be represented as vectors $\{\mathbf{p}^1, \dots, \mathbf{p}^n, \dots, \mathbf{p}^{12}\}$ containing $I = 360$ measured discrete values $\mathbf{p}^n = [p_1^n, p_2^n, \dots, p_I^n]^T$. Thus, the Eq. (4) can be rewritten as [14]:

Table 1 DoA estimation errors for PPCC algorithm for various distances between the transmitter and the receiver [15]

Distance (m)	Estimation error			
	Mean	RMSE	Standard dev.	Precision
3	6.08°	7.91°	5.12°	17.00°
5	5.08°	6.58°	4.23°	16.00°
10	6.72°	9.47°	6.76°	36.00°

$$\mathbf{g} = \frac{\sum_{n=1}^{12} (\mathbf{p}^n Y(V_{max}^n))}{\sqrt{\sum_{n=1}^{12} (\mathbf{p}^n \circ \mathbf{p}^n)} \sqrt{\sum_{n=1}^{12} Y(V_{max}^n)^2}} \quad (5)$$

where the ‘ \circ ’ is the element-wise product of vectors, while $\mathbf{g} = [\Gamma(\varphi_1), \Gamma(\varphi_2), \dots, \Gamma(\varphi_I)]^T$ is a vector of length $I = 360$ containing discretized values of correlation coefficient $\Gamma(\varphi)$ for every value of φ in $\boldsymbol{\varphi} = [\varphi_1, \varphi_2, \dots, \varphi_I]^T$. The maximum value of \mathbf{g} corresponds with the estimated angle $\hat{\varphi}$. The PPCC algorithm in this formulation can easily be implemented within an IoT device to compute DoA estimation even when relatively low-efficient microcontrollers with simple transceiver modules for RSS packet measurements are used.

The hardware implementation of the algorithm was initially presented in [15], as a NXP JN5168 wireless microcontroller integrated with the ESPAR antenna on a single board. In [16], the construction of an ESPAR antenna with a directly connected custom-made nRF52840 module board was introduced. As presented in [15], the root mean square error (RMSE) of estimation for the distance between the transmitter and the ESPAR antenna of up to 10 m is less than 10° for measurements of 100 packets at each testing position (Table 1).

3.3 Interpolation-Based Estimation

As mentioned in the previous subsection, for the DoA estimation using the PPCC method, measurements of the antenna radiation patterns are necessary during a calibration phase. The calibration angular step $\Delta\varphi$ is linearly related to the precision of the algorithm, while very fine measurements might be time-consuming, especially when the system consists of a many ESPAR antennas. To reduce the time necessary for the calibration of the system, interpolation of radiation patterns has been introduced [14, 17]. To shorten the calibration phase, it is possible to measure the radiation pattern with a coarser step $\Delta\varphi > 1$ and then interpolate the radiation patterns \mathbf{p}^n obtaining as a result interpolated radiation patterns $\tilde{\mathbf{p}}^n$. Then, (5) can be rewritten as:

$$\mathbf{g} = \frac{\sum_{n=1}^{12} (\tilde{\mathbf{p}}^n Y(V_{max}^n))}{\sqrt{\sum_{n=1}^{12} (\tilde{\mathbf{p}}^n \circ \tilde{\mathbf{p}}^n)} \sqrt{\sum_{n=1}^{12} Y(V_{max}^n)^2}} \quad (6)$$

where $\tilde{\mathbf{g}} = [\Gamma(\tilde{\varphi}_1), \Gamma(\tilde{\varphi}_2), \dots, \Gamma(\tilde{\varphi}_I)]^T$ is a vector of length \tilde{I} of interpolated values of correlation coefficient $\Gamma(\varphi)$ for every value of φ in $\tilde{\boldsymbol{\varphi}} = [\varphi_1, \varphi_2, \dots, \varphi_I]^T$. As presented in [14], when linear interpolation is used, the direction of arrival estimation is still relatively accurate even when the number of measurement points is reduced from 360 to 8, reducing the calibration time by more than half. Furthermore, in [17] instead of linear interpolation, spline interpolated patterns were used. In this approach, the RMSE of estimation is further reduced giving good results even for 4 measurement points for every antenna radiation pattern.

3.4 Multiplane Calibration for 2D DoA Estimation

The accuracy of the DoA estimation highly relies on the diversity between different main beam directions. For the analysed ESPAR antenna design, the 3-dB beamwidth is 73.2° for horizontal and 61.8° for vertical planes. As described in the previous subsections, by switching between 12 radiation patterns the direction for maximum value in the horizontal plane can be shifted by 30° . Nevertheless, at the same time, the maximum value is still in the same direction. In [18], 2D DoA Estimation has been introduced, utilizing calibration not only for one horizontal plane $\theta = 90^\circ$ but in $10^\circ < \theta < 90^\circ$. In such an approach, estimator (4) can be rewritten into its 2D form as (Fig. 5):

$$\Gamma_{2D}(\theta, \varphi) = \frac{\sum_{n=1}^N (P(V_{max}^n, \theta, \varphi) Y(V_{max}^n))}{\sqrt{\sum_{n=1}^N P(V_{max}^n, \theta, \varphi)^2} \sqrt{\sum_{n=1}^{12} Y(V_{max}^n)^2}} \quad (7)$$

Considering that $\theta \in \langle 0^\circ, 90^\circ \rangle$ and $\varphi \in \langle 0^\circ, 360^\circ \rangle$. As a result of the measurements performed in an anechoic chamber, it was possible to estimate the direction of arrival of the signal in 2D with the mean error of 1.86° in horizontal plane and 4.31° in vertical plane for $\theta \in \langle 40^\circ, 90^\circ \rangle$.

3.5 DoA-Based Object Positioning

Number of position estimation techniques that base on RF signal analysis have been developed up to today. From the IoT perspective, it is important for the method to be relatively easy to implement and computation cost-efficient. Thus, algorithms that rely solely on RSS values are most interesting. To utilize benefits of DoA capabilities

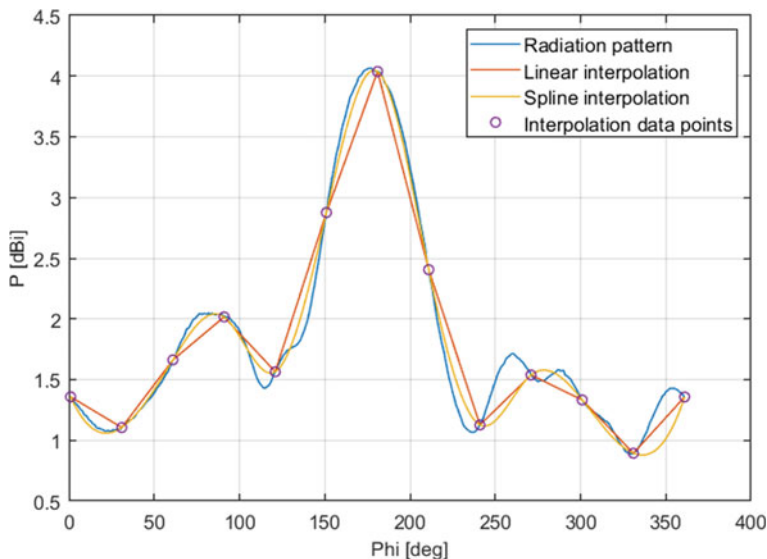


Fig. 5 Original radiation pattern and patterns interpolated using 12 data points

of ESPAR antenna, algorithms that estimate not only direction of arrival but also a position of signal source are researched, which require many WSN nodes and, thus, has limited applicability.

3.6 Single-Anchor Positioning System

One of the important factors of IoT indoor positioning systems is the deployment cost. Traditional RF-based positioning systems require a relatively large number of nodes to be installed in the area as the accuracy of the system strongly depends on the number of information sources [19]. In such cases, the simplest and most cost-effective solutions are systems that rely on proximity algorithms, in which the estimated position is based on the location of the node that receives the signal. For more accurate estimation, in most cases, the RF signal transmitted by a localized device has to be received by at least three nodes [20, 21]. On the other hand, the benefits of the DoA estimation capabilities of ESPAR antennas provide an opportunity to estimate the position based on information about the received signal measured with a single node. Such an approach allows reducing the complexity of the positioning system and decrease the time necessary for the deployment of the system.

Within the InSecTT project, a single-anchor positioning system that relies on the abovementioned PPCC algorithm was developed. In this approach, the position estimation process is twofold: first, the direction of the signal source is estimated using the PPCC algorithm. In the second step, the distance between the ESPAR antenna and

the localized object is calculated using the free-space loss formula which for most applications is a sufficient trade-off between accuracy and computational complexity when compared to other, more sophisticated methods of distance estimation for RF signals

$$\frac{P_r}{P_t} = D_r D_t \left(\frac{\lambda}{4\pi d} \right)^2 \tag{8}$$

where P_r and P_t are, respectively, power received and power transmitted, D_r is the directivity of the receiving antenna, D_t is the directivity of the transmitting antenna, λ is the wavelength of the signal and d is the distance between antennas. The formula can be easily transformed into distance calculation for the particular antenna radiation pattern:

$$d = \frac{\lambda}{4\pi} \sqrt{\frac{D_r D_t P_t}{P_r}} \tag{9}$$

The system consists of several gateway nodes based on nRF52480 equipped with ESPAR antenna which measures RSS values for each antenna radiation pattern and transmits the data to the localization service database using message queuing telemetry transport (MQTT) protocol. The service estimates the position which then is presented in the visual application. The architecture of the system is presented in Fig. 6.

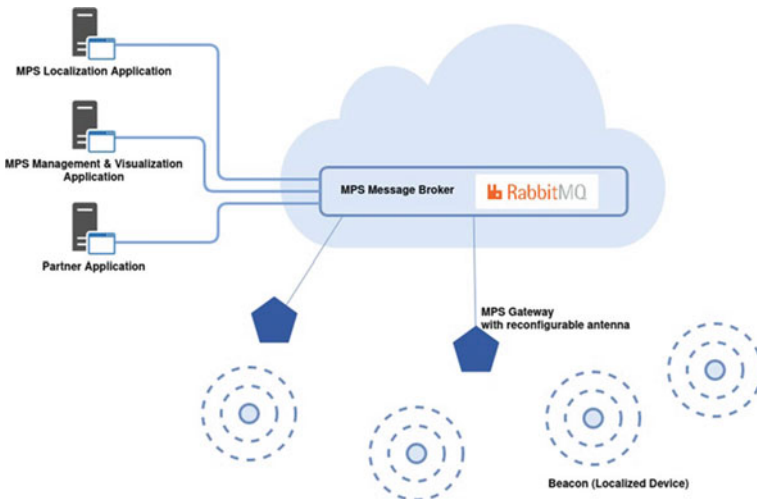


Fig. 6 Architecture diagram of multimodal positioning system (MPS)—an indoor positioning system that relies solely on the ESPAR antenna-based single-anchor approach



Fig. 7 Test installation of the system in St. Adalbert's Hospital in Gdansk, Poland. **a** ESPAR antenna installed on the ceiling, **b** localized device attached to a mobile asset (wheelchair)

The designed method was applied in the positioning system, whose working instance was deployed in the form of a demonstrator in St. Adalbert's Hospital in Gdansk, Poland. Within the demonstrated deployment, the system of 12 gateways has been installed and a number of test positioning modules have been mounted on the mobile assets of the hospital, according to the requirements defined by the hospital management (Fig. 7).

As a result, hospital employees can monitor in real-time the current position of necessary equipment, such as wheelchairs or hospital beds, increasing the efficiency of their work by minimizing the necessity of searching for assets needed to provide medical services. Additionally, the solution reduces the number of thefts as the system is able to monitor cases when objects are taken out of the building.

3.7 Calibration-Free Indoor Localization

The main drawback of the positioning method presented in the previous subsection is the necessity of the laborious calibration of each ESPAR antenna, which has to be performed after system installation and then periodically as the system environment as well as also propagation conditions and effects, may change over time. To address this challenge, a novel calibration-free algorithm has been proposed [16]. The method integrates fingerprinting [21] and trilateration [20] approaches combined

with auxiliary reference nodes installed in known positions. The role of the reference nodes is to decrease the negative influence of indoor RF propagation phenomena as well as radio map drift, which occurs in traditional fingerprinting-based positioning systems. Received signal strength (RSS) for each reference node is measured for each of 12 radiation patterns of ESPAR antenna together with RSS values of the localized device. The RSS vector of j th reference node can be denoted as:

$$\mathbf{V}_{ref_j} = [RSS_{ref_{j1}}, RSS_{ref_{j2}}, \dots, RSS_{ref_{j12}}] \quad (10)$$

where $RSS_{ref_{ji}}$ is the RSS value of j th reference node measured for the i th ESPAR radiation pattern. Similarly, the RSS values for the localized device can be denoted as:

$$\mathbf{V}_{loc} = [RSS_{loc1}, RSS_{loc2}, \dots, RSS_{loc12}] \quad (11)$$

where RSS_{loci} is the RSS value measured for the i th ESPAR radiation pattern.

In this approach, the localization is divided into two steps. First, the Euclidean distance D_j between \mathbf{V}_{ref_j} and \mathbf{V}_{loc} is calculated for each reference node:

$$D_j = \|\mathbf{V}_{loc} - \mathbf{V}_{ref_j}\| = \sqrt{\sum_{i=1}^{12} (RSS_{loci} - RSS_{ref_{ji}})^2} \quad (12)$$

We can assume that the j th reference node for which D_j has the lowest value is located in the closest vicinity of the localized node. Thus, by sorting D_j from the lowest to the highest value we arrange the corresponding reference nodes in growing order

$$D_j = \|\mathbf{V}_{loc} - \mathbf{V}_{ref_j}\| = \sqrt{\sum_{i=1}^{12} (RSS_{loci} - RSS_{ref_{ji}})^2} \quad (13)$$

where J is a total number of reference nodes and k_1, \dots, k_J are the indices of the distances. In the second step, K reference nodes for which the calculated distance is the lowest are chosen. The estimated position of localized device is calculated using the K -nearest neighbors method:

$$(x, y) = \left(\sum_{j=k_1}^{k_K} w_j x_j, \sum_{j=k_1}^{k_K} w_j y_j \right) \quad (14)$$

where (x_n, y_n) are the coordinated of j th reference nodes and w_j is the weight for j th reference node expressed as:

$$w_j = \frac{D_j}{\sum_{i=1}^K D_{k_i}} \quad (15)$$

Additionally, the normalization of RSS values was included to reduce the potential influence of hardware diversity and uneven distances between the ESPAR antenna and the nodes. For that, the RSS values have been rescaled:

$$RSS_{normref_j i} = \frac{RSS_{ref_j i} - \min(\mathbf{V}_{ref_j})}{\max(\mathbf{V}_{ref_j}) - \min(\mathbf{V}_{ref_j})} \quad (16)$$

$$RSS_{normloc_i} = \frac{RSS_{loc_i} - \min(\mathbf{V}_{loc})}{\max(\mathbf{V}_{loc}) - \min(\mathbf{V}_{loc})} \quad (17)$$

In consequence, the Euclidean distances can be denoted as:

$$D_{norm_j} = \sqrt{\sum_{i=1}^{12} \left(RSS_{normloc_i} - RSS_{normref_j i} \right)^2} \quad (18)$$

The proposed method has been verified in a 5.6 m × 6.6 m laboratory room using 24 reference nodes installed on the walls and one ESPAR antenna installed on the ceiling in the center of the room. During the test, measurements of RSS values in the reference nodes and the localized device for 12 ESPAR antenna radiation patterns and 90 test positions of the localized node have been taken.

As a result, the best accuracy was observed for $K = 3$ and 4 reference modules used for the algorithm. For such a configuration, a mean estimation error of 1.39 m can be achieved. Due to lack of laborious calibration process and acceptable estimation errors, this approach was well-received by St. Adalbert's Hospital management and personnel involved in the system maintenance. Therefore, the proposed calibration-free approach has been integrated within the system demonstrator in St. Adalbert's Hospital in Gdansk, Poland. Additionally, based on the results available in [22], which show that it is possible to reduce ESPAR antenna radiation patterns used in estimation from 12 down to 3 and still achieve comparable localization accuracy results, the whole system operation has been improved. Using fewer ESPAR antenna radiation patterns allowed to reduce time necessary to perform localization. In consequence, localization time of hospital's assets can be as low as 1 s. Therefore, the final system is capable to perform the positioning of devices that are in motion (Table 2).

3.8 Other Applications

Considering the beam switching feature of the ESPAR antenna, various applications related to the signal spatial diversity can be proposed for trustable devices. One of the approaches is to utilize the directivity aspect of radiation patterns to increase

Table 2 Estimation accuracy results for $K = 3$ and various layouts of reference modules [22]

Reference modules configuration	With normalization			Without normalization		
	Max. error [m]	Mean error [m]	RMSE [m]	Max. error [m]	Mean error [m]	RMSE [m]
Corners (4 modules)	3.95	1.63	1.82	4.09	1.82	2.05
Middle (4 modules)	3.58	1.39	1.58	3.68	1.97	2.14
Corners and middle (8 modules)	3.52	1.84	2.04	3.97	1.98	2.18
All (24 modules)	3.90	1.99	2.18	5.36	2.16	2.40

the connectivity performance, related to a signal-to-noise ratio (SNR). Taking into account that each of the ESPAR antenna passive element can be in one of two modes (shorted or open), it is possible to obtain 4096 antenna radiation patterns for a 12 passive elements antenna. By efficiently and adaptively switching between respective radiation patterns, one can increase the SNR to maintain above desired threshold in case of interferences or jamming attack. The same feature can also be used to minimize the radiation in an undesired direction, i.e., to spatially separate communications or to minimize the risk of eavesdropping.

References

1. Al-Karaki, J.N., Gawanmeh, A.: The optimal deployment, coverage, and connectivity problems in wireless sensor networks: revisited. *IEEE Access* **5**, 18051–18065 (2017). <https://doi.org/10.1109/ACCESS.2017.2740382>
2. Tran, T., An, M.K., Huynh, D.T.: Symmetric connectivity in WSNs equipped with multiple directional antennas. *Proc. Int. Conf. Comput. Netw. Commun. (ICNC)* 609–614 (2017).
3. Loh, T., Liu, K., Qin, F., Liu, H.: Assessment of the adaptive routing performance of a wireless sensor network using smart antennas. *IET Wirel. Sensor Syst.* **4**(4), 195–205 (2014)
4. Viani, F., Lizzi, L., Donelli, M., Pregolato, D., Oliveri, G., Massa, A.: Exploitation of parasitic smart antennas in wireless sensor networks. *J. Electromagn. Waves Appl.* **24**(7), 993–1003 (2010)
5. Skiani, E.D., Mitilineos, S.A., Thomopoulos, S.C.A.: A study of the performance of wireless sensor networks operating with smart antennas. *IEEE Antennas Propag. Mag.* **54**(3), 50–67 (2012)
6. Ademaj, F., Rzymowski, M., Bernhard, H.-P., Nyka, K., Kulas, L.: Relay-aided wireless sensor network discovery algorithm for dense industrial IoT utilizing ESPAR antennas. *IEEE Internet Things J.* **8**(22), 16653–16665 (2021)
7. Thiel, D.V., Smith, S.: *Switched Parasitic Antennas for Cellular Communications*. Artech House (2001)
8. Rzymowski, M., Nyka, K., Kulas, L.: Enhanced switched parasitic antenna with switched active monopoles for indoor positioning systems. In: 2014 20th International Conference on

- Microwaves, Radar and Wireless Communications (MIKON), Gdansk, Poland, pp. 1–4 (2014). <https://doi.org/10.1109/MIKON.2014.6899850>
9. Gyoda, K., Ohira, T.: Design of electronically steerable passive array radiator (ESPAR) antennas. *Proc. IEEE Antennas Propag. Symp.* **2**, 922–925. Salt Lake City
 10. Taillefer, E., Plapous, C., Cheng, J., Iigusa, K., Ohira, T.: Reactance domain MUSIC for ESPAR antennas (experiment). In: *Proceedings of IEEE Wireless Communications and Networking Conference*, vol. 1, pp. 98–102. New Orleans, LA
 11. Plapous, C., Cheng, J., Taillefer, E., Hirata, A., Ohira, T.: Reactance domain MUSIC algorithm for electronically steerable parasitic array radiator. *IEEE Trans. Antennas Propag.* **52**(12), 3257–3264 (2004)
 12. Taillefer, E., Hirata, A., Ohira, T.: Reactance-domain ESPRIT algorithm for a hexagonally shaped seven-element ESPAR antenna. *IEEE Trans. Antennas Propag.* **53**(11), 3486–3495 (2005)
 13. Taillefer, E., Hirata, A., Ohira, T.: Direction-of-arrival estimation using radiation power pattern with an ESPAR antenna. *IEEE Trans. Antennas Propag.* **53**(2), 678–684 (2005)
 14. Kulas, L.: RSS-based DoA estimation using ESPAR antennas and interpolated radiation patterns. *IEEE Antennas Wirel. Propag. Lett.* **17**(1), 25–28 (2018)
 15. Groth, M., Rzymowski, M., Nyka, K., Kulas, L.: ESPAR antenna-based WSN node With DoA estimation capability. *IEEE Access* **8**, 91435–91447 (2020). <https://doi.org/10.1109/ACCESS.2020.2994364>
 16. Groth, M., Nyka, K., Kulas, L.: Calibration-free single-anchor indoor localization using an ESPAR antenna. *Sensors* **21**(10), 3431 (2021). <https://doi.org/10.3390/s21103431>
 17. Groth, M., Nyka, K., Kulas, L.: RSS-based DoA estimation using ESPAR antenna radiation patterns spline interpolation. In: *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Nantes, France, pp. 1–5 (2018). <https://doi.org/10.1109/IPIN.2018.8533818>
 18. Kulas, L.: Simple 2-D direction-of-arrival estimation using an ESPAR antenna. *IEEE Antennas Wirel. Propag. Lett.* **16**, 2513–2516 (2017). <https://doi.org/10.1109/LAWP.2017.2728322>
 19. Soewito, B., Hassyr, F.A., Geri Arisandi, T.G.: A systematic literature review of indoor position system accuracy and implementation. In: *Proceedings of the 2018 International Conference on Applied Science and Technology (iCAST)*, pp. 358–362. IEEE, Manado, Indonesia (2018)
 20. Pakanon, N., Chamchoy, M., Supanakoon, P.: Study on accuracy of trilateration method for indoor positioning with BLE beacons. In: *Proceedings of the 2020 6th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, pp. 1–4. IEEE, Chiang Mai, Thailand (2020)
 21. Kaemarungsi, K.: Efficient design of indoor positioning systems based on location fingerprinting. In: *Proceedings of the 2005 International Conference on Wireless Networks, Communications and Mobile Computing*, Vol. 1, pp. 181–186. IEEE, Maui, HI, USA (2005)
 22. Groth, M., Nyka, K., Kulas, L.: Fast calibration-free single-anchor indoor localization based on limited number of ESPAR antenna radiation patterns. In: *17th European Conference on Antennas and Propagation*

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



AI-Enhanced Connection Management for Cellular Networks



Bernd-Ludwig Wenning

Abstract Cellular networks such as 4G, 5G and beyond are essential components of current and future communication infrastructure. Current research topics such as smart mobility, intelligent transportation systems and autonomous driving heavily depend on the availability of ubiquitous connectivity. In addition, consumer demand for cellular network resources is ever growing. In the context of the InSecTT project, cellular connectivity contributes, e.g., to use cases in the rail domain, where remote monitoring, maintenance and diagnostics rely on the presence of cellular uplinks from the rolling stock to cloud services. As the rolling stock moves, these uplinks are subject to continuously changing conditions that affect the availability and quality of the connection. To counteract this and to increase resilience, onboard systems may be equipped with a number of uplink options that can be used. This chapter discusses the use of AI to assess and manage the available connections and to move data traffic between the available links according to their expected data rates.

1 Introduction

Intelligent Transportation Systems (ITS) in the rail sector, as well as autonomous vehicles on the roads as they are envisaged for the near future, rely heavily on the availability and quality of mobile network connections. Applications and services such as tracking and monitoring of cargo, remote diagnostics of vehicle systems, collaborative navigation, security applications or value-added passenger services have to be able to operate continuously while the vehicle is on the move. Some of these applications are expected to mainly use the downlink, such as infotainment services for passengers, while others are bidirectional or mainly use the uplink, such as monitoring and diagnostic services or CCTV. A vehicle's onboard systems may have multiple connectivity options through commercial networks, e.g., fourth generation (4G) or fifth generation (5G) mobile networks for this purpose. Generally, these networks are managed by a commercial network operator, and the onboard

B.-L. Wenning (✉)
Munster Technological University, Bishopstown, Cork, Ireland
e-mail: berndludwig.wenning@mtu.ie

© The Author(s) 2024
M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_10

systems, as clients in those networks, will have little control over the allocation of resources by the network operator. To make the best use of the available options, the onboard system therefore has to continuously monitor the available connections, assess them, and route its traffic accordingly.

In the scope of this research, Artificial Intelligence (AI) including Machine Learning (ML) methods are developed for two purposes: First, the estimation of available uplink resources (in particular achievable data rates) and second, the selection of connections based on these estimations. To avoid active throughput measurements which create additional load on the network and may lead to additional costs, readily available channel parameters are used as input values to an AI based estimation of the available data rates. The estimation results are then supplied to a decision making agent that directs the traffic to the most favourable connection.

2 Related Work

2.1 Data Rate Estimation

The traditional approach to data rate determination is active sendprobing, i.e., test traffic is pushed onto the connection and the queueing delays are measured. Curve fitting approaches are then applied to the measured delays to determine the data rate. There is some variety of methods following this approach, e.g., [1–4]. These can in principle be applied to any kind of network interfaces, so would be applicable to 5G as well. Their downside is that they require additional test traffic, thereby contributing to network congestion.

Passive approaches aim to avoid the disadvantages of sendprobing, i.e., the necessity of sending test traffic and delays that are introduced by it, generally at the expense of lower accuracy as they rely on assumptions about the existing traffic or on relationships between channel information and data rate that have to be modeled. Those passive approaches may rely on monitoring existing traffic, e.g., [5], or they use lower level channel information and train a machine learning system with it. Most of the latter, such as the Adaptive Similarity-based Regressor (ASR) approach presented in [6] or a sniffer-based method presented in [7], are designed for 4G networks. Sliwa et al. [8] presents a passive approach for 5G, using a combination of network context (which includes, among others, the same parameters as used in this research), mobility context and application context. A selection of different machine learning methods is used on these inputs and compared for performance evaluation.

As already mentioned, several of these approaches, in particular the active ones, require additional communication traffic, either for probing the channel or for communication with central or cloud-based entities that assist the process. This results in unwanted additional congestion in the network as well as additional costs for the subscriber, especially on mobile connections that may be billed by traffic volume. In addition, it takes time to actively measure the channel, so active measurements

introduce delays. Other approaches require access to parameters that are not normally available to the user equipment, limiting applicability to equipment that does provide the parameters.

The approach presented in [8] addresses some of the problems, but does not discriminate between cases where all parameters are available and cases where some are unavailable, which will be shown to be a relevant issue in the scope of this article. This creates a dependency on the characteristics of the input data, potentially favouring situations that are similar to those that have been used for training. For example, if the training data included mostly situations where 5G parameters were available, estimation may fail to properly address cases where they are not.

2.2 Interface Selection

Interface selection in the presence of multiple options has been an area of research since the early 2000s. Early approaches such as those presented in [9, 10] or [11] usually rely on the availability of many parameters such as packet delay, packet jitter, etc. in order to make a decision. Many of these parameters are not available from device drivers by default or would require regular active measurements, which come with the same downsides that were already mentioned above for active data rate estimation methods. Other approaches such as [12] which presents an extended attractor selection model (EASM) require the involvement of the network infrastructure, i.e., the base stations, so are unsuitable for client-side only setups. The amount of information that is broadcast in this approach also adds a substantial amount to the network load and makes this approach impractical.

Centralized or cloud-assisted schemes are presented in [13, 14]. While they are offloading computational effort from the device to a centralized entity, they create a potential single point of failure, and they again introduce additional overhead on the communication interfaces. Niyato and Hossain [14] also includes a decentralized approach based on Q -learning, but this approach is unable to handle unexpected network congestion.

In [15], the Modified Linear Upper Confidence Bound (ModLinUCB) was presented as an approach based on the concept of bandit algorithms. Bandit algorithms use an analogy to slot machines in the gambling industry: An input is provided, then an “arm” is pulled (i.e., a decision is made), and a reward is received. In ModLinUCB, two interfaces are modelled as two arms of a bandit algorithm. The reward for the use of each arm is the achieved data rate, which is predicted from the channel quality parameters used as input, using a ridge regression. ModLinUCB was developed to decide between a 3G and a 4G interface, hence parameters that are specific to these technologies and readily available from interface drivers were chosen as inputs. In the 3G case, these were Received Signal Strength Indicator (RSSI) and the chip to interference power ratio E_c/I_0 . For 4G, Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ), Signal to Noise Ratio (SNR) and RSSI were chosen. All the parameters are scaled to [0, 1] intervals before they

are used in the ridge regression. A key component in ModLinUCB was a parameter called “additional confidence” that was added to the interface selection policy to address unexpected network congestion. If the actual throughput is much lower than expected, this parameter modifies the confidence related to the affected interface to encourage the algorithm to explore the other interface and switch if a sufficient data rate is expected on that other interface.

Further development lead to the Multi-Armed Bandit Adaptive Similarity-based Regressor (MABASR) [16], which, while maintaining the overall concept introduced with ModLinUCB, replaced the ridge regression by an Adaptive Similarity-based Regressor (ASR), which includes a smart strategy to learn and forget data points. This enables the algorithm to preserve important model knowledge. MABASR was taken a step further in [17], extending the concept to more than two interfaces and adding 5G as additional cellular technology.

3 Use Case and Research Challenge

As already mentioned in the introduction, this research is set in a vehicular context. More specifically, the InSecTT use case that provides the scope is a rail use case. Onboard systems on a train are envisaged to upload data to cloud services for remote monitoring and surveillance. These data come, e.g., from sensors or other onboard equipment such as CCTV cameras. The train is equipped with an onboard gateway that features multiple cellular uplink options. When the train is travelling, it will encounter changing conditions on any cellular uplink as it moves across network cells. Earlier work [6] has provided an approach to estimate the uplink data rate based on downlink parameters in 4G, and to use these estimates as a basis for an interface selection approach in [16].

In the context of 5G Non-Standalone (NSA) networks, the uplink data rate estimation on the User Equipment (UE) faces some challenges, partially already known from the 4G case and partially specific to 5G NSA:

- The UE generally does not have any information about the uplink channel quality parameters. This information is only available at the base stations and is not signalled to the UE. This was already a challenge in 4G networks and is the same for 5G.
- 5G NSA, which is the main type of 5G networks that is currently in use by commercial network operators, is a combination of 4G and 5G technology. This means the 5G network does not exist on its own, but it relies on underlying 4G infrastructure. While a 4G link is maintained continuously, the 5G carrier is only used if the link is active, i.e., if there is data traffic on the interface. Essentially, when a 5G connection is established, the device is connected to 4G and 5G infrastructure at the same time, while the two are operating on different frequency bands. In that case, both may be relevant for the achievable data rate. If 5G coverage is lost, only the 4G link is being used. Hence there are two different cases where only the 4G

link is active: One where there is no 5G coverage, and the other when the UE is not transmitting.

- If the 5G link is not active for either reason given above, 5G channel quality parameters are also unavailable. Therefore, taking the earlier approach that was developed for 4G and applying it to 5G NSA does not address the specific issues outlined here and will likely fail when the 5G parameters are unavailable.

4 Data Collection and Analysis

As a basis for the uplink data rate estimation, data had to be collected and the relationship between channel parameters and the available data rate had to be assessed. For this purpose, a TRX R6 rail certified gateway computer that was provided by project partner KLAS was used. The gateway computer, shown in Fig. 1, was equipped with a Telit FN980m 5G modem and a Subscriber Identity Module (SIM) from a commercial operator. In addition, the gateway featured two Sierra Wireless MC7455 modems for 3G and 4G, each also with their own SIM from a commercial operator. UDP traffic was generated with IPerf with a maximum data rate of 100 MBit/s on the uplink and the actual amount of traffic sent through the 5G modem was recorded at intervals of approximately 3 s. In the same intervals, queries were made to the modem for the 4G parameters RSRP, RSRQ and RSSI as well as the 5G parameters New Radio RSRP (NR_RSRP), New Radio RSRQ (NR_RSRQ) and New Radio RSSI (NR_RSSI), which were all accessible via AT commands. With this setup, data was collected driving a car through Cork City on different routes on different days in December 2021 and August 2022, collecting three sets of measurement data. The first December 2021 set contains 350 samples, the second December 2021 set contains 604 samples, the August 2022 set contains 894 samples. In all cases, the routes crossed areas where 5G coverage is present, as well as areas where only 4G is available. Significant geographical overlaps between the routes were avoided to ensure independence between the data sets. The recorded data was saved to CSV files.¹

In order to assess whether the 4G and 5G channel parameters have sufficient relationship with the achievable data rate, a correlation analysis similar to the one presented for 4G in [16] was done. The following results of the 5G correlation analysis were first presented in [17] and are repeated in this chapter as a foundation to the subsequent work on data rate estimation.

It was expected that the 5G case is more challenging than the previously investigated 3G and 4G cases, due to the specifics of 5G NSA mentioned in Sect. 3: The 5G carrier is only used if the link is active, i.e., if there is data traffic on the interface. If either 5G coverage is unavailable or no data is being sent, only the 4G parameters are available. When a 5G connection is established, the device is connected to 4G

¹The recorded data has been published in a Github repository: <https://github.com/MTU-Insectt/Measurements5G>.



Fig. 1 KLAS TRX R6 with cellular modems. The expansion slot in the middle with the four short antennas contains the 5G modem. The left slot with the larger antennas contains modems for 3G and 4G

and 5G infrastructure at the same time, while the two are operating on different frequency bands. If 5G coverage is unavailable, the interface falls back to 4G for data transmission, which then obviously uses the frequency band on which the 4G link is established. All of this is expected to lead to a more complex relationship between the measured parameters and the data rate.

Hence, it first has to be established whether there is a correlation between the measured channel parameters and the uplink data rate. For this correlation analysis, the two data sets from December 2021 were concatenated into a joint set. On this joint set, the Pearson correlation coefficients were determined among all the parameters and the data rate.

Table 1 shows the correlation matrix for the full concatenated set, which includes 954 samples. Some of the samples are with and others without 5G coverage. In case of no 5G coverage, the traffic is sent through the 4G link. While there is a moderate to strong correlation among the 4G parameters and the data rate, the correlation between the 5G parameters and the data rate is poor, as is the correlation between 4G and 5G parameters. However, the poor correlation can be explained by the fact that 257 out of the 954 samples, or 27%, recorded zeros for the 5G parameters, meaning that the device was out of 5G coverage (as iPerf traffic was pushed to the interface continuously, the 5G link did not turn off due to lack of traffic). The zeros

Table 1 Correlation matrix for the full concatenated data set

	RSRP	RSRQ	RSSI	NR_RSRP	NR_RSRQ	NR_RSSI	Rate
RSRP	1	0.72	0.93	-0.14	-0.16	-0.13	0.67
RSRQ	0.72	1	0.54	-0.07	-0.05	-0.08	0.54
RSSI	0.93	0.54	1	-0.08	-0.14	-0.07	0.60
NR_RSRP	-0.14	-0.07	-0.08	1	0.93	0.99	-0.05
NR_RSRQ	-0.16	-0.05	-0.14	0.93	1	0.89	-0.06
NR_RSSI	-0.13	-0.08	-0.07	0.99	0.89	1	-0.04
Rate	0.67	0.54	0.60	-0.05	-0.06	-0.04	1

Table 2 Correlation matrix only based on the samples where 5G is active

	RSRP	RSRQ	RSSI	NR_RSRP	NR_RSRQ	NR_RSSI	Rate
RSRP	1	0.58	0.92	0.28	0.06	0.31	0.49
RSRQ	0.58	1	0.38	0.14	0.16	0.13	0.34
RSSI	0.92	0.38	1	0.20	-0.08	0.26	0.42
NR_RSRP	0.28	0.14	0.20	1	0.53	0.88	0.57
NR_RSRQ	0.06	0.16	-0.08	0.53	1	0.19	0.24
NR_RSSI	0.31	0.13	0.26	0.88	0.19	1	0.56
Rate	0.49	0.34	0.42	0.57	0.24	0.56	1

are obviously uncorrelated to any of the other values and therefore have a negative impact on any correlation between the 5G parameters and the data rate. The 4G parameters however are continuously available, and even though data traffic is using the 4G link when 5G is unavailable and is using the 5G link when available, the 4G parameters seem to be a useful indicator for the achievable data rate throughout. For a more complete understanding of whether one can still also make use of the 5G parameters, it makes sense to have a further look at the cases where the 5G parameters are nonzero.

The second correlation matrix shown in Table 2 is only based on the 697 samples that contain nonzero 5G parameters. The correlation between the 4G samples and the data rate has now reduced compared to the previous table, but it remains significant. The correlation between the 5G parameters and the data rate however has increased, with the NR_RSRP and NR_RSSI parameters in particular exhibiting a moderate positive correlation. This suggests that the 5G parameters can be useful for data rate estimation as well.

5 Uplink Data Rate Estimation

For uplink data rate estimation in 4G, the Adaptive Similarity-based Regressor (ASR) [6] was presented. ASR uses a similarity-based approach to train an online Support-Vector Regression (SVR) for data rate estimation. A set of channel parameters is

compressed by Principal Component Analysis (PCA), and the compressed value is compared to previous samples. If it is similar to a known sample, it replaces the sample. If not, a generalization is applied where existing samples are forgotten temporarily in an iterative process, and the impact on estimation accuracy is assessed, ensuring that the sample which contributes least to the estimation accuracy is discarded permanently.

A straightforward approach for 5G could be to use the same, and add the 5G parameters to the data set before compression. However, as the correlation analysis has confirmed, the 5G parameters are obviously only correlated to the data rate when they are nonzero, i.e., 5G coverage is there and the link is active.

Therefore, there are several options how to determine the available uplink data rate in 5G networks, particularly 5G NSA networks with the specific challenges mentioned in Sect. 3. As stated in that section, both 4G and 5G channel conditions are potentially relevant for the estimation of available uplink data rates.

Hence, the original ASR approach can be taken further: Depending on whether only 4G parameters or 4G and 5G parameters are available, two separate learning algorithms that run in parallel can be used. This article proposes the following approach:

- Firstly, a new sample of channel parameters is acquired. Depending on whether the 5G link is active or not, it either includes valid values for 4G parameters only or for both 4G and 5G parameters.
- If the sample contains valid parameters for 4G only, it is sent to one learning algorithm (named Algorithm A in the further course of this document), otherwise it is sent to the other (Algorithm B).
- Algorithm A uses PCA to compress only the 4G parameters, whereas for Algorithm B, there are two possible variants: use PCA to either compress both the 4G and 5G parameters or compress only the 5G parameters. In either case, this reduces the dimensionality of the data to reduce the computational complexity.
- From there onwards, both Algorithm A and Algorithm B work essentially the same and follow the ASR approach:
 - Based on a similarity score calculated between the new sample and the most similar one already known to the algorithm, learning and forgetting is based on two different methods:

If the similarity score exceeds a given threshold, the most similar old sample is replaced by the new sample and the old sample is forgotten.

If there is no old sample that is similar enough, samples are iteratively forgotten, and generalization is measured at each iteration. The generalization ability is implemented as the mean-squared-error of the prediction of every currently known sample. The sample resulting in the least loss of generalization upon its temporary unlearning is forgotten permanently. It is possible that this is the newest data point, or one of the old ones.

Table 3 Data rate estimation results for all three approaches

	RMSE (MBit/s)	MRE	CV
Approach 1	16.21	0.275	0.318
Approach 2	14.41	0.247	0.283
Approach 3	14.81	0.297	0.291

After a new sample is learned and an old one is forgotten, the online learning algorithm adjusts its parameters so that it can adapt to the new data. This is necessary since the algorithm aims to capture the latest network state in order to improve data rate estimation.

- Depending on whether the new sample was supplied to Algorithm A or Algorithm B, the estimated data rate is taken from the selected algorithm and provided as a result.

In the following, three different approaches are compared:

- **Approach 1** The straightforward approach: all 4G and 5G parameters are fed into the PCA, which compresses them into one principle component, which then serves as input to a single ASR. In cases where the 5G parameters are zero, those zeros are replaced with minimum valid values as defined in 5G standardisation.
- **Approach 2** The first variant with two learning algorithms: If 5G values are nonzero, *all 4G and 5G parameters* are compressed by PCA and fed into an ASR; if 5G values are zero, only the 4G parameters are compressed by PCA and fed into a second ASR.
- **Approach 3** The second variant with two learning algorithms: If 5G values are nonzero, *only the 5G parameters* are compressed by PCA and fed into an ASR; if 5G values are zero, only the 4G parameters are compressed by PCA and fed into a second ASR.

For all three approaches, the hyperparameters used in the ASRs are tuned through Bayesian optimization. The PCAs are incremental PCAs that are pre-trained on the first data set from December 2021. The data set from August 2022 was used as test set.

Table 3 shows the root mean square error (RMSE), the mean relative error (MRE) and the coefficient of variation (CV) for the uplink data rate estimation on the August 2022 data set. It can be seen that Approach 2 shows the best performance, while Approach 1 shows the worst RMSE while being competitive in terms of MRE.

Figures 2, 3 and 4 further illustrate the findings. While the overall performance of all approaches looks similar (e.g., all of them struggle with the highly volatile situation between samples 600 and 700), the variants with two ASRs perform better in particular between sample 700 and 800.

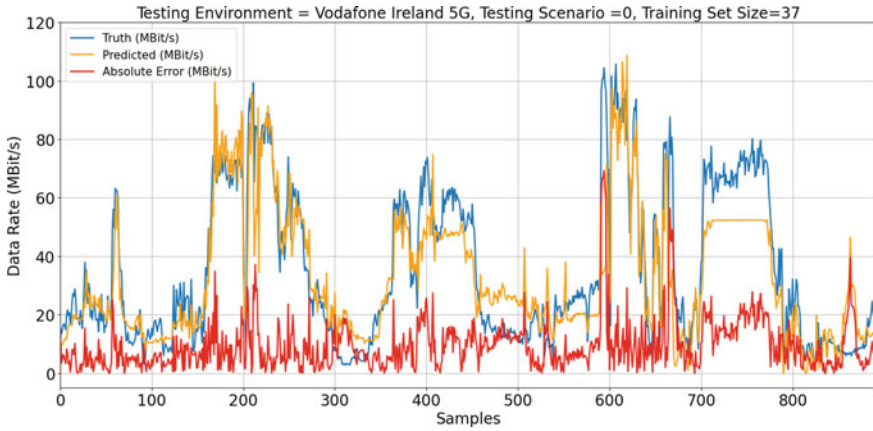


Fig. 2 Uplink data rate estimation, Approach 1

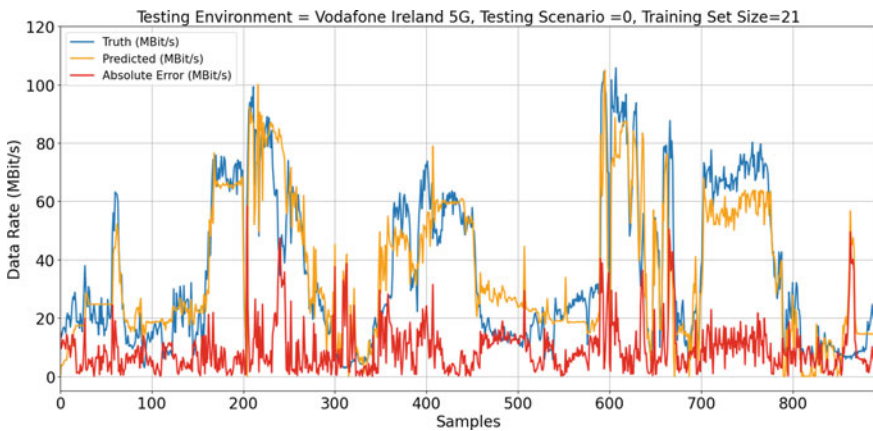


Fig. 3 Uplink data rate estimation, Approach 2

6 Interface Decision

As outlined earlier in this article, the aim is not just to estimate the available uplink data rate, but to manage connections, i.e., to select interfaces based on the estimates. An approach named MABASR+ was presented in [17]. MABASR+ estimates the uplink data rate of each link individually and selects the perceived best link based on the estimated instantaneous data rate and feedback about the difference between estimated and achieved data rate in the previous samples. In [17], different combinations of 4G and 5G parameters were used for the 5G estimation, while the estimation approach itself was the one that is labeled as *Approach 1* in this document.

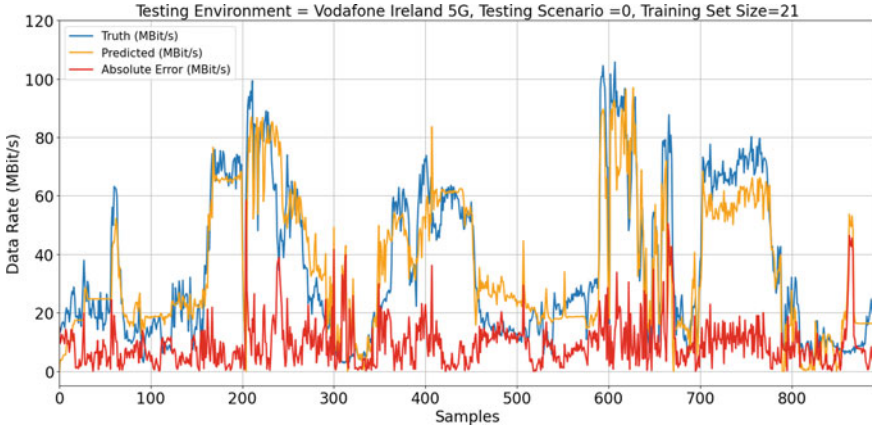


Fig. 4 Uplink data rate estimation, Approach 3

In the following, MABASR+ is used with Approaches 1 to 3 to further assess which of the approaches is most suitable for the purpose. For a more challenging scenario compared to the one used in [17], new measurements were taken in August 2022 pushing iPerf traffic on three modems (3G, 4G and 5G) simultaneously for a real-life situation where the UE faces varying conditions on all three links that sometimes favour one link and sometimes another. Other than sending traffic on all interfaces, the measurement setup was the same as described in Sect. 4. Figure 5 shows the measurement trace, with Link 1 being the 3G modem, Link 2 being the 4G modem and Link 3 being the 5G modem. The trace has been smoothed by a five-sample moving average filter. It is clearly visible that the 4G modem and the 5G modem each are the best option for significant periods of time, so it makes sense to switch between them instead of permanently favouring one over the other.

For the interface decision evaluation, the mean achieved reward, i.e., the mean achieved data rate, is used as metric. If perfect decisions are made, the achievable mean data rate is 38.699 MBit/s in the evaluation scenario. A policy that always chooses the 4G modem would achieve 28.625 Mbit/s, a policy that always chooses the 5G modem would achieve 29.709 MBit/s.

The estimators for the 3G and 4G modems used here were trained with the same data that was used in [16], whereas the estimator for the 5G modem was trained with the second measurement from December 2021. Figure 6 shows the result when Approach 1 is used for the 5G estimator. This result shows a strong preference for the 4G modem, indicating that either the estimation results are too low for the 5G modem, or too high for the 4G modem. While the mean achieved data rate of 30.140 MBit/s is higher than for both the always 4G or always 5G policy, the gain is rather modest and leaves significant room for improvement.

Figures 7 and 8 present the result for Approaches 2 and 3. It can be seen that there is a significant improvement in the achieved data rate compared to Approach 1. As there were no modifications to the estimators for the 3G and 4G modems, the

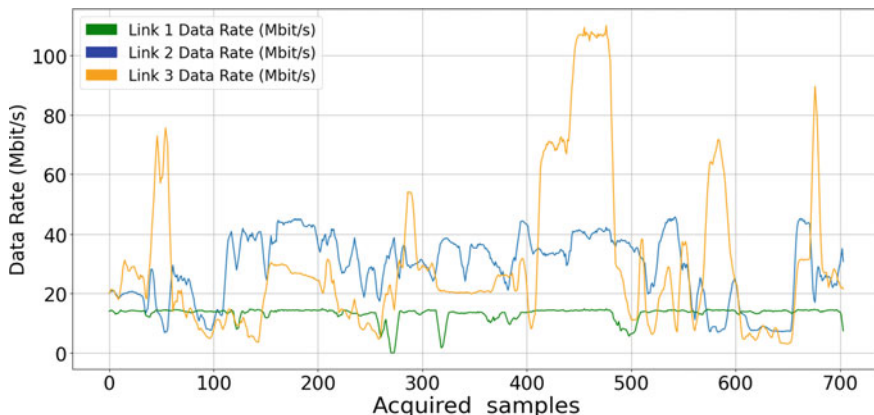


Fig. 5 Measurement trace with traffic on all links. Link 1: 3G modem, Link 2: 4G modem, Link 3: 5G modem

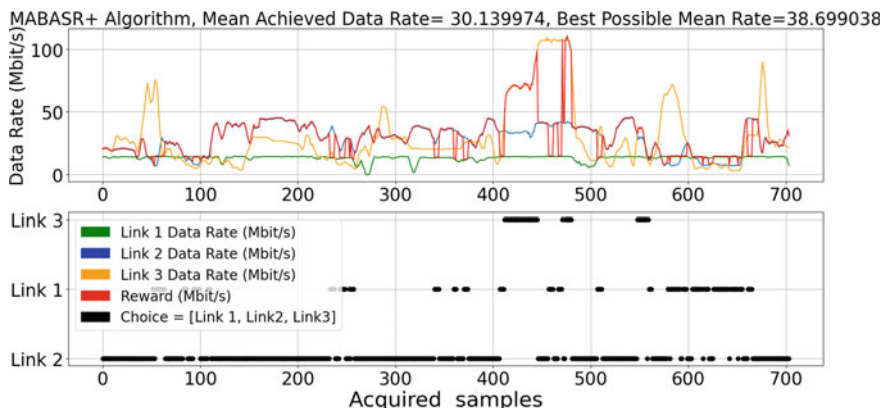


Fig. 6 MABASR+ with Approach 1

estimator for the 5G modem is the cause for the improved performance. It can be seen that the 5G modem is chosen more frequently than in the case of Approach 1. This further supports the observation made in Sect. 5 that the approaches with two ASRs perform better than the one with only one ASR for the 5G data rate estimation. The advantage of the approaches with two ASRs is even more prominent than it was before.

Another observation that can be made in Figs. 7 and 8 is the probing of other interfaces, i.e. brief interface changes which occur when the achieved data rate is significantly less than the estimated rate. For example, for a period starting from around sample number 150 in both cases, the 4G modem is selected most of the time, while the 5G modem is probed briefly on several occasions. This indicates that the estimation for the 5G modem would produce a higher data rate than that for the

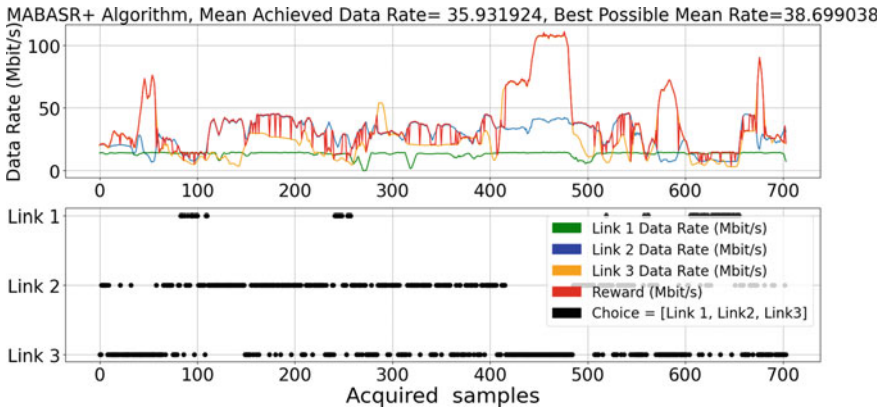


Fig. 7 MABASR+ with Approach 2

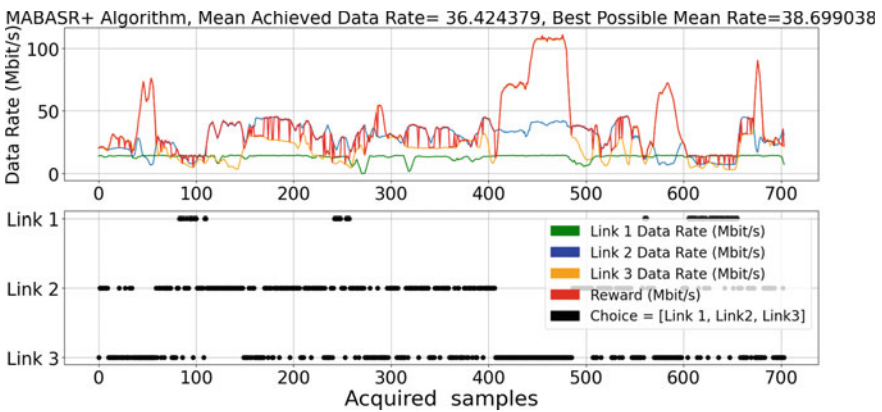


Fig. 8 MABASR+ with Approach 3

4G modem, but the achieved rate is less. In that case, the 4G modem is selected, while probing the 5G modem from time to time to check whether its actual achievable data rate has recovered to match its estimate. The interval in which this probing happens is subject to a trade-off between the ability to quickly react to changes and the need to limit the amount of connection changes as too many of those would negatively impact ongoing transmissions.

7 Conclusion

This article presents the use of AI algorithms for data rate estimation and interface selection in cellular networks, with a particular focus on the specific challenges of 5G NSA. Three variants are proposed for data rate estimation based on available

channel parameters, and their estimation performance is shown. Then, in extension, it is shown how they can be combined with estimators for other interfaces (in this case, 3G and 4G) and how they perform in interface selection in such a scenario. Future work will include further optimizations to the algorithms, as well as the inclusion of short term predictions to assess the expected data rate a few samples ahead.

Acknowledgements The author of this work would like to thank KLAS for providing the equipment used for the cellular network measurements and thus making this research possible.

References

1. Oshiba, T., Nogami, K., Nihei, K., Satoda, K.: Robust available bandwidth estimation against dynamic behavior of packet scheduler in operational LTE networks. In: IEEE Symposium on Computers and Communication (ISCC) (2016)
2. Tachibana, A., Paul, A.K., Hasegawa, T.: Next-t: Available bandwidth measurement over 4G/LTE networks - a curve-fitting approach. In: IEEE 30th International Conference on Advanced Information Networking and Applications (AINA) (2016)
3. Shiobara, S., Okamawari, T.: A novel available bandwidth estimation method for mobile networks using a train of packet groups. In: 11th International Conference on Ubiquitous Information Management and Communication (IMCOM) (2017)
4. Oshiba, T., Sato, N., Nogami, K., Sawabe, A., Satoda, K.: Experimental comparison of machine learning-based available bandwidth estimation methods over operational LTE networks. In: IEEE Symposium on Computers and Communications (ISCC) (2017)
5. Qian, F., Huang, J., Guo, Y., Zhou, Y., Mao, Z.M., Xu, Q., Sen, S., Spatscheck, O.: An in-depth study of LTE: effect of network protocol and application behavior on performance. *ACM SIGCOMM* **2013**, 363–374 (2013)
6. Nikolov, G., Kuhn, M., McGibney, A., Wenning, B.-L.: ASR—adaptive similarity-based regressor for uplink data rate estimation in mobile networks. *IEEE J. Sel. Areas Commun.* **38**(10), 2284–2294 (2020)
7. Heimann, K., Falkenberg, R., Wietfeld, C.: Discover your competition in LTE: client-based passive data rate prediction by machine learning. In: IEEE Global Communications Conference (GLOBECOM) (2017)
8. Sliwa, B., Schippers, H., Wietfeld, C.: Machine learning-enabled data rate prediction for 5G NSA vehicle-to-cloud communications. In: IEEE 4th 5G World Forum (5GWF) (2021)
9. Ylitalo, J., Jokikyyny, T., Kauppinen, T., Tuominen, A.J., Laine, J.: Dynamic network interface selection in multihomed mobile hosts. In: 36th Hawaii International Conference on System Sciences (2002)
10. Song, Q., Jamalipour, A.: Network selection in an integrated wireless LAN and UMTS environment using mathematical modeling and computing techniques. *IEEE Wirel. Commun.* **12**(3), 42–48 (2005)
11. Bari, F., Leung, V.C.M.: Automated network selection in a heterogeneous wireless network environment. *IEEE Netw.* **21**(1), 34–40 (2007)
12. Tian, D., Zhou, J., Wang, Y., Lu, Y., Xia, H., Yi, Z.: A dynamic and self-adaptive network selection method for multimode communications in heterogeneous vehicular telematics. *IEEE Trans. Intell. Transp. Syst.* **16**(6), 3033–3049 (2015)
13. Xu, K., Wang, K.-C., Amin, R., Martin, J., Izard, R.: A fast cloud-based network selection scheme using coalition formation games in vehicular networks. *IEEE Trans. Veh. Technol.* **64**(11), 5327–5339 (2015)

14. Niyato, D., Hossain, E.: Dynamics of network selection in heterogeneous wireless networks: an evolutionary game approach. *IEEE Trans. Veh. Technol.* **58**(4), 2008–2017 (2009)
15. Nikolov, G., Kuhn, M., Wenning, B.-L.: A contextual bandit approach to the interface selection problem. In: *IEEE 24th International Conference on Emerging Technologies and Factory Automation (ETFA)* (2019)
16. Nikolov, G., Kuhn, M., McGibney, A., Wenning, B.-L.: MABASR—a robust wireless interface selection policy for heterogeneous vehicular networks. *IEEE Access* **10**, 26068–26077 (2022)
17. Wenning, B.-L.: Cellular interface selection in multi-homed vehicular onboard gateways. In: *IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)* (2022)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Vehicle Communication Platform to Anything-VehicleCAPTAIN



Christoph Pilz

Abstract Vehicle-to-everything (V2X) is on the verge of being integrated as an integral part of modern vehicles. However, the battle for the final V2X radio technology is still not decided, and the available message standards are complex, hindering research and development. Hence, in our work, we provide a toolbox for early-stage development within V2X, called vehicle communication platform to anything (vehicleCAPTAIN). The vehicleCAPTAIN toolbox comprises (i) a set of software that mitigates the need to adapt for different V2X hardware vendors by decoupling and simplifying implementation, (ii) a set of libraries that supports encoding and decoding of messages, and (iii) a set of software bindings to enable development with ROS2. All software is provided as free and open source (FOSS). Within this work, we prove the functionality of the toolbox with classic end-to-end tests, as well as Qosium, a quality-of-service testing application. Finally, we highlight the applicability and benefits of the vehicleCAPTAIN toolbox for early-stage V2X research and development.

Acronyms

3GPP	3rd Generation Partnership Project, is a union of seven mobile network standard development organizations.
ADD	Automated Driving Demonstrator, is used short for vehicles that are used for automated driving demonstrations.
API	Application Programming Interface, is a software interface to control another piece of hardware or software.
ASN1	Abstract Syntax Notation One, is a syntax notation format.
CAM	Cooperative Awareness Message, is a V2X message that allows an ITS-S to share information about itself.
CPM	Collective Perception Message, is a V2X message that allows the exchange of sensor data, such as detected objects.

C. Pilz (✉)

Virtual Vehicle Research GmbH, 21a Inffeldgasse, Graz 8010, Austria

e-mail: christoph.pilz@v2c2.at

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_11

185

C-V2X	Cellular V2X, is a V2X communication standard based on mobile networks.
DENM	Decentralized Environmental Notification Message, is a V2X message that allows sharing of information, such as potholes, accidents, construction sites, etc.
DSRC	Dedicated Short Range Communications, is protocol within WAVE.
ETSI	European Telecommunications Standards Institute, is a European standardization institute.
FOSS	Free and Open Source Software, is a term used for free software with publically available source code.
GNSS	Global Navigation Satellite System, location discovery systems, such as GLONASS, GPS, etc.
HLA	High-Level Architecture, is the high-level perspective of a complex system, typically in a diagram.
ITS	Intelligent Transport Systems, the conglomeration of V2X capable actors.
IEEE	Institute of Electrical and Electronics Engineers, is a standardization organization.
ITS-S	ITS Station, a V2X actor.
ITSG5	Intelligent Transport Systems G5; G5 stands for the frequency band (5.9GHz), is a V2X communication standard based on WIFI
IVIM	In-Vehicle Information Message, is a V2X message that allows sharing of control information for driver and vehicle, such as speed limits, allowed automated driving levels, etc.
LTE	Long Term Evolution, is a mobile network standard.
MQTT	Message Queueing Telemetry Transport, is a middleware for data exchange.
OBU	Onboard Unit, is a hardware component, within V2X, that is used in the vehicle for communication.
ROS2	Robot Operating System 2, is a platform for rapid prototyping of robots.
RSSI	Received Signal Strength Indicator, is an indicator for received signal strength.
RSU	Roadside Unit, is a hardware component, within V2X, that is used by infrastructure for communication.
RTCMEM	Radio Technical Commission for Maritime services Environmental Message, is a V2X message that allows sharing of GNSS correction data.
SAE	Society of Automotive Engineers, is a standardization organization in the automotive domain.
UART	Universal Asynchronous Receiver/Transmitter, is an interface for data exchange.
V2N	Vehicle-to-Network, is the network-based communication between vehicles and everything else.
V2X	Vehicle-to-Everything, is the wireless communication between vehicles and everything else.

WAVE	Wireless Access in Vehicular Environments, is the wireless communication environment for 802.11p in the U.S.A.
ZMQ	Zero Message Queue, is a middleware for data exchange.
WIFI	Wireless Fidelity, is a communication standard for wireless networks.

1 Introduction

Automotive research started the standardization of the wireless standard 802.11 for vehicle-to-vehicle (V2V) communication in the early 2000s. Nowadays, the standard includes, among others, vehicle-to-infrastructure (V2I), and vehicle-to-network (V2N), the latter meaning communication via web-service backends. The family of vehicle communications with everything else is called vehicle-to-everything (V2X) communication. From the technology perspective, the 802.11p standard was released in 2010 by the Institute of Electrical and Electronics Engineers (IEEE) [1], which derives from the commonly known WiFi standard. Then, in 2017 the 3rd Generation Partnership Project (3GPP) [2] released the cellular V2X (C-V2X) standard based on the PC5 side channel in LTE. The IEEE and the 3GPP also announced predecessors 802.11bd and C-V2X (5G-based), respectively. Before the hardware crisis started in 2020, the soon available four technologies competed for a central standard. To date, the battle is still ongoing.

Communication for intelligent transport systems (ITS) also needed a software protocol. The Society of Automotive Engineers (SAE) and European Telecommunications Standards Institute (ETSI) started specification in parallel with their respective protocols, Wireless Access in Vehicular Environments (WAVE) and Intelligent Transport Systems G5 (ITS-G5). While the network protocol is not fully compatible, the messages found a common baseline. The ETSI consortia define the messages using parts of the messages defined in the Dedicated Short Range Communications (DSRC) protocol, which is part of WAVE. Nowadays, following the combined effort, all technologies mentioned above, i.e., 802.11p/bd and C-V2X (LTE/5G), are designed to use the message standards defined by ETSI, with messages such as cooperative awareness message (CAM) and decentralized environmental notification message (DENM).

V2X development now has an easy and complicated side. The easy side is that there is a common message protocol. The complicated side is the multitude of (soon) available hardware technologies. Before 2020, manufacturers of V2X solutions were providing mainly 802.11p systems but quickly adopting the released C-V2X (LTE) standard. Among others, Cohda,¹ Commsignia,² Yogoko,³ and Yunex⁴ are providing onboard units (OBUs) and roadside units (RSUs), each with its application program-

¹ <https://www.cohdawireless.com/>.

² <https://www.commsignia.com/>.

³ <https://www.yogoko.com/>.

⁴ <https://www.yunextraffic.com/de/>.

ming interface (API). Adding multiple APIs now increases the effort in research and development for implementing application software if there is a need to stay independent of the underlying V2X technology. Each of the companies mentioned above aims to provide a single API. However, independent of the used technology, there is still the issue of adapting to multiple APIs and using other prototypes, such as V2N.

In our work, we aim to provide a solution for the interface problem. We created a platform that can host multiple V2X interfaces and route the messages from the interfaces to the user and from the user to the respective interfaces. This way, the user only has to implement one interface. The platform handles the actual hardware interfaces. In addition, we also provide a free and open source (FOSS) message library for the ETSI-defined messages to get started, as well as a ROS2 interface.

In the following Chap. 2, we provide more detail on the underlying issue of interfaces. In Chap. 3, we present the vehicle communication platform to anything (vehicleCAPTAIN) as the solution. The functionality of the vehicleCAPTAIN is then verified in Chap. 4. In Chap. 5, we discuss the applications and benefits of the vehicleCAPTAIN before concluding in Chap. 6.

2 Problem Statement

V2X has several communication standards. The most common are 802.11p and C-V2X (LTE), with their predecessors 802.11bd and C-V2X (5G). V2N extends this range with various web-based interfaces, with Ethernet, WiFi, 4G, and 5G as underlying technologies [3]. However, V2N needs a management system that relays messages to specific users, similar to the GeoNetworking protocol, with middleware, such as MQTT [4]. Going a step further, there are also mmWave protocols [5–7] currently under development that may be included in the V2X communication domain.

The complexity of this multi-interface structure should not hinder research and development of the application of V2X. Hence, there is a need for a platform that takes away the implementation effort for interfaces from the user. In other words, there is the need for a platform that is open enough to integrate new hardware interfaces with their respective APIs; have the option to configure the routing of messages, depending on the message type, payload, and connectivity; yet also provide one user interface for straightforward integration.

3 VehicleCAPTAIN—A V2X Platform for Research and Development

The vehicleCAPTAIN is a V2X toolbox designed for research and development and developed by Pilz et al. [8]. It is designed to lower the entry barrier for V2X development by providing various software repositories with sample code and libraries.

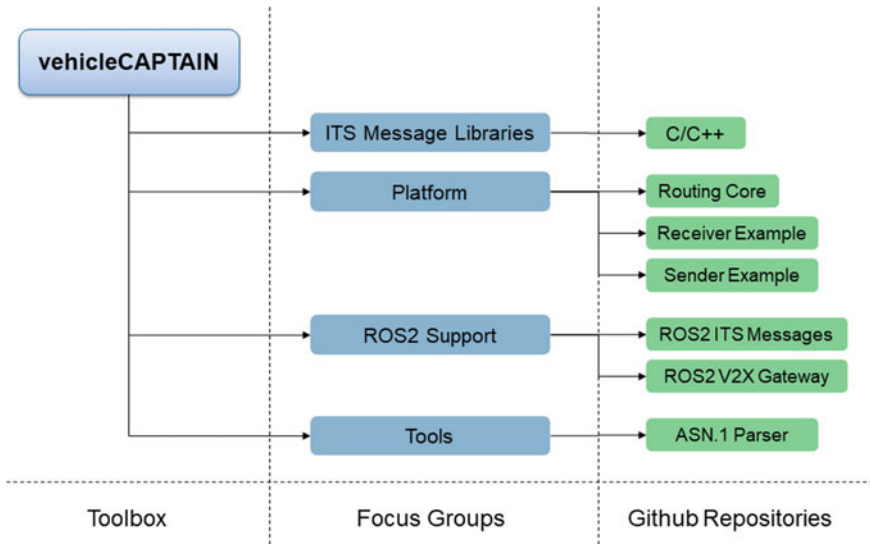


Fig. 1 The vehicleCAPTAIN toolbox is split into four major groups. These four major groups currently result in seven public GitHub repositories

The components, shown in Fig. 1, are available as FOSS by Pilz et al. [8]. Essential details of the components are discussed in the following chapters. Section 3.1 introduces the routing platform and the core concept of how it eases development with V2X as a developing standard. Section 3.2 continues by introducing the C/C++ message library, followed by Sect. 3.3 introducing the ROS2 support.

3.1 The Platform

The core elements of the vehicleCAPTAIN consist of a hardware configuration and the software routing core. The hardware configuration is independent of the software as long as the hardware can host the software and as long as the hardware configuration contains at least one transmission device with V2X capability. The software provides one interface to control one or more V2X interfaces simultaneously. The software simply needs an interface driver for the respective transmission device.

One can look at the HLA of the vehicleCAPTAIN, shown in Fig. 2, to understand how the platform works. The vehicleCAPTAIN provides a simple ZMQ interface for the user. Single V2X messages are exchanged as byte streams in both directions, receiving and sending. Starting with sending, the message is relayed to the Routing part. The Routing may be extended as required for research purposes. One may, for example, handle the GeoNetworking protocol here or prioritize specific messages. In the simplest case, the messages are relayed to the Communication Control. The

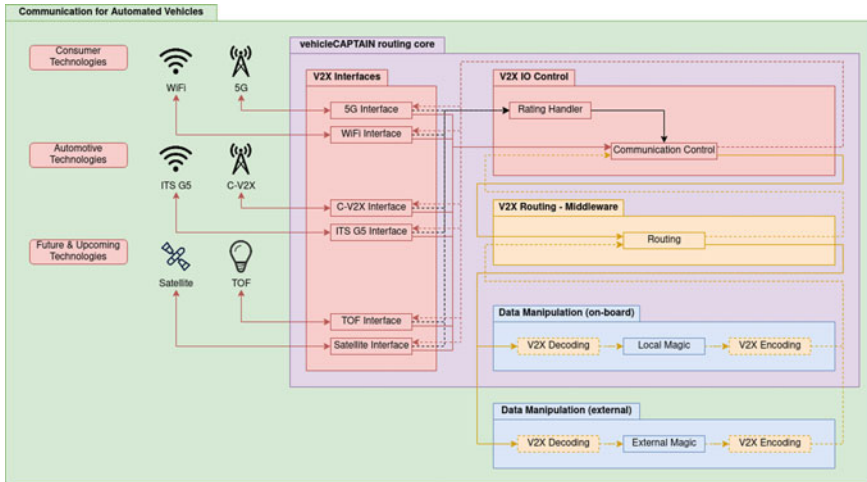


Fig. 2 The HLA of the vehicleCAPTAIN routing core. The left side shows V2X hardware interfaces. The bottom right has the user magic connected via ZMQ

Communication Control decides on the message type or other parameters via which interfaces to distribute the message. The message is then forwarded to the specific Interface implementations. Each Interface implementation then takes each message and transmits it by sending it via the respective hardware interface. Conversely, another vehicleCAPTAIN may receive a message via an Interface. The Interface implementation extends the incoming message with other received information, such as RSSI or geolocation. The Interface implementation holds this raw message in its incoming queue until the Communication Control collects it. The Communication Control collects and relays the messages to the Routing part. The Communication Control and the Routing part may also use the additional receiver information to adapt their behavior. In its simplest form, the message is relayed to the ZMQ interface for the user.

All in all, the vehicleCAPTAIN routing core is a simple message-passing software. But depending on which and how many interfaces have to be tested, it eases implementation, as user code is separated from message relaying. One may test other messaging approaches, such as mmWave, UART communication, or Morse code. It simply depends on the implementation of the Interface implementation.

Last but not least, the vehicleCAPTAIN needs hardware. Currently supported is ITSG5 communication via the Unex SOM301-E and MQTT via Ethernet or plugged-in mobile network connections, such as 5G. The Virtual Vehicle Research GmbH can provide schematics or assembled hardware, as shown in Fig. 3. We currently use this setup with three modules. The Unex SOM301-E for ITSG5 communication, the Simcom sim8202g for 5G and V2N, and the Ublox F9P for highly accurate time-synchronization via 1pps.



Fig. 3 The vehicleCAPTAIN development kit in Quad Expander configuration, with a Raspberry Pi 4 as the computing platform (underneath the circuit board). The Quad Expander configuration can host up to four mPCI-E or M.2 modules. The configuration is from left to right (i) Ublox F9P with 1pps, (ii) empty, (iii) Unex SOM301-E, and (iv) Simcom sim8202g

3.2 Message Library

The ETSI provides the message exchange protocol for all V2X-related standards as Abstract Syntax Notation One (ASN1) in a public git repository [9]. Descriptions of each message element and the usage of each message are distributed across various public standardization documents. The ETSI repository contains scattered instructions on generating a C and C++ library from the specifications. However, it takes a few days of effort to set up everything.

A library can ease this entry barrier. Hence, Pilz et al. [8] made a generator script, which creates a ready-to-use library. The creation process is shown in Fig. 4. As one can see, everything is run in a Docker environment to provide a clean environment. A clean environment is needed for the ASN1 compiler to be installed correctly so that the generated library contains clean user paths, which are included as generated comments in the header. Within the setup environment, the script handles the generation. At startup, the script automatically loads the newest asn1 specification files for ETSI specifications, its dependencies, and additional defined experimental messages. After loading, the script provides a simple menu where the messages to generate can be selected. The options are (i) to generate specific messages, (ii) to only generate standardized messages, or (iii) to generate all messages which are available to the script. Each selection triggers the generation of multiple C-Header and C-Source files dumped in a folder of the respective message. One may already use this dump or the CMake option in the menu. The CMake option will rearrange the generated files to fit a library structure and automatically generate the required CMake files. The final library is ready to use in CMake projects. However, depending on the CMake version and CMake environment, it may be necessary to adapt the CMake syntax, as usual.

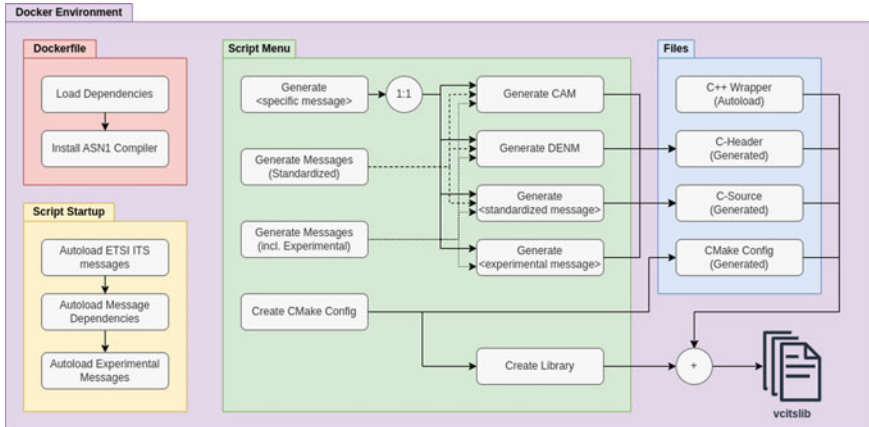


Fig. 4 The *vcitslib* creation process, within its Docker environment. On the left side is the setup of the environment. In the middle is the creation process of the files. On the right are the final files which are rearranged into a library

One must say that vendors of complete OBU and RSU hardware platforms already provide such libraries. However, they are not free and also not open source. The latter may be necessary for research of message standards. The script provided in the repository allows the addition of ASN1 specifications outside existing standards. The generated library then includes those new messages without any extra effort.

The *vcitslib* is a core component of the vehicleCAPTAIN toolbox and, therefore, already integrated into all necessary parts to allow rapid development. However, one may use a commercial ITS message library at any time. Within InSecTT, we use this library to exchange Collective Perception Messages (CPMs) in the airport use case. One robot can detect debris and share this information via V2X with other robots.

3.3 ROS2 Support

Research and development teams are often using ROS2 for rapid prototyping. ROS2 also provides much debugging functionality. Hence part of the vehicleCAPTAIN toolbox is the ROS2 interface. This interface can be used specifically for V2X messages.

Part of the ROS2 interface is the ROS2-type messages as a complement to the ETSI messages discussed in Sect. 3.2. The ROS2-type messages are generated from the ASN1 specifications. To achieve this, the ASN1 code generator [10], recommended by ETSI, is forked to include functionalities to generate ROS2 messages from ASN1 specifications. The parser adaptations are not yet perfect. Hence a few hotfixes are necessary for complex edge cases in the specification. However, the provided generator script can generate ROS2-type messages from provided ASN1 files and output a ready-to-use ROS2 message library as source code and an installable package.

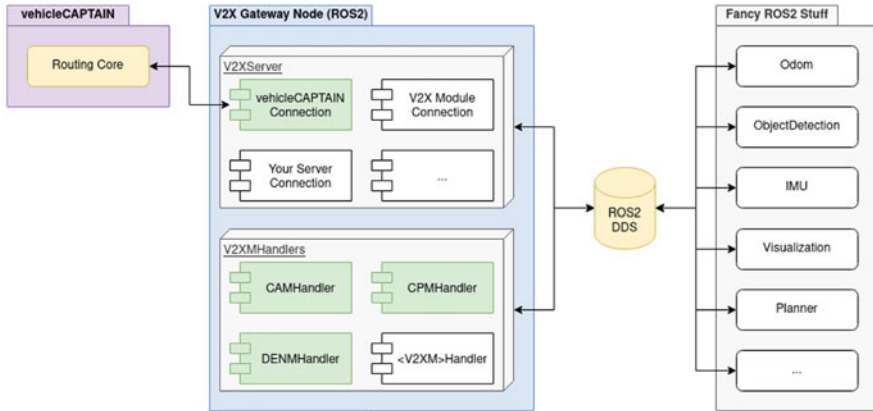


Fig. 5 The vehicleCAPTAIN gateway for ROS2 is a translator between asn1 encoded V2X messages and ROS2 encoded V2X messages. The gateway is compatible with the vehicleCAPTAIN routing core but can be easily adapted to support only a specific interface

The other part is the ROS2 V2X gateway, a translator between ETSI V2X messages and ROS2 V2X messages, shown as HLA in Fig. 5. The gateway is designed to use the vehicleCAPTAIN routing core to input and output ETSI V2X messages. Still, the internal abstractions allow straightforward changes with a different hardware and software environment for ETSI V2X message transmission. Furthermore, the gateway has handlers for every V2X message to account for the different requirements of the message itself. For example, the gateway may be used to buffer messages to account for 1–10Hz sending, or it might be used to pre-filter specific message types for applicability. However, in the provided form, the handlers simply translate V2X messages between ETSI format and ROS2 format. Within the InSecTT use cases, the logic for handling V2X messages is implemented behind the gateway. The advantage is that one can use the input and output of the gateway on the ROS2 side to debug V2X messages beyond the capability provided by network analyzing tools, such as Wireshark. Additionally, the abstraction allows decoupling the message translation from the actual implementation, i.e., one can focus on the generation and interpretation of messages while not having to deal with the translation and verification of messages.

4 Verification

The functionality of the vehicleCAPTAIN is regularly tested in the lab environment, and its performance was verified with the aid of Qosium,⁵ provided by the InSecTT partner Kaitotek. This book chapter will focus on the discussion of the validity of transmission. In parallel, the journal paper by Pilz et al. [11] has focused on

⁵ <https://www.kaitotek.com>.

highly time-accurate delay measurements and packet loss caused by the wireless connectivity, including the overhead of the vehicleCAPTAIN platform and thereby proving its overall functionality.

This book chapter will analyze the system and how the payload transmission can be proven. Section 4.1 will briefly discuss the input/output relation of the vehicleCAPTAIN before explaining the test setups. Section 4.2 will then show the found statistics before Sect. 4.3 reviews the outcome.

4.1 Test Methodology

If one has two vehicleCAPTAINs, one sender and one receiver, the messages' payload on both ends must be the same, as discussed in Sect. 3.1. This basic functionality can prove that the system is working correctly. Three separate tests verify the platform: (i) the input/output verification with simple senders and receivers, (ii) the input/output verification with the ROS2 gateway, and (iii) the packet loss test with Qosium.

In the first two cases, the two vehicleCAPTAINs stand side-by-side in our lab, as shown in Fig. 6. In those two cases, the aim is zero packet loss, as the goal is to verify the correctness of the transmission.



Fig. 6 Two vehicleCAPTAIN development kits stand on each other for continuous integration tests

The first case is simple input/output testing. A small set of ten asnl encoded bitstreams, including CAM and DENM data, is sent out via vehicleCAPTAIN A with a set frequency. The data is then received with vehicleCAPTAIN B and compared for similarity.

The second test is similar. Here we use the ROS2 gateway to verify that the transmission and encoder are working correctly. Hence, the ROS2 gateway generates CAM data with our automated driving demonstrators (ADDs) simulation environment and sends them via vehicleCAPTAIN A. The data is then received with vehicleCAPTAIN B, respectively the second gateway, and we compare the received and sent messages.

In the third case, the vehicleCAPTAINs are mounted into two street legal ADDs of the Virtual Vehicle Research GmbH. As discussed above, the goal is to test the transmission delay and packet loss, which we published in detail in [11]. The principle is similar to the first test. Qosium probes are installed on the vehicleCAPTAIN computing hardware and check the incoming and outgoing ZMQ TCP/IP packets. Because of ZMQ, the payload of the TCP/IP packets for V2X messages are the same when sending and receiving, even though they are repacked in between for V2X message transmission. The ADDs drive around a building block of $100\text{ m} \times 250\text{ m}$ with specific distances. This way, one can verify the delay and the packet loss in a field scenario.

4.2 Results

The results are provided in three tables. Table 1 shows the simple sending and receiving results. The packet loss is minimal and most likely caused by another Yunex RSU in the same room sending messages. However, one can see that the packets are transmitted correctly, as at least one correct message of each type was received correctly.

The results shown in Table 2 show that also the ROS2 gateway is working correctly, as the messages sent out equal the messages coming in on the other end.

Finally, shown in Table 3 is an excerpt of the measurements conducted with Qosium. Here one can see a different metric for counting packets. Qosium averages the quality of service results over a definable period, in our case, 1000ms. Hence Table 3 shows the sum of averaging periods with the direct result of overall packet loss.

Table 1 Simple sender receiver tests

Sending frequency [Hz]	\sum_{sent} [#]	$\sum_{received}$ [#]	Packet loss [%] ^a
1 Hz	1000	999	0.10
10 Hz	10000	9971	0.29
20 Hz	10000	9963	0.37

^aEach message in the set is transmitted at least once correctly

Table 2 ROS2 gateway sender receiver tests

Sending frequency [Hz]	$\sum sent$ [#]	$\sum received$ [#]	Packet loss [%]
1 Hz	1021	1019	0.20
10Hz	10113	10085	0.28
20Hz	10125	10084	0.41

Table 3 Qosium packet loss test. Averaging interval (T) = 1000 ms. CAM frequency = 10Hz

Distances	$\sum T$	N/A	Packet loss [%]
Campus standstill (10 m)	173	–	0.12
Campus round (50 m ^a)	259	–	13.69
Campus round (150 m ^a)	266	–	55.74
Campus round (250 m ^a)	278	–	57.78

^aDriving around a building complex (100 m × 250 m) with different distances at the Graz University of Technology campus

4.3 Discussion

The results presented in Sect. 4.2 prove the fundamental purpose of the platform to be able to relay messages reliably. Results shown in Tables 1 and 2 show that each generated message can be received at least once. Table 3 then includes packet loss results from test drives where the line of sight between sender and receiver was obstructed. The packet loss in those scenarios is as expected. The output of the vehicleCAPTAIN platform is also continuously tested with the aid of Wireshark as a reference decoder. Known issues are tracked within the respective GitHub repositories [8].

Overall, one must also verify the other components, such as the C/C++ encoder called *vcitslib* and the ROS2 message library. The conducted sending and receiving tests implicitly prove their functionality, as messages can be encoded and decoded. However, we currently only test messages we implement in use cases. In other words, both libraries are constantly under development because the V2X message standard is extremely nested and complex and, therefore, not yet implemented and tested in every aspect.

5 Key Performance Indicators

As discussed, the vehicleCAPTAIN toolbox consists of four major elements: (i) the platform itself, providing V2X transmission hardware and a software routing core, (ii) the ITS message libraries, currently supporting C and C++ development, (iii) the ROS2 support, which consists of a ROS2 type ITS message library and a supporting message translation gateway, as well as (iv) the additional tools, which currently support the generation of ITS messages for ROS2.

Each of the components lowers the entry barrier for V2X development. The components are not designed to provide a complete and ready implementation but to provide a basis for research and development. The toolboxes for ready-to-use OBUs, as discussed in Chap. 1, may be enough. But as soon as specific requirements have to be fulfilled, the vehicleCAPTAIN toolbox can help. Requirements could be the parallel support of different V2X radio standards, testing new and yet unsupported message types, or using ROS2 components.

5.1 Use Cases Within InSecTT

The vehicleCAPTAIN demonstrates its capability in two demonstrators. One demonstrator shows V2X management from the infrastructure side with a complex simulation and an actual intersection in Istanbul, Turkey. Here the vehicleCAPTAIN is used for the communication of various V2X status messages.

The other demonstrator showcases the sharing of perception information between robots and vehicles. This demonstration uses the *vcits* library to generate an encoder for pre-standardized CPMs. All of the systems in the demonstrator run ROS. Hence additionally, the ROS2 interface is used for the translation of messages.

5.2 Use Cases Within the Virtual Vehicle Research GmbH

The use cases are plentiful, as designed by the approach of V2X. To get a better understanding, this section provides an overview of scenarios in which the vehicleCAPTAIN toolbox is already of high value for the Virtual Vehicle Research GmbH.

The *vcitslib* is the most used component of the vehicleCAPTAIN toolbox. We use this library across various projects and in multiple scenarios. We use it extensively to share positional information for sensor fusion, i.e., Cooperative Awareness Messages (CAMs) for ego information and CPMs for second-person information. We also use other messages, such as Decentralized Environmental Notification Messages (DENMs) to react at active tramway crossings, In-Vehicle Notification Messages (IVIMs) to react on speed limit changes, automated driving levels, and open bus lanes, as well as Radio Technical Commission for Maritime Services Environmental Messages (RTCMEMs), to receive GNSS correction data.

The vehicleCAPTAIN hardware and routing core are highly valuable because of their versatility. At first, used the Unex SOM301-E for ITSG5 communication, which could have also been done with a complete OBU from another vendor. However, soon we needed highly accurate time synchronization for testing and verification. The vehicleCAPTAIN hardware allowed us to adapt quickly by including 1pps synchronization via a Ublox F9P. The next challenge was to provide better connectivity over farther distances in city environments. The simple solution was the integration of a 5G module and the implementation of V2N. As we are using the vehicleCAPTAIN

toolbox, it was as simple as integrating a new interface into the vehicleCAPTAIN routing core. This way, we did not have to change any other parts, and all other use cases were immediately able to use the same way of communication.

The vehicleCAPTAIN routing core and its connectivity are used for major projects. Additionally, as mentioned in Sect. 3.3, we see high potential in using ROS2 in our use cases. ROS2 allows the reusing of components, allowing rapid prototyping and focusing on separate integration elements. In other words, at the Virtual Vehicle Research GmbH, we have street-legal ADDs running Autoware based on ROS2. The ROS2 gateway allows for a simple vehicle extension with V2X capability.

6 Conclusion

In this book chapter, we discussed the vehicleCAPTAIN toolbox. The toolbox is designed (i) to ease entry into the V2X development domain, (ii) to provide a platform for cross-development with existing and upcoming V2X interfaces, and (iii) to provide development interfaces for rapid prototyping with ROS2.

We found the toolbox to fulfill all these needs by (i) the discussion verifying its design and by providing FOSS releases for its components, (ii) the verification of sending capability and performance in lab and field environments of the vehicleCAPTAIN development kit and routing core, and (iii) the discussion of use cases already implemented within and beyond InSecTT.

Development is currently focusing on releasing additional papers proving the capability of the vehicleCAPTAIN toolbox and providing quickstart tutorials for every component. Aside from those crucial steps, future development will focus on upgrading the provided V2X interfaces with the latest releases and integrating upcoming technologies.

Finally, we encourage V2X researchers and developers to use the vehicleCAPTAIN toolbox for inspiration and implementation, as it combines the know-how necessary for basic V2X implementations.

Glossary

vehicleCAPTAIN The vehicle communication platform to anything is a toolbox designed for easy entry into the V2X domain. It was created within the Ph.D. works of Christoph Pilz, at the Virtual Vehicle Research GmbH, in cooperation with Graz University of Technology.

References

1. IEEE: Working group for 802.11. Working group and project timelines. https://grouper.ieee.org/groups/802/11/Reports/802.11_Timelines.htm. Accessed 25 Apr 2023
2. 3GPP: Release 14 of mobile network specifications and technologies. <https://www.3gpp.org/specifications-technologies/releases/release-14>. Accessed 25 Apr 2023
3. Correia, M., Almeida, J., Bartolomeu, P., Fonseca, J., Ferreira, J.: Performance assessment of collective perception service supported by the roadside infrastructure. *MDPI Electron.* (2022). <https://doi.org/10.3390/electronics11030347>
4. Mishra, B., Kertesz, A.: The use of MQTT in M2M and IoT systems: a survey. *IEEE Access* (2020). <https://doi.org/10.1109/ACCESS.2020.3035849>
5. Yin, Y., Yu, T., Maruta, K., Sakaguchi, K.: Distributed and scalable radio resource management for mmWave V2V relays towards safe automated driving. *Sensors* **22**(1), 93 (2022). <https://doi.org/10.3390/s22010093>
6. Fukatsu, R., Sakaguchi, K.: Automated driving with cooperative perception using millimeter-wave V2V communications for safe overtaking. *Sensors* **21**(8), 2659 (2021). <https://doi.org/10.3390/s21082659>
7. Yin, Y., Yu, T., Maruta, K., Sakaguchi, K., Fukatsu, R., Sakaguchi, K.: Automated driving with cooperative perception based on CVFH and millimeter-wave V2I communications for safe and efficient passing through intersections. *Sensors* **21**(17), 5854 (2021). <https://doi.org/10.3390/s21175854>
8. Pilz, C.: Main Repository. In: vehicleCAPTAIN toolbox. Available via GitHub. https://github.com/virtual-vehicle/vehicle_captain. Accessed 26 Apr 2023
9. ETSI.: ASN1 specifications for V2X messages. <https://forge.etsi.org/rep/ITS/asn1>. Accessed 26 Apr 2023
10. Walking, L.: ASN1 compiler with fixes for V2X messages. Available via GitHub. https://github.com/brchiu/asn1c/tree/velichkov_s1ap_plus_option_group_plus_adding_trailing_ull. Accessed 26 Apr 2023
11. Pilz, C., Sammer, P., Steinbauer-Wagner, G., Grossschedl, U., Piri, E., Kuschnig, L., Steinberger, A., Schratte, M.: Collective perception: a delay evaluation (under review)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



AI-Enhanced UWB-Based Localisation in Wireless Networks



Eshagh Dehmollaian, Bernhard Etzlinger, Philipp Peterseil,
and Andreas Springer

Abstract Thanks to low complex and affordable hardware, low power consumption, and pulse-based communication, ultra-wideband (UWB) technology has brought the possibility of positioning in wireless networks for various applications with high precision. Nowadays, the widespread use of this technology for location-based applications together with the integration of this technology in smartphones, motivates more research on the use of this technology for localisation systems. Current research results emphasize that artificial intelligence (AI) algorithms can help to improve the positioning performance of UWB technology due to the use of large amounts of data. In this work, we provide an overview of the challenges and their AI-based solutions in UWB-based localisation systems. This is followed by an overview of related work and an application example.

1 Introduction

Location awareness has long attracted much attention due to the increasing widespread of smartphones, smart vehicles, and Internet of Things (IoT) connected devices. In addition, the need of location information in various applications such as driverless cars, smart homes, passive keyless entry systems, etc. lead researchers to explore different aspects of localisation systems.

Localisation accuracy and broad availability of global navigation satellite systems (GNSS) such as global positioning system (GPS) and Galileo in smartphones and other devices are in most cases sufficient for outdoor applications. However, since their signals are too weak or degraded by multipath effects in indoor environments, other approaches are required for indoor applications. These approaches can be part of existing communication systems (e.g., WiFi access points) or be a dedicated technology like ultra-wide band (UWB). According to studies in the literature, UWB technology is one of the best candidates for indoor localisation. This is due to high time resolution, low power consumption, and low complex hardware of UWB tech-

E. Dehmollaian (✉) · B. Etzlinger · P. Peterseil · A. Springer
JKU Linz, Altenberger Straße 69, 4040 Linz, Austria
e-mail: eshagh.dehmollaian@jku.at

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_12

201

nology. Therefore, manufacturers have recently started to integrate UWB technology into mobile devices for location-based services (LBS) [1].

With the help of high bandwidth communications, UWB radio technology (IEEE 802.15.4z [2]) can achieve high time resolution. As a result, localisation with high accuracy can be achieved. However, multipath propagation (MPP) and non-line-of-sight (NLOS) conditions in UWB-based localisation technology, which frequently occur in indoor environments, are the challenges that impede the achievement of high accuracy [3]. Another challenge that compromises the accuracy of localisation systems is an intrusion attack. Since those systems are often used in unattended environments, security threats are crucial concerns to consider [4].

To solve the challenges, different classical signal processing methods and AI-based approaches can be used. Many recent publications demonstrate the effectiveness of AI algorithms in extracting knowledge, learning important features, and improving localisation performance. The advantages of AI over classical approaches are as follows: AI approaches are significantly more effective than traditional approaches in complex nonlinear problems. Due to the ease of adjusting AI-based approaches in case new data sets are added, AI algorithms can offer scalable solutions. Furthermore, unlike statistical methods, AI algorithms can easily be expanded to offer stable performance in a variety of environmental circumstances. In comparison to traditional approaches, they can incrementally adapt to changing environmental scenarios, thanks to their online learning abilities.

In this chapter we will have an overview of the challenges in UWB-based localisation systems and of the AI algorithms that can address these challenges. We have investigated 26 existing work in the literature [5–30] accordingly.

2 Method Overview

MPP, NLOS, and attacks distort the accuracy of UWB-based localisation systems. All considered AI approaches in this chapter aim to improve accuracy with or without considering also energy efficiency or security concerns. Therefore, we focus on maintaining or improving the accuracy of UWB-based localisation systems while having different additional objectives.

To improve accuracy only, these objectives vary significantly, considering if the localisation method is range based or range free. For range based approaches, where the accuracy of the inter-node distance estimation is of most importance, these objectives are (i) **feature selection** for extracting the most relevant features, (ii) **NLOS detection** for eliminating erroneous ranging information, and (iii) **NLOS mitigation** for error compensation. There is ample existing research in the literature which provides solutions to select most relevant features, detect, and mitigate ranging errors using AI algorithms. Clustering and AI-based dimensionality reduction (DR) techniques are frequently used for feature selection. For the detection and mitigation of NLOS errors classification, regression, clustering, etc. techniques are extensively used.

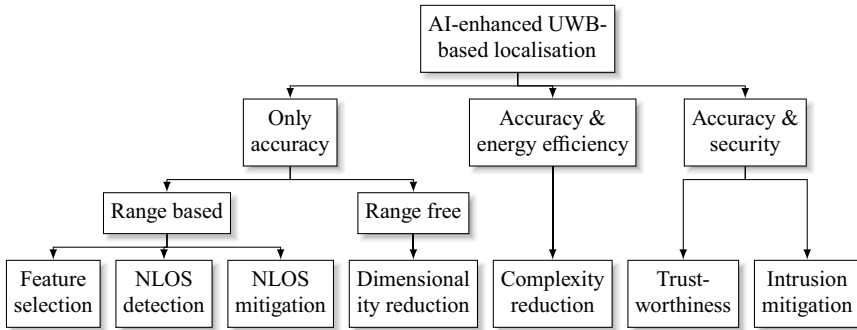


Fig. 1 Classification of the objectives of AI-enhanced UWB-based localisation systems

In range free approaches fingerprinting determines the localisation accuracy. The processing of high-dimensional data is a common concern. The utilized AI algorithms in this domain mainly reduce the dimensions of the data and thus reduce computational complexity. Therefore, **(iv) dimensionality reduction** is the objective in range free UWB-based localisation systems. In most cases AI-based DR and clustering techniques are proposed here.

Higher accuracy often comes with higher complexity. Therefore, energy efficiency has to be considered in the case that power is limited. The objective for the aim of accuracy along with energy efficiency is **(v) complexity reduction**.

For the aim of accuracy with energy efficiency concern, most related works apply complexity reduction methods. There has been some research on using AI algorithms to enhance accuracy along with complexity reduction.

If accuracy is compromised through attacks, the integrity of UWB measurements deteriorates. Hence, the secure and reliable operation of the system is compromised. There has been limited research focused on maintaining the system security together with accuracy, while focusing on the objectives **(vi) trustworthiness** and **(vii) intrusion mitigation**. Security concerns in UWB-based localisation have gained some attention recently. The aforementioned three aims and seven objectives are depicted in Fig. 1.

3 AI for Solving UWB-Based Localisation Challenges

3.1 Localisation Challenges

As stated, localisation methods that are considered in the references presented in this chapter can be categorized into range based and range free approaches. Therefore, their challenges are described in detail in the following two subsections.

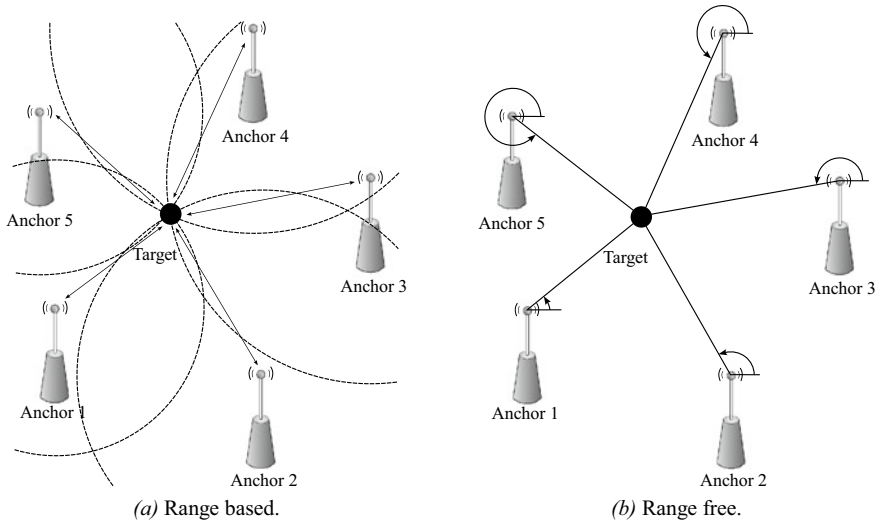


Fig. 2 Schematic diagram of positioning techniques for ideal cases

3.1.1 Range Based Approaches

With range based approaches, each anchor node has an estimate of how far it is from the target node. The estimates can be achieved through different measurements of the received signals. These measurements can be received signal strength indicator (RSSI) [31], time of arrival (TOA) [32], two-way ranging (TWR) and variants [33], time difference of arrival (TDOA) [34], and hybrid ones [35–37]. The range estimates along with the known position of the anchor nodes are used to compute the position of the target node. It is worth noting that in a two-dimensional space, at least three anchor nodes are required to compute the position of the target node. In an ideal case, the position of the target node is the intersection of imaginary circles with radius of the true distance, as shown in Fig. 2a. However, in real environments, due to several errors, these circles rarely intersect at a single point, which leads to an error in the computed position of the target node. For a single position estimate, as the distance estimates deviate from the true distance, it yields errors. The sources of this distance deviation either depend on the system, such as measurement noise or dilution of precision due to anchor node placement, or on the environment such as MPP and NLOS. While the system dependent errors cannot be influenced by algorithmic processing, MPP and NLOS can be observed by channel state measurements and can hence be integrated into the algorithmic design.

3.1.2 Range Free Approaches

Range free approaches mainly comprise RF fingerprinting and AOA. In RF fingerprinting data from a target node is collected in an offline phase at access points (APs) for multiple known positions, the so-called reference points (RPs). This data is used to construct a radio map. By comparing measured data at APs with the radio map, it is possible to estimate the target node position in the online phase in real-time.

The accuracy of fingerprinting approaches depends, among others, on the number of RPs. Creating a radio map for a large area is thus a tremendous effort if a reasonable accuracy is required. Furthermore, if the position of even one AP is changed, the offline database needs to be recreated. Additionally, the high dimension of signals in most cases is another concern of fingerprinting approaches, which increases the computational complexity.

If AOA information is available at the anchor nodes, triangulation can be used to compute the position of the target node. It requires at least two anchor nodes to compute the position of the target node in a two-dimensional space. In an ideal case, the position of the target node is the intersection of imaginary lines, representing the angle estimates, as shown in Fig. 2b. However, in real environments, the lines do not cross at a single point resulting in a localisation error. Angle estimates that deviate from the true angle at a single position produce errors. Similar to the range based approaches, the error source can be either system-related or environmental, like MPP and NLOS. MPP and NLOS can be observed by channel state measurement and thus integrated into the algorithmic design, whereas system dependent errors cannot be affected by algorithmic processing.

3.2 *AI Algorithms in UWB-Based Localisation Systems*

Location accuracy, energy efficiency, and security are the most challenging problems in LBS that can be solved by AI algorithms. The ability of AI to learn useful information from input data with known or unknown statistics is its most significant advantage. As shown in Fig. 3, we briefly categorize the AI algorithms which have been used by researchers for solving the aforementioned challenges in UWB-based localisation systems. In the following, a brief background on the utilized AI algorithms is given.

3.2.1 Supervised Learning

Supervised learning is a subclass of AI-methods that refers to algorithms that build a predictive model using data points with known outputs. As a general rule, supervised approaches are applicable to problems where labels are available. Supervised learning approaches can be separated into classification and regression.

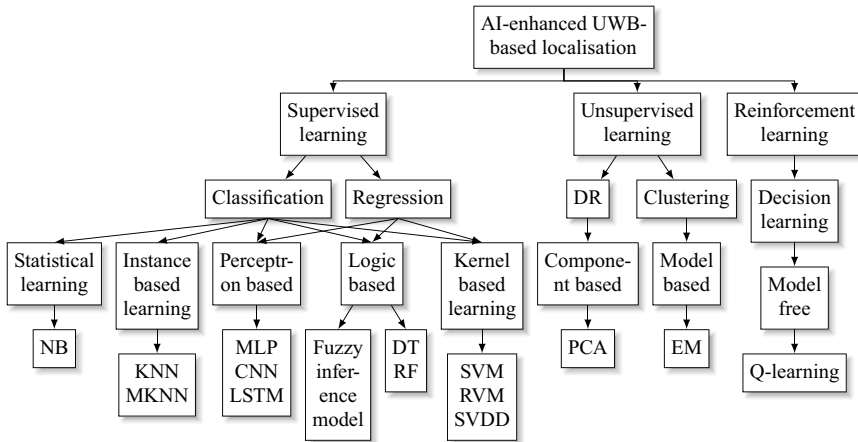


Fig. 3 Classification of AI-based methods used for UWB-localisation in the literature

Classification algorithms categorize new observations into categorical classes on the basis of training data. **Regression** algorithms model the relationship between a certain number of features and continuous target variables on the basis of training data.

Several approaches for carrying out the aforementioned tasks have been proposed in the literature which are briefly discussed in the following.

Naive Bayes (NB) is a probabilistic learning approach based on Bayes' theory used for classification. This algorithm assumes that all variables in the data set are naive, which means that they are not related to each other. The implementation of NB is fast and straightforward. However, its drawback is the need for independent features. Typically, features in real-life applications are, however, correlated, which adversely affects the performance of this classifier.

K-Nearest Neighbors (KNN) is a supervised learning algorithm that is both simple and widely used in the field of AI. Assuming that $K = 10$, the first 10 available data points in the data set with the smallest distance from a new data point are chosen to select the label of the new data point based on majority voting.

Modified K-Nearest Neighbor (MKNN) is a type of weighted KNN that selects the label of the new data point based on a modified majority voting in which the distances between the data point and its neighbors weights their votes.

Perceptron learning methods are based on neural networks and comprise a series of algorithms that try to recognize underlying relationships in data through processes that mimic the operation of a human brain. Networks including **Multi-Layer Perceptron (MLP)**, **Convolutional Neural Networks (CNN)**, and **Long Short-Term Memory (LSTM)** are available in the literature. For the sake of brevity, the network architectures are not covered in this chapter. These methods can be used either for classification or regression tasks.

Logic-based approaches can also be used for performing classification or regression tasks. A **fuzzy inference model** uses uncertainty to estimate target variables. The **Decision Tree (DT)** is a map of possible outcomes of a series of related choices or options that allows an individual to weigh possible actions in terms of costs, probabilities, and benefits. A decision tree typically starts with an initial node, after which possible outcomes are branched from it, and each of those outcomes leads to other nodes, which in turn create branches of other possibilities. This branching structure finally turns into a tree-like diagram. An alternative to DT is **Random forest (RF)**, which combines a number of decision trees, to predict the label based on the majority of votes in each tree. More trees in the forest lead to higher accuracy and avoid the problem of overfitting.

Kernel-based approaches map input data into a higher dimensional feature space using a kernel function. Then, the underlying relationships of data will be recognized in the feature space. Published kernel-based approaches for UWB-based localisation include **Support Vector Machine (SVM)**, **Relevant Vector Machine (RVM)**, and **Support Vector Data Description (SVDD)**. These methods can be used either for classification or regression tasks.

3.2.2 Unsupervised Learning

In contrast to supervised learning, unsupervised learning refers to AI algorithms for the identification of patterns in data sets that contain neither classified nor labelled data points. Without the need for human intervention, these algorithms help discover patterns or data groupings. **DR** and **Clustering** are two typical fields of unsupervised learning approaches which are used in AI-enhanced UWB-based localisation systems.

DR is a process which transforms data from a high-dimensional space into a low-dimensional space. The aim is to learn relationships between features and represent the data using so-called latent features that relate to the original features of the data. The aim is usually to reduce the complexity of a model and avoid overfitting. **Principal Component Analysis (PCA)** is a well-known unsupervised learning technique for reducing data dimensionality. It is a linear projection from a higher-dimensional input space to a lower-dimensional output space. The catch is to minimize the amount of information or variance lost when variables are removed.

Clustering is the process of dividing unlabelled data points into groups (or clusters) such that data points in a group have a greater degree of similarity than data points in other groups. **Expectation-Maximization (EM)** is a model-based clustering technique that has been used for UWB-based localisation. It is an approach for performing maximum likelihood estimation in the presence of latent variables. This is accomplished by first estimating the values of the latent variables, then optimizing the model, and then repeating these two steps until convergence is achieved.

3.2.3 Reinforcement Learning

As the name implies, reinforcement learning (RL) involves the process of learning through trial and error. RL draws its inspiration from the learning behavior of humans, which use past experiences to react to new circumstances. RL decisions are made based on received rewards and penalties. The algorithm is rewarded for correct decision and penalized for an incorrect decision. **Q-learning** is a model-free reinforcement learning algorithm that determines how useful a given action at each state is in gaining some future reward. It can manage problems regarding stochastic transitions and rewards without requiring adaptations.

4 Overview of Related Work

In this section, we briefly describe the related work in the field of UWB-based localisation that utilize AI algorithms to enhance accuracy, energy efficiency, and security. We categorize related work into three aims (according to Fig. 1): accuracy only (Table 1), accuracy along with energy efficiency (Table 2), and accuracy together with security (Table 3). The objective(s), the AI algorithm(s), the description, and the remarks are all mentioned for each related work in the tables.

The majority of work in Table 1 concentrate on NLOS detection and mitigation. In the most cases, supervised and unsupervised learning approaches have been used for NLOS detection and NLOS mitigation, respectively.

The majority of work in Table 2 concentrates on feature selection and dimensionality reduction. In the most cases, unsupervised learning approaches have been used for these purposes.

Considering security aspects in AI-enhanced UWB-based localisation systems is less investigated in the literature. As shown in Table 3, there are only two publications in which supervised learning approaches have been used for security concerns.

5 Application Example

In the following application example, which has been developed in the course of the InSecTT project, we focus on LOS/NLOS detection [8] and trustworthiness [29] objectives. We describe how KNN algorithms can effectively address these challenges.

Table 1 AI-enhanced UWB-based localisation work in literature whose aim is accuracy only

Refs.	Objective	Algorithms	Description	Remarks
[5]	Feature selection	PCA DR, DT, SVM classification	A feature-based approach that exploits ML methods for positioning with UWB CIRs	Features in complex propagation environments improve positional accuracy
[6]	NLOS detection	Q-learning	A reinforcement learning approach to select LOS connections between a tag and anchor nodes	A balance between battery life and localisation accuracy of UWB anchor nodes
[7]	NLOS detection	MLP, Boosted DT classification	Two ML methods are developed to improve NLOS detection	Experimental results from tests in a factory scenario
[8]	NLOS detection	KNN and SVM classification	The ML methods are used to demonstrate benefits of LOS/NLOS labeling method	Extensive experimental test data set has been used of-the-shelf components
[9]	NLOS detection	Naive Bayes classification	Offline phase: data is collected and pre-processed. Online phase: data is compared with the database using an NB classifier	Distance error increases as the distance between the anchors and tags increase
[10]	NLOS detection and feature selection	CNN-LSTM classification	CNN extracts the features of CIR data. Then, the CNN outputs are fed into the LSTM for classification	Extensive experimentation with different settings were considered
[11]	NLOS mitigation	SVM, MLP, CNN regression	Two DL algorithms (i.e., MLP and CNN) to improve the UWB-based localisation system	A novel localisation framework
[12]	NLOS mitigation	CNN regression	An input image generation method to localize a target node	An asymmetric environment is considered for evaluation
[13]	NLOS mitigation	SVDD regression	SVDD based regression methodology for ranging error mitigation	A one-step procedure
[14]	NLOS mitigation	LSTM regression	A feature-based localisation approach using LSTM	Evaluation of the proposed method by means of simulation
[15]	NLOS detection and mitigation	SVM regression	Based on the received waveform and the estimated distance, 7 features are selected to mitigate ranging errors	NLOS detection and mitigation in one step
[16]	NLOS detection and mitigation	LS-SVM classification and regression	LS-SVM to identify the LOS/NLOS conditions and to mitigate the ranging error bias	Evaluation of the proposed method by means of simulation

Table 2 AI-enhanced UWB-based localisation work in literature whose aim is accuracy and energy efficiency

Refs.	Objective	Algorithms	Description	Remarks
[17]	Feature selection and complexity reduction	PCA DR, MC-SVM classification	ML approaches to both localize targets and identify them	PCA to reduce dimensionality and MC-SVM to understand the location of a target within a building
[18]	NLOS detection and complexity reduction	EM-GMM clustering	An algorithm to classify LOS and NLOS components with an unsupervised learning approach	Evaluation of the proposed method by means of simulation
[19]	NLOS detection and complexity reduction	SVM, RF and deep MLP classification	Improve ranging accuracy to detect if either: LOS, NLOS or MPP	Analyzing different features and complexity
[20]	NLOS detection and mitigation, complexity reduction	RVM classification and regression	An algorithm based on RVM techniques to improve the localisation accuracy in LOS and NLOS coexisting environment	Two-step iterative with fast convergence
[21]	NLOS detection, mitigation and complexity reduction	CNN regression	Ranging error regression models with UWB CIRs	Designed for computationally restricted devices
[22]	NLOS mitigation and complexity reduction	L-STM regression	Based on distance estimations, the model predicts the user position	Different aspects of the model are analyzed
[23]	NLOS mitigation and complexity reduction	Fuzzy regression	A fuzzy inference model to estimate the position of users	The ranging error is estimated and corrected using parameters' uncertainties
[24]	NLOS mitigation and complexity reduction	ANN regression	Ranging error mitigation model with energy efficiency consideration	The ANN is trained only with LOS measurements
[25]	NLOS mitigation, feature selection, complexity reduction	DR, CNN regression	CNN-based deep learning with raw CIR input	Extensive experimentation with different settings are considered
[26]	Dimensionality reduction and NLOS mitigation	KNN classification and regression	A hybrid method (combination trilateration and fingerprinting) for UWB-based localisation	An extensive indoor measurement campaign was performed
[27]	Dimensionality reduction and NLOS mitigation	KNN, NN classification	A fingerprinting based approach along with trilateration that utilizes KNN and NN to mitigate errors	Extensive experimentation with different settings are considered
[28]	Dimensionality reduction and complexity reduction	PCA DR, SVM classification	A localisation approach using Grid Search algorithm based on PCA-SVM scheme	PCA to reduce dimensionality and SVM to estimate the position of users

Table 3 AI-enhanced UWB-based localisation work whose aim is accuracy and security

Refs.	Objective	Algorithms	Description	Remarks
[29]	NLOS mitigation, trustworthiness	modified KNN and modified RF regression	ML methods used to estimate error correction together with a trustworthiness score	Trustworthiness score allows to select measurements on which correction algorithm has good performance
[30]	Intrusion mitigation	LSTM regression	A deep learning approach to localize a tag with security consideration	Sensor measurement error (caused by attacks) in localisation is considered

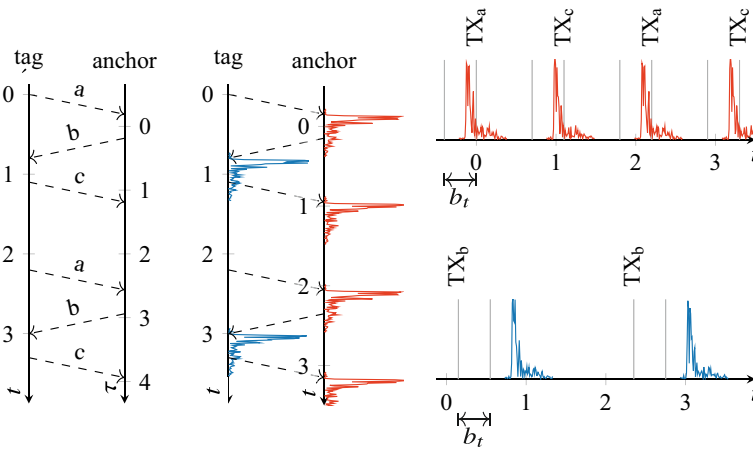


Fig. 4 The channel impulse response is estimated at each node (left side) from which the received timestamp is derived (right side) (from [8])

5.1 KNN for LOS/NLOS Detection

As stated in [8], NLOS conditions in UWB-based localisation systems lead to errors and must be detected. This is a binary classification problem with LOS and NLOS classes. From a data perspective, [8] uses channel impulse response (CIR) estimates to calculate the received timestamps as input data (see Fig. 4).

LOS and NLOS conditions are distinguished by comparing the received timestamps. The timestamps for all LOS cases are similar and the same is true for NLOS. Thus, KNN is well suited for this classification task. However, several key points have to be considered. Firstly, the used distance metric affects performance. Choosing the K value as hyperparameter is another concern. If one chooses a too small value, it will lead to unstable decision boundaries. However, a large value can be computationally very expensive. As a rule of thumb, the optimal K value is usually the square root of n , where n is the number of samples.

To formulate the KNN algorithm for this LOS/NLOS detection problem, assume that we have training data D given by

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}, \quad (1)$$

where x_i and y_i are the sample and the label of the i th input data, respectively. It is obvious that the samples are the vector of timestamps and the labels are either LOS or NLOS.

For a test point $x \in \mathbb{R}$, a set $S_x \subseteq D$ is defined as a set of K neighbors. Using the function $dist$ that computes the distance between two points in \mathbb{R} , a set S_x of size K can be defined as:

$$dist(x, x') \geq \max_{x'' \in S_x} dist(x, x''), \forall x' \in D \setminus S_x. \quad (2)$$

Peterseil et al. [8] uses the KNN algorithm to detect weak NLOS conditions in the double sided two way ranging (DS-TWR) scheme and assigns a correct label to the measurement data. In [8], a packet-wise approach is proposed instead of a cyclic approach [38] which outperforms the cyclic approach by up to 15 cm improvement of measurement error in a collected data set by considering only those measurement data, which are classified as taken from LOS conditions.

5.2 KNN for Error Mitigation and Trustworthiness

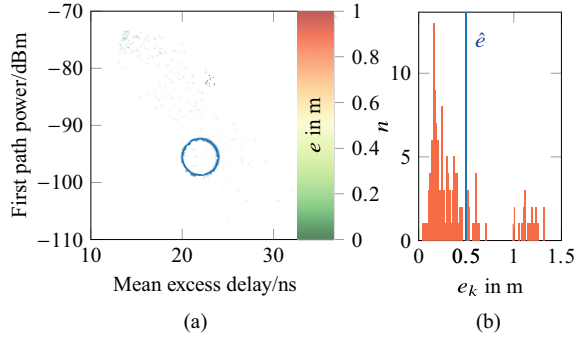
NLOS mitigation is another objective in UWB-based localisation systems that can be addressed with KNN-based algorithms. The goal is to estimate the measurement error due to the NLOS condition and use it to correct the raw measurement. As we have seen, a NLOS condition can be detected by the KNN algorithm and thus it can be compensated for in the positioning computation. This translates in a regression problem. [29] uses first path power and mean excess delay derived from the received signal and the CIR, respectively.

The KNN algorithm assumes that the measurement error of training and testing samples is similar if they are close in feature space. The predicted measurement error

$$\hat{e} = \frac{1}{K} \sum_{i=1}^K e_i \quad (3)$$

is therefore calculated as the average error of the K nearest neighbors in the feature space.

Fig. 5 **a** Feature space; for an example feature measurement the circle depicts the KNN-neighborhood (for the sake of clarity, errors higher than 1 m are truncated to 1 m) and **b** Histogram of label values from neighborhood (from [29])



In practice it can happen that certain areas of the feature space contradict each other, leading to error-prone estimations. In [29], a modified KNN algorithm (mKNN) was proposed by additionally evaluating the standard deviation of the neighborhood

$$\hat{\sigma} = \sqrt{\frac{1}{K} \sum_{i=1}^K (e_i - \hat{e})^2}. \tag{4}$$

In Fig. 5 one can distinguish between areas (or neighborhoods) with mostly low measurement error represented by green dots, areas with mostly high measurement error represented by red dots and areas with “mixed” error in which red and green dots appear with similar density. In neighborhoods with mostly green or red dots the standard deviation of the measurements is low because the condition is clearly either LOS or NLOS. In “mixed” neighborhoods, however, the LOS/NLOS condition is unclear and thus the standard deviation is high. Thus, by mapping σ to a range between 0 and 1, it can be interpreted as a trustworthiness measure, indicating whether the estimation of the measurement error, i.e. the correction term, is likely to be accurate or not.

Peterseil et al. [29] shows that the modified KNN algorithm reduces the ranging root mean square error from 36 cm if the whole data set is used, to 19 cm those 70% of the collected data with highest trustworthiness score are used.

6 Conclusion

With the growing demand for accurate and reliable location information in wireless networks, the use of UWB technology becomes increasingly important. UWB-based localisation systems face a variety of challenges that can be addressed by AI algorithms. All AI algorithms covered in this chapter aim to improve accuracy, with or without consideration of energy efficiency or security aspects. The larger part of the related work uses AI algorithms to improve the localisation accuracy. Also research

on energy efficiency together with appropriate accuracy has been presented in several publications and was shown in an application example developed in the scope of the InSecTT project. However, AI algorithms to improve the security of localisation systems have only been covered in a few works and require more research.

References

1. Leu, P., Camurati, G., Heinrich, A., Roeschlin, M., Anliker, C., Hollick, M., Capkun, S., Classen, J.: Ghost peak: practical distance reduction attacks against HRP UWB ranging. *CoRR* 1–15 (2021). <https://doi.org/10.48550/arXiv.2111.05313>
2. IEEE Standard for Low-Rate Wireless Networks—Amendment 1: Enhanced Ultra Wideband (UWB) Physical Layers (PHYS) and Associated Ranging Techniques, IEEE Std 802.15.4z-2020 (Amendment to IEEE Std 802.15.4-2020), pp. 1–174 (2020) <https://doi.org/10.1109/IEEESTD.2020.9179124>.
3. Denis, B., Keignart, J., Daniele, N.: Impact of NLOS propagation upon ranging precision in UWB systems. In: *IEEE Conferences on Ultra Wideband Systems and Technical*, pp. 379–383 (2003). <https://doi.org/10.1109/UWBST.2003.1267868>.
4. Capkun, S.: Physical-layer attacks and their impact on wireless networks: two case studies. In: *Proceedings of 15th ACM Conferences Security and Privacy in Wirel. and Mobile Networks*, pp. 1–1 (2022). <https://doi.org/10.1145/3507657.3528562>.
5. Kram, S., Stahlke, M., Feigl, T., Seitz, J., Thielecke, J.: UWB channel impulse responses for positioning in complex environments: a detailed feature analysis. *Sensors* 5547–5572 (2019). <https://doi.org/10.3390/s19245547>
6. Hajiakhondi-Meybodi, Z., Mohammadi, A., Hou, M., Plataniotis, K.N.: DQLEL: deep Q-learning for energy-optimized LoS/NLoS UWB node selection. *IEEE Trans. Signal Proc.* 2532–2547 (2022). <https://doi.org/10.1109/TSP.2022.3171678>
7. Krishnan, S., Santos, R.X.M., Yap, E.R., Zin, M.T.: Improving UWB based indoor positioning in industrial environments through machine learning. In: *2018 15th International Conference Control, Automation, Robotics and Vision (ICARCV)*, pp. 1484–1488 (2018). <https://doi.org/10.1109/ICARCV.2018.8581305>
8. Peterseil, P., Märzinger, D., Etzlinger, B., Springer, A.: Labeling for UWB ranging in weak NLOS conditions. In: *International Conference on Localization and GNSS (ICL-GNSS)*, pp. 1–6 (2022). <https://doi.org/10.1109/ICL-GNSS54081.2022.9797024>
9. Che, F., Ahmed, A., Ahmed, Q.Z., Zaidi, S.A.R., Shakir, M.Z.: Machine learning based approach for indoor localization using ultra-wide bandwidth (UWB) system for industrial internet of things (IIoT). In: *International Conference UK-China Emerging Technologies (UCET)*, pp. 1–4 (2020). <https://doi.org/10.1109/UCET51115.2020.9205352>
10. Jiang, C., Shen, J., Chen, S., Chen, Y., Liu, D., Bo, Y.: UWB NLOS/LOS classification using deep learning method. *IEEE Commun. Lett.* 2226–2230 (2020). <https://doi.org/10.1109/LCOMM.2020.2999904>
11. Nosrati, L., Fazel, M.S., Ghavami, M.: Improving indoor localization using mobile UWB sensor and deep neural networks. *IEEE Access* 20420–20431 (2022). <https://doi.org/10.1109/ACCESS.2022.3151436>
12. Nguyen, D.T.A., Lee, H.G., Jeong, E.R., Lee, H.L., Joung, J.: Deep learning-based localization for UWB systems. *Electronics* 1712–1729 (2020). <https://doi.org/10.3390/electronics9101712>
13. Tian, S., Zhao, L., Li, G.: A support vector data description approach to NLOS identification in UWB positioning. *Math. Probl. Eng., Art. no. 963418* (2014). <https://doi.org/10.1155/2014/963418>

14. Poulouse, A., Han, D.S.: Feature-based deep LSTM network for indoor localization using UWB measurements. In: International Conference Artificial Intelligence in Information and Communication (ICAIIIC), pp. 298–301 (2021). <https://doi.org/10.1109/ICAIIIC51459.2021.9415277>
15. Wymeersch, H., Marano, S., Gifford, W.M., Win, M.Z.: A machine learning approach to ranging error mitigation for UWB localization. in IEEE Trans. Commun. 1719–1728 (2012). <https://doi.org/10.1109/TCOMM.2012.042712.110035>
16. Marano, S., Gifford, W.M., Wymeersch, H., Win, M.Z.: NLOS identification and mitigation for localization based on UWB experimental data. IEEE J. Sel. Areas Commun. 1026–1035 (2010). <https://doi.org/10.1109/JSAC.2010.100907>
17. Rana, S.P., Dey, M., Siddiqui, H.U., Tiberi, G., Ghavami, M., Dudley, S.: UWB localization employing supervised learning method. IEEE 17th International Conference Ubiquitous Wireless Broadband (ICUWB), pp. 1–5 (2017). <https://doi.org/10.1109/ICUWB.2017.8250971>
18. Fan, J., Awan, A.S.: Non-line-of-sight identification based on unsupervised machine learning in ultra wideband systems. IEEE Access 32464–32471 (2019). <https://doi.org/10.1109/ACCESS.2019.2903236>
19. Sang, C.L., Steinhagen, B., Homburg, J.D., Adams, M., Hesse, M., Rückert, U.: Identification of NLOS and multi-path conditions in UWB localization using machine learning methods. Appl. Sci. 3980–4004 (2020). <https://doi.org/10.3390/app10113980>
20. Van Nguyen, T., Jeong, Y., Shin, H., Win, M.Z.: Machine learning for wideband localization. IEEE J. Sel Areas Commun. 1357–1380 (2015). <https://doi.org/10.1109/JSAC.2015.2430191>
21. Bregar, K., Mohorčič, M.: Improving indoor localization using convolutional neural networks on computationally restricted devices. IEEE Access 17429–17441 (2018). <https://doi.org/10.1109/ACCESS.2018.2817800>
22. Poulouse, A., Han, DS.: UWB indoor localization using deep learning LSTM networks. Appl. Sci. 2076–3417 (2020). <https://doi.org/10.3390/app10186290>
23. Meghani, S.K., Asif, M., Awin, F., Tepe, K.: Empirical based ranging error mitigation in IR-UWB: a fuzzy approach. IEEE Access 33686–33697 (2019). <https://doi.org/10.1109/ACCESS.2019.2904201>
24. Shenoy, M.V., Karuppiah, A., Manjarekar, N.: A lightweight ANN based robust localization technique for rapid deployment of autonomous systems. J. Ambient. Intell. Hum. Comput. 2715–2730 (2020). <https://doi.org/10.1007/s12652-019-01331-0>
25. Angarano, S., Mazzia, V., Salvetti, F., Fantin, G., Chiaberge, M.: Robust ultra-wideband range error mitigation with deep learning at the edge. Eng. Appl. Artif. Intell. 104278–104286 (2021). <https://doi.org/10.1016/j.engappai.2021.104278>
26. Djosic, S., Stojanovic, I., Jovanovic, M., Nikolic, T., and Djordjevic, G.L.: Fingerprinting-assisted UWB-based localization technique for complex indoor environments. Expert Syst. Appl. 114188–114201 (2021). <https://doi.org/10.1016/j.eswa.2020>
27. Murari, R.: Practical and robust approach for a neural networks based indoor positioning system using ultrawide band (Doctoral dissertation, Ryerson University) (2020)
28. Zhang, L., Li, Y., Gu, Y., Yang, W.: An efficient machine learning approach for indoor localization. China Commun. 141–150 (2017). <https://doi.org/10.1109/CC.2017.8233657>
29. Peterseil, P., Etlzinger, B., Märzinger, D., Khanzadeh, R., Springer, A.: Data trustworthiness for UWB ranging in IoT. In: IEEE Globecom Workshops (GC Wkshps), pp. 939–944 (2022). <https://doi.org/10.1109/GCWkshps56602.2022.10008777>
30. Xue, Y., Su, W., Wang, H., Yang, D., Jiang, Y.: DeepTAL: deep Learning for TDOA-based asynchronous localization security with measurement error and missing data. IEEE Access 122492–122502 (2019). <https://doi.org/10.1109/ACCESS.2019.2937975>
31. Venkatesh, S., Buehrer, R.M.: Non-line-of-sight identification in ultra-wideband systems based on received signal statistics. IET Microw. Antennas Propag. 1120–1130 (2007). <https://doi.org/10.1049/iet-map:20060273>
32. Alsindi, N.A., Alavi, B., Pahlavan, K.: Measurement and modeling of ultrawideband TOA-based ranging in indoor multipath environments. IEEE Trans. Veh. Tech. 1046–1058 (2009). <https://doi.org/10.1109/TVT.2008.926071>

33. Kim, H.: Double-sided two-way ranging algorithm to reduce ranging time. *IEEE Commun. Lett.* 486–488 (2009). <https://doi.org/10.1109/LCOMM.2009.090093>
34. Xu, J., Ma, M., Law, C.L.: Position estimation using UWB TDOA measurements. In: 2006 IEEE International Conferences Ultra-Wideband, pp 605–610 (2006). <https://doi.org/10.1109/ICU.2006.281617>
35. Taponecco, L., D’Amico, A.A., Mengali, U.: Joint TOA and AOA estimation for UWB localization applications. *IEEE Trans. Wirel. Commun.* 2207–2217 (2011). <https://doi.org/10.1109/TWC.2011.042211.100966>
36. Luo, Y., Law, C.L.: Indoor positioning using UWB-IR signals in the presence of dense multipath with path overlapping. *IEEE Trans. Wirel. Commun.* 3734–3743 (2012). <https://doi.org/10.1109/TWC.2012.081612.120045>
37. Shang, F., Champagne, B., Psaromiligkos, I.N.: A ML-based framework for joint TOA/AOA estimation of UWB pulses in dense multipath environments. *IEEE Trans. Wirel. Commun.* 5305–5318 (2014). <https://doi.org/10.1109/TWC.2014.2343634>
38. Neiryneck, D., Luk, E., McLaughlin, M.: An alternative double-sided two-way ranging method. In: 13th Workshop on Positioning, Navigation and Communications (WPNC), pp. 1–4 (2016). <https://doi.org/10.1109/WPNC.2016.7822844>.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Industrial Applications

Approaches for Automating Cybersecurity Testing of Connected Vehicles



Stefan Marksteiner, Peter Priller, and Markus Wolf

Abstract Vehicles are on the verge building highly networked and interconnected systems with each other. This requires open architectures with standardized interfaces. These interfaces provide huge surfaces for potential threats from cyber attacks. Regulators therefore demand to mitigate these risks using structured security engineering processes. Testing the effectiveness of this measures, on the other hand, is less standardized. To fill this gap, this book chapter contains an approach for structured and comprehensive cybersecurity testing of contemporary vehicular systems. It gives an overview of how to define secure systems and contains specific approaches for (semi-)automated cybersecurity testing of vehicular systems, including model-based testing and the description of an automated platform for executing tests.

1 Introduction

Mobility is a high priority in our society. Statistics report global annual car sales between 60 and 75 million¹ during recent years. According to the European Automobile Manufacturers' Association (ACEA), just in Europe approximately 350 million cars are currently in use [8], and the number grows to beyond 1 billion for a world-wide estimation. Cars are ubiquitous, for many families and businesses around the world, since decades. What has changed, however, is the fact that today's vehicles have become complex IT systems, often also called "computers on wheels". Modern

S. Marksteiner (✉) · P. Priller · M. Wolf
AVL List GmbH, Graz, Austria
e-mail: stefan.marksteiner@avl.com

P. Priller
e-mail: peter.priller@avl.com

M. Wolf
e-mail: markus.wolf@avl.com

S. Marksteiner
Mälardalen University, Västerås, Sweden

¹ <https://www.statista.com/statistics/200002/international-car-sales-since-1990/>.

cars run 100+ million lines of source code (MLOSC), and host complex computer networks both internally (in-vehicle networks) and externally. Many modern cars are now connected via the Internet to (maybe even multiple) cloud services, as well as to cellular networks (3G, LTE, 5G), and to specific vehicular networks (also known as car-to-car (C2C) or vehicle-to-everything (V2X), like ITS-G5). And that's not all: most vehicles also provide local networking capabilities (also called personal area networking, PAN). Typically based on WIFI and Bluetooth, it is used to connect to users' personal devices like smart phones and tablets, or to their home WLAN. To complement that already impressive array of wireless communication interfaces, some car manufacturers (or Original Equipment Manufacturers—OEMs) might add ultra-wide band (UWB) radios to communicate with car access systems like owner's keys or keycards. In addition, advanced driver assistance systems (ADAS) and future fully automated driving (AD) capabilities add GNSS receivers (Global Navigation Satellite System), TMC receivers (Traffic Message Channel), and active radar systems. Modern vehicles combine deeply complex software with exposure to a wide range of wireless networking technologies to both public and closed networks). In cybersecurity, this is called opening a large attack surface. This is worsened by the fact that vehicles are exposed for a much longer time than, e.g., personal computers (PC) or mobile devices like smart phones. Cars are in operation for some 15 years and more, which increases the threat that a vulnerability is found, shared, and at some point in time exploited by an attack. With such significant high exposure, let's consider potential threats which could evolve from malign attacks. Vehicles are highly dynamic (by nature), provide high levels of energy (storing 100 kWh and more), are valuable (sometimes beyond 100 k€) and exist as worldwide accessible objects in public, thus unrestricted places. When exploited by an attack taking over remote control, vehicles could become dangerous weapons, for both passengers inside, and for other road participants. Worse, if groups of vehicles would come under attacker's control, they could be used to stage threats on city or even national level. State-sponsored attackers could stage war or terror attacks of not-yet-seen scale. Other scenarios might be less about harming humans, but could include denial-of service on single vehicles (e.g., owners cannot use their vehicles unless a ransom is paid) or on fleet level (e.g., blocking important road infrastructure, threatening whole communities, and some serious damage of a brand's reputation). And of course, there is simple car theft. Data privacy is also an important aspect. Modern cars might "know" quite a lot about their users, including their past and present locations, driving habits, additional passengers, anything spoken in the vehicle, attention level while driving, contact information like phone numbers from connected personal devices, etc. An attack could therefore retrieve quite a lot of personal information and thus become considerable value to attackers. While not all of these threats have been discussed widely in public, the automotive industry is very much aware of it, and has stepped up efforts in designing more secure systems in cars, and establishing secure life cycle processes to provide necessary updates to fix vulnerabilities. An important part of securing these vehicular systems is the verification and validation of the effectiveness of taken security measures through testing. This testing needs to be done continuously through the life cycle (as new exploits might come up over time), and also

as updating a system (or just a part of it) might alter its behavior in a way relevant to its security. In essence, (cyber)security testing must assure a system to display a small attack surface, be resilient and (possibly) to fix vulnerabilities before they are exploited in the wild.

The remainder of the chapter is structured the following way: Sect. 2 contains the current state of the art and related work. Section 3 contains measures for securing automotive systems. Section 4 contains specific approaches for automated cybersecurity testing of vehicular systems, including model-based testing and the description of an automated platform for executing tests. Section 5, eventually concludes this chapter.

2 State of the Art and Related Work

The automotive industry can draw from experience in other domains regarding security testing. General IT (managing e.g., corporate networks and IT systems) has established a history of penetration testing (abbreviated: pen testing), as simulated, authorized cyber-attacks. Typically executed by cybersecurity experts (acting as “white-hat hackers”), the goal is to identify weaknesses by letting these experts try to hack into the system under test (SUT) under pre-defined constraints (e.g., no physical access, no permanent harm), typically within a defined time window. If successful, these tests can thereby discover and document weaknesses. Translated to automotive industry, several companies offer similar pen-testing as a service on different levels (component, system, vehicle). While pen-testing might provide highly valuable insights into what level of security has been achieved for the vehicle, and might even uncover previously unknown vulnerabilities, it suffers from limited scalability and repeatability, as it is driven by and dependent on human experts. Security experts have toolboxes with highly effective tools (like the open source Metastability framework²), but often need to supervise and configure these tools, and to adapt existing or write new scripts for complete attack chains to match a specific SUT. This requires skills and labor, and often involves considerable costs, which clearly limits scalability. Due to the sheer complexity of automotive software code (100+ MLOC), it is also quite challenging for the experts to correctly hypothesize vulnerabilities, and to select (and execute) the most effective attacks, given the limited time available. This might heavily depend on expertise of the human testers, further limiting repeatability and comparability between pen tests campaigns. The threat of cyber attacks by adversaries has, however, also been recognized by standards and regulatory bodies. The United Nations Economic Council for Europe (UNECE) has issued a regulation (R 155 [36]) that prescribes the installation of a cybersecurity management system (CSMS). A CSMS is a process framework that accompanies the automotive development process over the complete life cycle and assures cybersecurity in every phase. Consequently, the International Organization for Standardization (ISO) and the Society of Automotive

² <https://github.com/rapid7/metasploit-framework>.

Engineers (SAE) have issued a joint standard (ISO/SAE 21434 [16]) that defines such a CSMS. As testing guidelines in these standards are somewhat underrepresented in contrast to security engineering, a structured approach is needed, e.g., as defined in [23, 24]. It further became clear that in order to establish dependable security covering all variants of vehicle lines in their full life cycle, supporting the upcoming accelerated software development cycles (automotive DevOps), an advanced process based on smart automation was required, as suggested in [6].

3 Automotive Cybersecurity Lifecycle Management

In order to maintain secure (and through, security-related impacts, also safe) vehicular systems, the respective system needs a security concept. The cybersecurity testing (see Sect. 4) will eventually validate and verify the effectiveness of that concept. To establish a security concept for the complete life cycle of a vehicle for testing, we mainly rely on five pillars:

1. Threat Modeling (see Sect. 3.1)
2. Variant Management
3. Vulnerability Assessment
4. Automated Test Generation (see Sect. 4.2)
5. Process Governance.

Threat modeling (see Sect. 3.1) is a widely proliferated technique in the automotive industry, mainly as part of a threat analysis and risk assessment (TARA) process [39].

As an OEM's fleet contains various vehicle model configurations, all of which contain tens of ECUs all of which again may display different hardware and software versions, keeping track of this potentially vast number of variants is crucial to determine the security posture of each member of the fleet. Our approach to tackle this problem is to use calibration data management that links technical attributes with software calibrations, to keep track of all ECU variations over the system's life cycle [5, 31]. This system, CRETA, contains exhaustive information about the variants, including their ECU firmware binaries.

This allows for the stored firmwares to be subsequently analyzed, generating a digital model of the software. To do so, firstly the firmware is extracted by iterating through the file tree, using an extraction algorithm and validating the extraction's correctness. The extracted software undergoes a composition analysis that pre-processes executables and normalizes the software in order to compare to a large database of mapped components, identified e.g. by file paths, file names, and characteristic strings in the software or configuration data, yielding a Software Bill-of-Materials (SBOM). Subsequently, the model is analyzed for security properties using pattern recognition. Patterns of known attacks from Common Vulnerabilities and Exposures (CVEs) are compared with each identified software library in the SBOM. Furthermore, the model undergoes a binary code analysis to find vulnerabilities not found in public databases: the binary is mapped in data and code sections, the code is then disassem-

bled and later mapped into an intermediate language (for normalizing purposes) that allows for reconstructing the functions, analyzing the parameters and stack behavior and building control and data flows [11]. This matching, for instance, is able to identify common flaws like buffer overflows and, hence, is able to uncover zero-day vulnerabilities in software in a black box setting. Thirdly, patterns for proliferated code guidelines and relevant security standards are implemented, allowing for compliance checking against a given set of standards. This analysis, paired with full life cycle-coverage of the variants, allows for dealing with the parts lists and vulnerability management requirements mentioned above, as well as for verifying security requirements.

Vulnerabilities found in the code through pattern matching, however, are not necessarily exploitable for a variety of reasons. For instance, the location in the code could not be reachable, the impact of the vulnerability could be nullified through write protection of the memory or file system, or the interface might be protected by access controls. Therefore, the generated model also allows for model-based cybersecurity test case generation by using either the generated behavior model for model checking or by directly using the found patterns as basis for vulnerability exploitation [22]. We also aim for deriving test cases from threat modeling with a certain degree of automation (see Sect. 4.2).

To govern the process we developed our tool, FUSE, that guides activities of a given standard and provides standards-compliant documentation given the necessary input. We implemented ISO/SAE 21434 [16] and UNECE R155 [36] (as well as ISO 26262 [15], ISO 25119 [14]). The modeled objectives from the standards allow for providing all necessary artifacts for performing a review or audit, as well as keeping track of the conformance to relevant standards inside the development project.

3.1 Threat Modeling

One key element of cybersecurity analysis in all life cycle phases is threat modeling. This technique for security analysis is around for many years and well proliferated. It basically consists of modeling the information flows in an SUT and consequently examining them in a comprehensive way, e.g., via STRIDE or a similarly structured method [34].

Numerous software capable of performing a thread modeling process exists, but prior to ThreatGet none was specifically developed for embedded or IoT systems. ThreatGet is a software tool developed by Austrian Institute of Technology (AIT) and based on Microsoft Enterprise Architect, a commonly used platform for systems model engineering [7].

It is used to examine models, objects, connections and charts in a system to enable iterative threat and risk analysis, covering the following categories:

- Actor,
- Sensor,

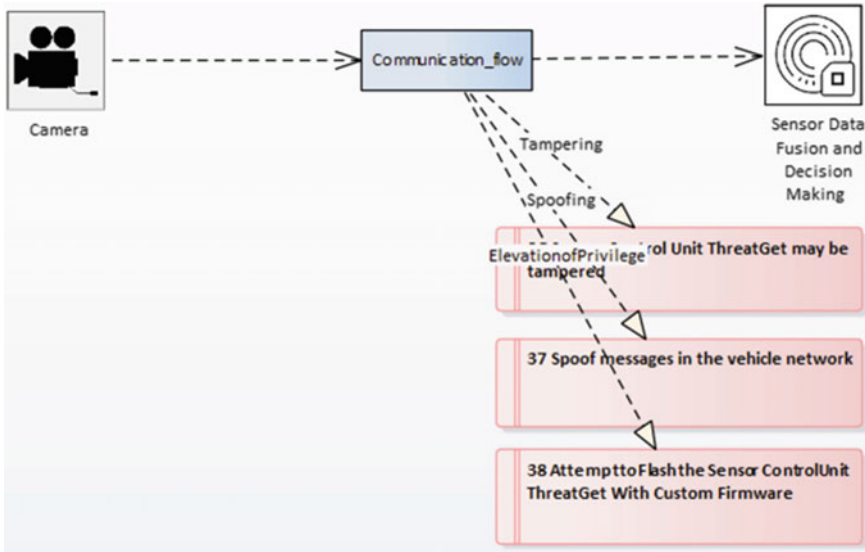


Fig. 1 A list of found threats between the camera and the sensor data fusion and decision making [7]

- Vehicle Unit,
- Data Store,
- Communication Interface,
- Communication Flow.

Objects and connections in ThreatGet have so called tagged values at creation time. These describe analysis or security relevant properties of elements. It is recommended for users to extend the properties in addition to already proposed tagged values. Additionally, a database is used in the background that contains objects, which can also be extended by a user [7].

As an application example, Fig. 1 shows the threat diagram of a communication flow inside ThreatGet. In this case, the environment data from the camera is directed to the “Sensor Data Fusion and Decision Making” unit. After all diagrams are completed, a threat-overview is derived. An automatic risk evaluation consists of suggested values and can be adapted in a manual risk evaluation. In this step it is possible to rate the impact and occurrence of a threat at different levels and afterwards results can be exported in a report [7].

4 Cybersecurity Testing

In order to assure the cybersecurity of automotive systems and provide evidence for the appropriateness and effectiveness of security measures (according to a cybersecurity management system), rigorous, structured and comprehensible testing is

necessary [36]. Therefore a structured process, aligned with ISO/SAE 21434 [16] is recommendable. Such a process for testing could contain the following activities [24]:

1. Item Definition
2. Threat Analysis and Risk Assessment
3. Security Concept Definition (mainly including the test targets)
4. Test Planning and Scenario Development
 - a. Penetration Test Scenario Development
 - b. Functional and Interface Test Development
 - c. Fuzz Testing Scenario Development
 - d. Vulnerability Scanning Scenario Development
5. Test Script Development
 - a. Test Script Validation
6. Test Case Generation
 - a. Test Environment Preparation
7. Test Case Execution
8. Test Reporting.

While items 1–3 correspond to a threat modeling process (see Sect. 3.1), the rest of them are the core testing process. To increase testing efficiency, these steps could be partially automated using model and learning-based approaches that can execute test planning and execution steps [6]. Here, the steps can be summarized into *concept design*. Item 4 forms *V & V planning*, while items 5 and 6 can be subsumed under *V & V Methods*. Finally, items 7 and 8 forms *V & V execution*. In between the planning and the methods, steps for automation can take effect: models from the concept design can be validated in an automated way and single components can be modeled using automated learning techniques and verified using methods from the V & V methods. An example of this used in the InSecTT project is described in Sect. 4.1. The full approach as described above consists of the following steps [6]:

1. Concept Design
2. V&V Planning
3. Model Validation
4. Model Learning
5. V&V Methods
6. V&V Execution.

4.1 Learning-Based Testing

Following the approach described above, we use learning, more concretely active automata learning to derive a model of a system [37]. The methodology uses a learner-teacher system where an all-knowing teacher answers the learning system queries

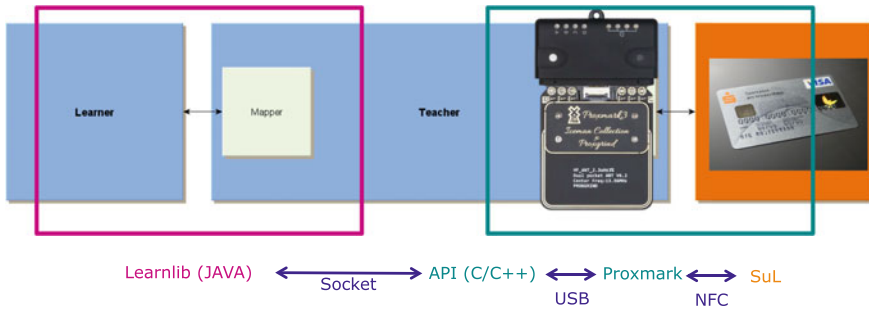


Fig. 2 NFC automata learning setup [26]

about the SUT, in the context of cyber-physical systems ordinarily by providing the output to a series of inputs. The learner tries to infer a state machine from the given information. Once it has a hypothesis of a state machine that describes the observed behavior, it presents it to the teacher who then acknowledges the hypothesis as correct or gives a counterexample. This again, in real-world situations of black-box learning will mostly be simulated by conformance testing algorithms: if conformance is shown, the hypothesis is assumed as correct, otherwise a failing test sequence serves as a counterexample. The counterexample is taken as new input to refine the hypothesis and the learning continues until no more counterexamples are found. This algorithm has been first formulated by Angluin [3] and has experienced significant improvements since (e.g., [17, 32]).

In accordance with the process outlined in Sect. 4, we use this technique to infer a model of a component. As a proof-of-concept we test a car access system based on Near-Field Communications (NFC). The testing setup consists on a learner (as described above) based on the *Learnlib* Java library [18] and a Proxmark NFC device [12] with an respective API that enables us to learn a model of the ISO 14443-3 NFC handshake protocol [13]. Figure 2 shows an overview of this setup. The used learning setup allows for inferring a state machine of the protocol and compare it to the specification in the standard to check its conformance. Figure 3 shows the learned model of the actual SUT (and NXP test card of a car access system prototype). Further use of the model is to do actual model checking or to use the model as an input for guided fuzz testing.

4.2 Model-Based Test Case Generation

On a macroscopic level, a model of a complete vehicle as defined in the threat model (see Sect. 3.1) has to be explored in order to identify single components and generate test cases based on an attack tree [30, 33], a petri net [29, 38], or similar. If the SUT is modeled manually and, therefore, the components are known, this is trivial. If the

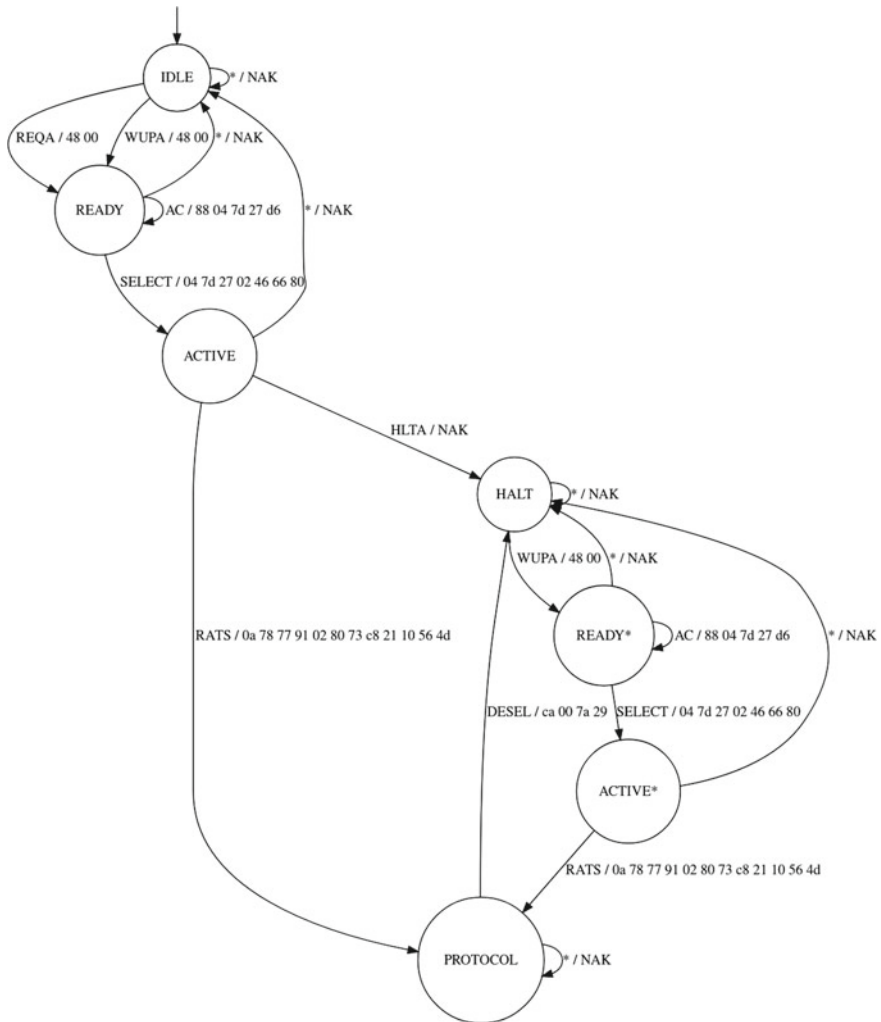


Fig. 3 Learned model of an NXP NFC test card

setting is a black or grey box situation, we follow the approach to assume a generic model as starting point and test various components of the model by, e.g., send certain CAN messages for enumeration or try out an exploit that is known to affect a very broad variety of systems. Based on a comparison of the expected and actual output of the test, one can narrow down the set of likely components and system architectures (as described in [25]), e.g., based on SAT solving [28].

In order to generate test cases on a component level, a model must be transformed into a form that can be examined using a model checker (e.g. the Rebeca model checker [35] or SLAM [4]). Violations of the specification found by a model checker

point towards an interesting position for a test case that could be extrapolated out of the traces leading to the respective states. There is also work regarding a toolchain using the UPPAAL framework [1].

Subsequently properties defining the security of a system shall be defined and used in the model checking. For c) where the model checking fails, a security problem might be present. The trace of the counter example can help in building a test case. Moreover, the input sequences used for the automata learning of the model shall be used to make test cases for the actual system-under-test. Using the traces as test vectors eliminate false positives from the model checking, as the exploitability of specification violations is test on the actual system. To concrete the abstract input, fuzzing techniques may be used [2].

4.3 Testing Platform

To realize the testing in the faction outlined in Sect. 4, a testing framework was developed and implemented. The high-level architecture was derived from the approach outlined in [23]. It has been adapted to suit the need of performing test in any phase of the product life cycle by adding co-simulation techniques into the testing framework architecture (see Fig. 4 for an overview). The core component is a Security Testing Framework (see Sect. 4.4). It gains test cases from a generation engine that is fed by two sources: security functional tests from security requirements and penetration test attack vectors that have been tried out before (see description in Sect. 4.4) from a library. The core framework executes the attacks directly onto the SUT or into a co-simulation platform (indicated as *framework interfaces* in the figure) that interconnects various simulation parts: environment (i.e. other vehicles' and infrastructure's interference), network (generating mainly ITS-G5 traffic), channel (capable of simulating various physical layer signals as well as emitting them physically) and application (Sect. 4.4 contains an example with a platooning application). This way, each component can be stimulated the same way regardless if it is a physical or simulated component.

4.4 Automated Test Execution

For test execution, the test cases that were derived as described in previous chapters are fed into the automated test execution environment. Test cases are either manually written or generated in the ALIA DSL [40] format, which aims to provide an abstract and system agnostic representation of logical steps in a test case. Out of the main test-script and its included sub-scripts (containing frequently occurring blocks that handle a specific task such as opening a listener) a JSON Object is generated. These test case descriptions in DSL and JSON format are stored into a Database and can be accessed through the Orchestration Application; a platform independent web application that

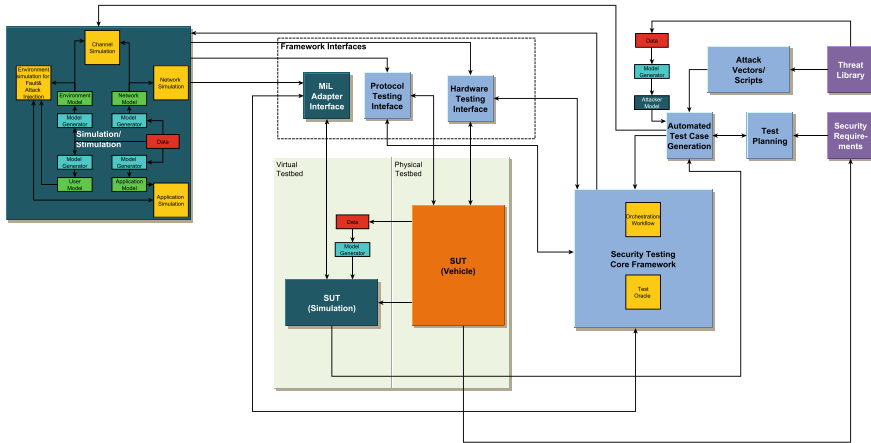


Fig. 4 Overview of the automotive cybersecurity testing framework’s high-level architecture

allows a user to manage information about the current SUT, schedule test execution and review results. This Orchestration Application then sends the test cases that the user wants to execute to the Execution Engine (AXE) and afterwards generates a report out of the received output from the AXE and the Test Oracle. The AXE is a Python based software that runs on an instance of Kali Linux and utilizes a variety of different interfaces, libraries and other software tools to perform a test case execution. It takes either a single test case or a structured collection of tests as input in JSON format and starts to subsequently execute contained steps. Figure 5 shows an overview of this architecture. This modular approach allows not only to target a specific SUT but also to control and parameterize whole (semi-) virtual SUT environments to manage SUT-behavior during a test scenario. Furthermore, it is possible to define and address different processes for tool execution which enables for example to host a malicious server, start a netcat listener and execute exploit code sequentially in a single test and afterwards perform code execution in an obtained reverse shell in the listener process.

One proof-of-concept use case implemented in the framework was security testing of the Ensemble platooning protocol [19] in a simulated environment. The concrete setup consisted of two truck simulations running on low-cost hardware connected via physical ITS-G5 [9] connection via Cohda modems. Another modem is used as an adversary to eavesdrop and interfere with the connection. The testing framework is able to start the simulation, so that the simulated trucks form a platoon. The actual test consists of (a) listening to the communications (b) distilling the session key out of a package (c) cracking the key (for testing purposes, the key was reduced to eight bits) (d) injecting a malicious message to disband the platoon. Figure 6 shows an overview of this setup. The result was that the injection failed for timing reasons, because the platoon keep-alive messages were sent in such a high frequency that they interfered with the break-up sequence. Even with reduced key (from AES-256 down

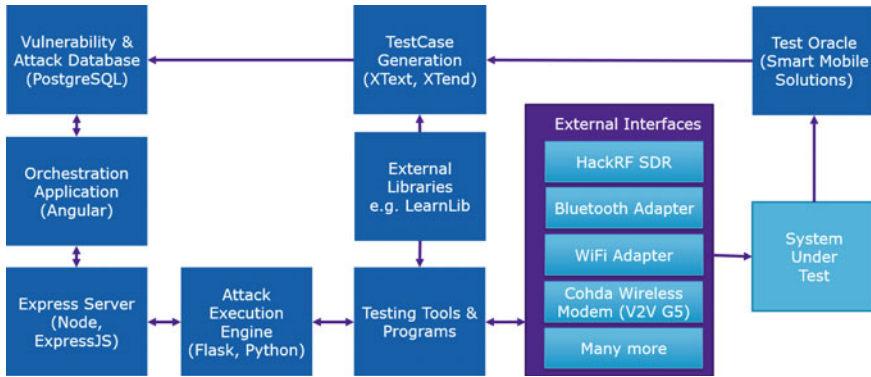


Fig. 5 AACT test execution framework [26]

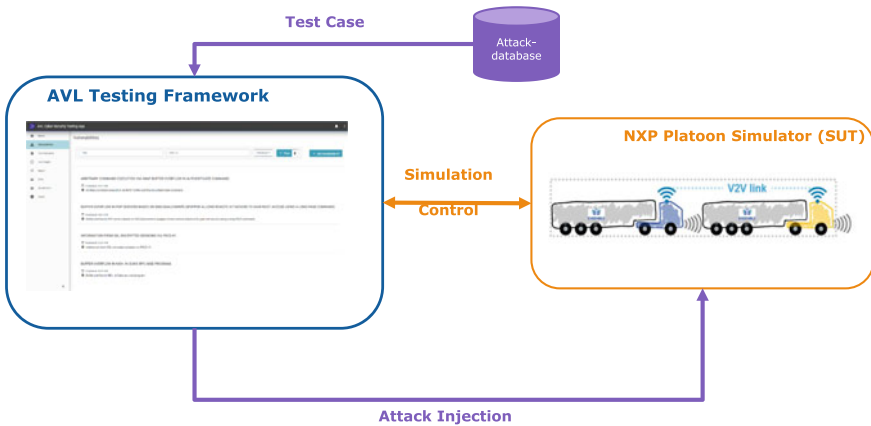


Fig. 6 Platooning use case overview

to 8 bits) the protocol was secure against the tested attack. Furthermore, ITS-G5 built-in signatures, that were disabled for the test, would have prevented a successful injection. The test could therefore show the security of the protocol in an automated way as described above.

4.5 Fuzzing

The goal of fuzzing is to reach a non-intended state of a SUT by using completely or partially random input. The latter technique may use a structured frame structure that is compliant with communication standards used by the SUT and randomized payload data [27]. In case of a CAN-Bus, a fuzzing tool can create packets that

consist of the standard ID Range (0 to 2047) and a previously learned or sniffed payload [20]. A fuzzer should include the following components [21]:

- A fuzz generator that assembles input from non-random components and random components with a sufficient amount of randomness
- A deliver mechanism that sends the generated inputs to the SUT
- A monitoring system (test oracle), which interprets the results such as SUT responses, monitored network communication, debug interface output, system signals or other physical responses and performs decisions based on it e.g. if a test passes or fails.

By using this approach, no in-depth knowledge about the SUT is needed and every component that provides external interfaces can be targeted for testing, including ECU software, ECU hardware, protocols and busses (e.g. CAN). Fuzzing may be utilized in the automotive environment to [10]:

- Reverse engineer messages on busses
- Disrupt an in-vehicle communication network
- perform a cyber-attack
- lead to vehicle component damage.

Depending on the used interface and protocol it may not be possible to fuzz-test every possible combination of input in its entirety in a feasible time frame. Therefore, it makes sense to pre-select meaningful value and position ranges for randomized content. Because of this potentially large test case space, fuzzing may be applied in parallel to other test methods as long as the complete run-time is still in a defined range and produces positive results.

In case of the AVL AXE, fuzzing CAN bus signals is a very common use case. A fuzzing software e.g. booFuzz, American Fuzzy Lop or caring caribou is armed with valid CAN Messages or a template with a specification which parts of messages should be randomized and then handles the tasks of subsequently sending the (generated) data to the SUT as well as receiving and interpreting the feedback (such as Vector Tools CANoe).

5 Conclusion

This chapter showed a holistic approach of cybersecurity testing of modern vehicles over the complete life cycle. It showed how, proceeding from threat modeling and variant management, test cases can (semi-)automatically be derived using structured processes and learning techniques. The generated tests are subsequently executed on an automated platform that is capable of controlling the test and/or simulation setup and applying the respective attack vector. The described methodology provides an end-to-end means to test vehicular systems over the complete life cycle.

References

1. Aarts, F., Heidarian, F., Kuppens, H., Olsen, P., Vaandrager, F.W.: Automata learning through counterexample guided abstraction refinement. In: FM 2012, pp. 10–27. Springer, Berlin (2012)
2. Aichernig, B.K., Muškardin, E., Pferscher, A.: Learning-based fuzzing of IoT message brokers. In: 2021 14th IEEE Conference on Software Testing, Verification and Validation (ICST), pp. 47–58 (2021). <https://doi.org/10.1109/ICST49551.2021.00017>
3. Angluin, D.: Learning regular sets from queries and counterexamples. *Inf. Comput.* **75**(2), 87–106 (1987). [https://doi.org/10.1016/0890-5401\(87\)90052-6](https://doi.org/10.1016/0890-5401(87)90052-6)
4. Ball, T., Cook, B., Levin, V., Rajamani, S.K.: Slam and static driver verifier: technology transfer of formal methods inside microsoft. In: International Conference on Integrated Formal Methods, pp. 1–20. Springer, Berlin (2004)
5. Dobes, T., Kaserer, T., Schuch, N., Storfer, G.: Smart variant calibration with data analytics. *ATZ - Automobiltechnische Zeitschrift (Extra August 2018)* (2018)
6. Ebrahimi, M., Marksteiner, S., Ničković, D., Bloem, R., Schögl, D., Eisner, P., Sprung, S., Schober, T., Chlup, S., Schmittner, C., König, S.: A systematic approach to automotive security. In: Chechik, M., Katoen, J.P., Leucker, M. (eds.) *Formal Methods*, pp. 598–609. Springer International Publishing, Cham (2023)
7. El Sadany, M., Schmittner, C., Kastner, W.: Assuring compliance with protection profiles with ThreatGet. In: Romanovsky, A., Troubitsyna, E., Gashi, I., Schoitsch, E., Bitsch, F. (eds.) *Computer Safety, Reliability, and Security*, pp. 62–73. Springer International Publishing, Cham (2019)
8. European Automobile Manufacturers' Association (ACEA): Vehicles in use europe 2022. Tech. rep., European Automobile Manufacturers' Association (ACEA) (2021)
9. European Telecommunications Standards Institute: Intelligent transport systems (its); vehicular communications; basic set of applications; definitions. ETSI “TS 102 638”, European Telecommunications Standards Institute (2009)
10. Fowler, D.S., Bryans, J., Shaikh, S.A., Wooderson, P.: Fuzz testing for automotive cybersecurity. In: 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), pp. 239–246 (2018). <https://doi.org/10.1109/DSN-W.2018.00070>
11. Franco da Silva, A.C., Wagner, S., Lazebnik, E., Traitel, E.: Using a cyber digital twin for continuous automotive security requirements verification. *IEEE Softw.* (2022). <https://doi.org/10.1109/MS.2022.3171305>
12. Garcia, F.D., de Koning Gans, G., Verdult, R.: Tutorial: Proxmark, the Swiss Army Knife for RFID Security Research: Tutorial at 8th Workshop on RFID Security and Privacy (RFIDsec 2012) (2012)
13. International Organization for Standardization: cards and security devices for personal identification—Contactless proximity objects—Part 3: initialization and anticollision. ISO/IEC Standard “14443-3”, International Organization for Standardization (2018)
14. International Organization for Standardization: tractors and machinery for agriculture and forestry—Safety-related parts of control systems. In: ISOStandard 25119, International Organization for Standardization (2018)
15. International Organization for Standardization, Society of Automotive Engineers: Road vehicles—Functional safety. ISOStandard 26262, International Organization for Standardization (2018)
16. International Organization for Standardization, Society of Automotive Engineers: Road Vehicles—Cybersecurity Engineering. ISO/SAE Standard “21434”, International Organization for Standardization (2022)
17. Isberner, M., Howar, F., Steffen, B.: The TTT Algorithm: a redundancy-free approach to active automata learning. In: Bonakdarpour, B., Smolka, S.A. (eds.) *Runtime Verification*, pp. 307–322. Lecture Notes in Computer Science, Springer International Publishing, Cham (2014). [10.1007/978-3-319-11164-3_26](https://doi.org/10.1007/978-3-319-11164-3_26)

18. Isberner, M., Howar, F., Steffen, B.: The open-source LearnLib. In: Kroening, D., Păsăreanu, C.S. (eds.) *Computer Aided Verification*, pp. 487–495. *Lecture Notes in Computer Science*, Springer International Publishing, Cham (2015). [10.1007/978-3-319-21690-4_32](https://doi.org/10.1007/978-3-319-21690-4_32)
19. Ladino, A., Xiao, L., Adjenugwhure, K., Deschle, N., Klunder, G.: Cross-platform simulation architecture with application to truck platooning impact assessment. In: *ITS world congress* (2021)
20. Lapczynski, P., Heinemann, H., Schöneberger, T., Metzker, E.: Automatically generating fuzz tests from automotive communication databases. In: *5th ESCAR USA, Detroit, isits AG* (2017)
21. Lee, H., Choi, K., Chung, K., Kim, J., Yim, K.: Fuzzing can packets into automobiles. In: *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*, pp. 817–821 (2015). <https://doi.org/10.1109/AINA.2015.274>
22. Marksteiner, S., Bronfman, S., Wolf, M., Lazebnik, E.: Using cyber digital twins for automated automotive cybersecurity testing. In: *2021 IEEE European Symposium on Security and Privacy Workshops (EuroSPW)*, pp. 123–128 (2021). <https://doi.org/10.1109/EuroSPW54576.2021.00020>
23. Marksteiner, S., Ma, Z.: Approaching the automation of cyber security testing of connected vehicles. In: *Proceedings of the Central European Cybersecurity Conference 2019. CECC 2019*, ACM, New York, NY, USA (2019). <https://doi.org/10.1145/3360664.3360729>
24. Marksteiner, S., Marko, N., Smulders, A., Karagiannis, S., Stahl, F., Hamazaryan, H., Schlick, R., Kraxberger, S., Vasenev, A.: A process to facilitate automated automotive cybersecurity testing. In: *2021 IEEE 93rd Vehicular Technology Conference (VTC Spring)*. IEEE, New York, NY, USA (2021)
25. Marksteiner, S., Priller, P.: A model-driven methodology for automotive cybersecurity test case generation. In: *2021 IEEE European Symposium on Security and Privacy Workshops (EuroSPW)*, pp. 129–135 (2021). <https://doi.org/10.1109/EuroSPW54576.2021.00021>
26. Marksteiner, S., Sirjani, M., Sjödin, M.: Using automata learning for compliance evaluation of communication protocols on an NFC handshake example. In: J. Kofron, J., Margaria, T., Seceleanu, C. (eds.) *Engineering of Computer-Based Systems*. vol. 14390, pp. 170–190. *Lecture Notes in Computer Science*, Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-49252-5_13
27. McNally, R., Yiu, K.K.H., Grove, D.A., Gerhardy, D.: Fuzzing: the state of the art (2012)
28. Otten, S., Glock, T., Hohl, C.P., Sax, E.: Model-based Variant management in automotive systems engineering. In: *2019 International Symposium on Systems Engineering (ISSE)*, pp. 1–7 (2019)
29. Petri, C.A.: *Kommunikation mit automaten*. Ph.D. thesis, Technische Universität Darmstadt (1962)
30. Phillips, C., Swiler, L.P.: A graph-based system for network-vulnerability analysis. In: *Proceedings of the 1998 Workshop on New Security Paradigms*, pp. 71–79. ACM (1998)
31. Rathfelder, M., Hsu, H., Brandau, T., Storfer, G.: Calibration data management for porsche chassis systems. *ATZ Worldw.* **118**(6), 16–21 (2016). <https://doi.org/10.1007/s38311-016-0047-z>
32. Rivest, R.L., Schapire, R.E.: Inference of finite automata using homing sequences. In: *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, pp. 411–420. *STOC '89*, Association for Computing Machinery, New York, NY, USA (1989). <https://doi.org/10.1145/73007.73047>
33. Schneier, B.: *Attack trees*. *Dr. Dobbs's J.* **24**(12), 21–29 (1999)
34. Shostack, A.: *Threat Modeling: Designing for Security*. Wiley (2014)
35. Sirjani, M.: Rebeca: theory, applications, and tools. In: de Boer, F.S., Bonsangue, M.M., Graf, S., de Roeper, W.P. (eds.) *Formal Methods for Components and Objects*, 5th International Symposium, FMCO 2006, Amsterdam, The Netherlands, November 7–10, 2006, Revised Lectures. *Lecture Notes in Computer Science*, vol. 4709, pp. 102–126. Springer (2006). [10.1007/978-3-540-74792-5_5](https://doi.org/10.1007/978-3-540-74792-5_5)

36. United Nations Economic and Social Council—Economic Commission for Europe: uniform provisions concerning the approval of vehicles with regards to cyber security and cyber security management system. Regulation “155”, United Nations Economic and Social Council—Economic Commission for Europe, Brussels (2021)
37. Vaandrager, F.: Model learning. *Commun. ACM* **60**(2), 86–95 (2017). <https://doi.org/10.1145/2967606>
38. Varadharajan, V.: Petri net based modelling of information flow security requirements. In: [1990] Proceedings. In: The Computer Security Foundations Workshop III, pp. 51–61 (1990)
39. Ward, D., Ibarra, I., Ruddle, A.: Threat analysis and risk assessment in automotive cyber security. *SAE Int. J. Passeng. Cars-Electron. Electr. Syst.* **6**(2013-01-1415), 507–513 (2013)
40. Wolschke, C., Marksteiner, S., Braun, T., Wolf, M.: An agnostic domain specific language for implementing attacks in an automotive use case. In: The 16th International Conference on Availability, Reliability and Security, pp. 1–9. ARES 2021, Association for Computing Machinery, New York, NY, USA (2021). <https://doi.org/10.1145/3465481.3470070>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Solar-Based Energy Harvesting and Low-Power Wireless Networks



Leander B. Hörmann, Julian Karoliny, and Philipp Peterseil

Abstract In modern industrial applications, machines and facilities, more and more sensors are used to control and optimise the processes. To be flexible and reduce cost, wireless sensors can be used in a broad range of applications. To prevent regular battery replacement, there is the possibility to supply wireless sensors by energy harvesting. In this chapter, we investigate the possibility to use solar-based energy harvesting to supply wireless sensors. For this, we consider four wireless network protocols and evaluate the power consumption using a simple sensor use case with different communication parameters. We further measure how much power can actually be harvested in a typical office environment with natural light. To provide realistic results, different sensor locations as well as seasonal changes are considered. The evaluations are combined to understand what size of solar cell is needed to supply the considered wireless protocols with different configurations.

1 Introduction

Modern industrial processes and machines require more and more data to control the operation and optimise it regarding, e.g., quality, speed, or energy consumption. This data has to be captured by sensors applied at the machines and facilities, and thus, more and more sensors are integrated. These sensors can be connected using Industrial Internet of Things (IIoT). The advances in ultra-low power electronics

L. B. Hörmann and J. Karoliny contributed equally to this work.

L. B. Hörmann (✉)

Linz Center of Mechatronics GmbH, Altenberger Straße 69, 4040 Linz, Austria
e-mail: leander.hoermann@lcm.at

J. Karoliny

Silicon Austria Labs, Altenberger Straße 66c, 4040 Linz, Austria
e-mail: julian.karoliny@silicon-austria.com

P. Peterseil

Johannes Kepler University, Altenberger Straße 69, 4040 Linz, Austria
e-mail: philipp.peterseil@jku.at

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_14

235

and wireless communication are the key enabling technologies to integrate computational power and wireless communication functionality directly into the sensors. Such intelligent sensors can be applied not only in industrial use cases, but also in typical Internet of Things (IoT) use cases, e.g., smart buildings, home automation, precision agriculture, and health care applications [1, 2]. Using wireless communication technology, sensors communicate with each other and form so-called wireless sensor networks (WSNs). As part of such a network, they are typically referred to as sensor nodes measuring physical quantities and transmitting them wirelessly towards a gateway using a certain communication protocol [3, 4]. Frequently used protocols to connect sensor nodes are Bluetooth Low Energy (BLE) and OpenThread [5]. If special requirements are necessary, specialised communication protocols may be necessary. One example would be ultra-wideband (UWB)-based communication if wireless localisation is necessary in the applications. Another example would be highly synchronised measurements utilised by a large number of energy-constrained sensor nodes. Here, highly specialised protocols like the Energy and Power Efficient Synchronous Sensor Network (EPHESOS) communication protocol [4] need to be considered.

Most wireless sensor nodes are not powered via the grid. Thus, they can be powered in two different ways: by batteries and by energy harvesting [6]. By using batteries, the energy is supplied already with the sensor node itself, with an operational time that is limited but guaranteed. In a lot of applications, this is acceptable since it is a relatively cheap solution and the cost is transferred to the customer. In this chapter, we discuss the second possibility. Using Energy Harvesting Devices (EHDs), the electrical power to supply the sensor nodes is converted from the power available in the environment. This could be for example solar or artificial light or temperature gradients. The theoretical operational time is not limited by using energy harvesting. This is of great interest if regular maintenance (e.g., for battery replacement) is not possible. Examples are industrial processes or applications where an interruption is unfeasible and cable-based measurement is impractical, e.g., rotating parts or high-voltage applications. In the following, we will describe the most common energy harvesting technology to transform environmental energy into electrical energy in order to supply embedded devices.

1.1 Solar-Based Energy Harvesting

Solar-based energy harvesting is very often used to power embedded devices because of its simplicity. Using solar cells, radiation available in the environment (e.g., visible light) is converted directly into electrical energy. However, typically environmental energy sources do not provide a continuous power [7, 8]. To guarantee the continuous operation of the supplied device, some kind of energy storage must be used. For example, supercapacitors or rechargeable batteries can be applied. The needed capacity depends on the expected variability of the energy source and the average power consumption of the supplied device. If the application area of the supplied device is

exposed to natural light from the sun (also indirectly inside buildings), the available power is still discontinuous but fortunately periodically accessible. This periodicity makes it possible to estimate the needed capacity of the energy storage and the size of the solar cells. A perpetual operation is achievable [9]. However, also weather conditions and seasonal changes must be considered in the calculations. In [10], we have presented real-world measurement results of solar-based harvestable energy in an office building. In addition to the good availability of light in most use cases, also simple conversion is an advantage of solar-based energy harvesting. In the simplest form, only a well-selected solar cell is necessary to supply embedded devices. In contrast to other energy harvesting methods (like electromechanical generators), there are no moving or rotating parts which can cause a malfunction. Furthermore, there are also off-the-shelf components to optimise the harvestable power by decoupling the solar cell and the supplied device with integrated maximum power point trackers (MPPTs). These are the reasons why solar-based energy harvesting is a highly favourable solution to supply embedded devices, and thus wireless sensor nodes. This chapter extends the already presented measurement results by combining them with real-world measurement results of the power consumption needed for the discussed communication protocols, and estimating the necessary size of the supplying solar cells for different protocol parameters.

The remaining part of this chapter is organised as follows. Section 2 describes the analysed communication protocols which can be used to connect wireless sensor nodes. Section 3 presents measurement results of the power consumption needed for the discussed communication protocols at various settings. Section 4 summarises the measurement results of the available energy in real-world scenarios. Finally, Sect. 5 presents the key results and Sect. 6 concludes the chapter and gives the direction for future work.

2 Low-Power Network Protocols

In this section, we will introduce different wireless protocols that are relevant for industrial applications and provide energy efficiency to combine with solar-based energy harvesting. We consider BLE, Thread, EPhESOS, and UWB in this work. BLE is one of the key enablers for IoT solutions and, with continuously added features like LE Audio, the number of BLE devices is drastically increasing. Thread or OpenThread is a representative of the home automation wireless protocols. With the huge support of major tech companies and the Matter project [11], Thread can be expected to be the main wireless protocol in this field. EPhESOS shall represent one proprietary wireless solution specifically tailored for industrial use cases. In detail, it supports a high degree of configurability, allows lots of sensor nodes, and determinism while maintaining low-power features. UWB gained a lot of momentum in the recent years, especially in indoor localisation topics. The high bandwidth of this technology allows high throughput and provides the time resolution for Time of Arrival (ToA) or Time Difference of Arrival (TDoA) measurements needed in local-

isation tasks. While there are many more protocols and standards worth considering, the presented ones will give a good overview of the different wireless sensor topics.

In the following, we summarise the key information of the proposed wireless protocols. We focus on the link layer parts of the protocols, i.e., how the access to the channel and thus the communication with other devices is managed. Although the physical (PHY) layer also plays a role in the comparison of wireless protocols regarding their power consumption, the major part that determines the low-power capability will be the link layer. To maintain the low power consumption needed to combine with energy harvesting, it is important to avoid transmitting messages without the target listening and going into receive mode without expecting a message at an exact time. As a result, we will focus on data exchange and channel access of the proposed protocols.

2.1 *Bluetooth Low Energy*

BLE is a Wireless Personal Area Network (WPAN) technology that operates in the 2.4 GHz industrial, scientific and medical (ISM) band. It is defined in the same standard as the classical version of Bluetooth BR/EDR, however, they are incompatible in terms of communication. Bluetooth Mesh uses the same PHY layer as BLE but builds different layers on top to support mesh functionalities. Due to the high energy efficiency of BLE, we will use it as a benchmark and representative of the Bluetooth communication protocols. BLE uses Gaussian Frequency Shift Keying (GFSK) modulation with a time-bandwidth-product of 0.5 as PHY layer. In this work, we will use the LE 1M PHY layer of BLE [12]. The specification defines 40 BLE channels with a bandwidth of 2 MHz in the 2.4 GHz ISM band, where channels 0 to 36 are used for general communication and channels 37–39 for advertisement. Before a BLE connection between a *central* and *peripheral* is formed, the peripheral uses these advertisement channels to announce its presence and supported features. The central requests for connection and, if the peripheral agrees, they switch to the general communication channel, exchange configuration parameters, and start communicating.

The communication is organised in so-called *connection events*, which happen periodically with a defined connection interval. The interval can be chosen between 7.5 ms and 4 s and is an important parameter defining the power consumption. Each connection event starts with a message from the central which is immediately answered by the peripheral. As long as there is data available or until the end of the connection event, they continue to exchange data. With the next connection event (the length of the connection interval later) the procedure starts again. To further save energy, the peripheral is also allowed to skip connection events, i.e., no reply to the central if no data is available. To reduce the chance of packet loss due to obstructed channels, BLE applies frequency hopping spread spectrum (FHSS). For each connection event, a new channel is calculated where both devices will communicate to exchange data. Besides reducing the probability of collisions, the channel-hopping itself has no effect on the consumed energy and is not further explained in this work.

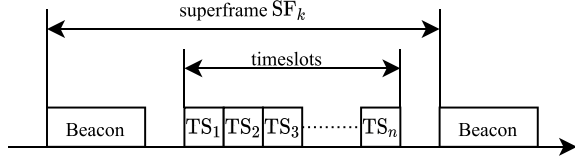
For the general channel access procedure we refer to [13]. Though both the central and peripheral are able to maintain a very low power consumption, we will use the peripheral for the energy harvesting considerations due to the additional low-power features. Parameters important for low-power operations are the connection interval and the number of connection events that the peripheral is allowed to skip.

2.2 IEEE 802.15.4 and Thread

IEEE 802.15.4 is a wireless standard that describes the operation of a low-rate wireless personal area network (LR-WPAN). It defines a PHY layer and media access control (MAC) layer on which many wireless protocols like ZigBee [14], WirelessHART [15], and Thread build on top. There are different PHY layer versions and frequency bands available, however, we focus on the 2.4 GHz ISM band PHY layer. Here the standard defines 16 channels with a bandwidth of 2 MHz and Offset Quadrature Phase Shift Keying (O-QPSK) for modulation. Additionally, direct-sequence spread spectrum (DSSS) is used which results in a transfer rate of 250 kbps. The MAC layer defines two operation modes, a communication purely based on Carrier Sense Multiple Access - Collision Avoidance (CSMA-CA) and a synchronised version with a central coordinator that sends out periodic beacons. We will focus on the unsynchronised version since it is the one used by the Thread specification. Thread is a low-power wireless mesh networking protocol which builds on IEEE 802.15.4 [5]. It supports IPv6 and is therefore perfectly tailored for IoT applications. Thread gained momentum in the home automation domain and major tech companies contribute to the specification. In a Thread-based network, there can be two types of nodes, *Full Thread Devices* and *Minimal Thread Devices*. Full Thread Devices are used to maintain the mesh features and the communication to other networks. These devices usually stay in receive mode and are not suitable for energy harvesting considerations. In this work, we focus on the Minimal Thread Devices, especially on the low-power end devices. For communication with the network, these devices always need a corresponding parent node for packet forwarding. The downlink communication to the node happens periodically to save energy on the low-power end device, while the uplink communication to the parent node can be performed at any time using CSMA-CA.

Unlike BLE, the uplink messages, e.g., sensor data, are not scheduled. However, since our applications are in the WSN domain, data will be provided periodically to the network similar to the other discussed communication protocols. For this, we are using User Datagram Protocol (UDP) messages which are directly supported by Thread. We will not consider the network establishment but only the data communication of a low-power end device to the parent. Thus, important energy-related parameters are the update period of sensor data and the downlink interval of the parent node. For the evaluation, we will use the open-source implementation of the Thread protocol, denoted as OpenThread, which is directly supported by various wireless transceivers.

Fig. 1 Format of the EPhESOS superframe including the beacon for synchronisation and the timeslots of the individual nodes



2.3 EPhESOS Protocol

In industrial environments, WSNs have to fulfil stringent requirements such as tight synchronisation, low power consumption, and deterministic latency. One way to meet these conditions is to guarantee deterministic channel access with time division multiple access (TDMA)-based network protocols and a centralised network coordinator. In TDMA-based networks, individual transmission timeslots are assigned to each sensor node by the network coordinator to prevent collisions between them. One such protocol is the EPhESOS protocol introduced in [4, 16]. A major advantage of EPhESOS is the independence of the PHY layer, which allows the application in different use cases and with various hardware. In this chapter, we choose one implementation of the EPhESOS protocol using the BLE PHY layer, however also others like IEEE 802.15.4 are possible choices. Low-power sensor nodes can join a network, i.e., register with a centralised network coordinator, during the so-called *sporadic mode*, further described in [4]. For the evaluation, we focus on the energy consumption during the operational state, the *continuous mode*.

In the continuous mode, all communication is performed highly synchronised in a TDMA structure. For this, every node in the network is only allowed to transmit in a specific timeslot assigned by the network coordinator in the previous steps. All possible timeslots are collected in a so-called *superframe* which is periodically transmitted with a certain period. Each superframe starts with a beacon, sent out by the network coordinator for synchronisation and as acknowledgements to the individual sensor nodes. The superframe structure of the EPhESOS network is depicted in Fig. 1. Due to the TDMA structure, collisions within the network are prevented and sensor nodes know when to transmit and receive at any time, which allows to maintain very low power consumption. Additionally, nodes only listen to beacons for acknowledgements and synchronisation if they transmitted data in the previous superframe. Otherwise, they can remain in sleep mode during the whole superframe, which additionally saves energy [16]. For the energy harvesting purpose, we will only consider the continuous mode of EPhESOS and the sensor nodes. The network coordinator is not suitable for low-power applications since it needs to stay in receive mode.

2.4 UWB Localisation

In recent years, the UWB technologies gained increasing popularity, especially in location-based topics like indoor tracking and access control. Many smartphone and car manufacturers are currently adding this technology to their products. UWB is a general term for radio communications with a bandwidth of 500 MHz or more. The current focus is on Impulse Radio UWB (IR-UWB) solutions which use radio frequency pulses with a very short time duration and thus a high bandwidth. This enables high timing resolution with steep edges that allows a very accurate ToA detection of received signals. With this, centimetre-level localisation utilising different ranging techniques like Two Way Ranging (TWR) or TDoA is possible. However, there exist many different UWB standards for the PHY and MAC layer. We focus here mainly on IEEE 802.15.4 and IEEE 802.15.4z since these are widely used. Similar to other protocols, we consider the access and communication scheme the most critical part for low-power considerations. However, the channel access scheme in UWB is not defined by any standard which causes many different proprietary solutions. Many chip suppliers leave the implementation of the MAC layer to the host system controlling the chip [17]. Additionally, the access scheme and communication strongly depend on the localisation approach. For the localisation, we differentiate between two different kinds of devices in the network, such as *Anchors* and *Tags*. Anchors are in most scenarios devices with a known reference location, while Tags are the mobile devices that will be localised. In this work, we consider the power consumption for the Tag, since this will be the device potentially powered by utilising energy harvesting. Here the most energy-efficient approach is TDoA, where the Tag sends out periodic beacons for the localisation. This beacon will be received by multiple anchors at different times due to the different distances to the Tag. If the Anchors are synchronised, the location of the Tag can be derived based on this time difference. Compared to the other discussed protocols, the location information is not in the transmitted packet, while it is the packet itself. This aspect, together with the completely different hardware makes a fair comparison challenging, however, we include also UWB measurements in our evaluation.

3 Power Consumption in Different Scenarios

In this section, we evaluate if the available energy in the different scenarios is enough to supply the proposed protocols. Here we focus on the WSN use case, i.e., the device-under-test shall provide its sensor data periodically to the network. We evaluate the energy consumption of the protocols introduced in Sect. 2 for the parts that are suitable for low-power sensor applications. As an example, it would be challenging to supply router nodes in a Thread network by energy harvesting since it is required to continuously activate their receiver unit.

In our example use case, a sensor node has to periodically transmit 2-byte sensor data to the network. We want to evaluate how the energy consumption depends on the transmission period including the overhead of the network protocol. Additionally, we want to investigate how much the energy consumption decreases if larger packets are transmitted, although less frequently. Since the beacon itself is the sensor value in the UWB case, we cannot transmit more location information less frequently. However, we still increase the data similar to the other protocols and see the localisation as a side product of the communication. To focus the evaluation only on the communication protocol, we will only transmit dummy data. This ensures that the evaluations are independent of the measurement acquisition.

3.1 Measurement Setup and Hardware

In our measurement setup, we use the NRF52840 transceiver [18], specifically, the corresponding development kit of Nordic Semiconductors. Since the NRF52840 supports multiple wireless protocols, we can cover all proposed network protocols but the UWB scenario. For this, we use the same hardware as described in [10] which combines the NRF52832 with the Qorvo DW1000 UWB transceiver unit. We assume that the power consumption in low-power sleep mode of both boards are comparable and differences in the measurements are due to the additional UWB transceiver unit and different network protocols. To make the individual measurements comparable, no acknowledgements or retransmissions are considered in the evaluations. For the energy consumption measurements, we use the Power Profiler Kit II [19] developed by Nordic Semiconductor. It is a low-cost solution for power profiling, which achieves a considerable measurement resolution of down to $0.2\ \mu\text{A}$ and a sampling speed of 100 kS/s. The Power Profiler Kit II is used in source mode to simultaneously power the device and measure the power consumption, which is averaged over 60 s. The power consumption measurements acquired within this work are published as open dataset under [20].

As BLE and OpenThread network stack, we use the implementation of the NRF Connect SDK [21] and adapted the provided example programs. For the BLE measurements, we evaluate the power consumption of a peripheral device. Here we choose a connection interval of 45 ms and allowed the device to skip up to 40 connection events if no data is available. For OpenThread, we implemented a *Sleepy End Device* which can transmit data anytime and periodically polls its parent node with a configured interval of 1 s. EPhESOS is implemented as described in [4] and for UWB we simply transmit periodic beacons to the anchor nodes for localisation. Here we performed our measurements with a centre frequency of 4.5 GHz and a bandwidth of 499.2 MHz. We choose a PRF of 64 MHz and 64 preamble symbols.

Fig. 2 Measured power consumption of the individual protocols with increasing update period

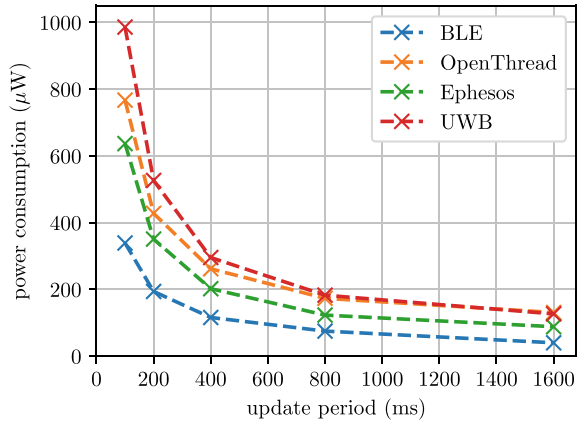


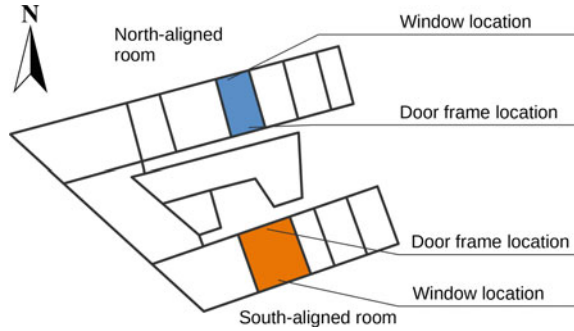
Table 1 Measured power consumption of the individual wireless network protocols with increasing transmit period

Period (ms)	Average power consumption (µW)			
	BLE	Thread	Ephesos	UWB
100	339.14	767.25	637.10	986.20
200	193.91	427.98	350.86	526.58
400	115.93	261.99	202.22	295.61
800	75.14	173.35	123.06	182.75
1600	40.13	130.84	88.28	126.65

3.2 Power Consumption with Increasing Update Period

One of the key factors for power consumption is the communication period of the network protocol. This period refers to the time between the transmission of two messages. If the sensor data is transmitted less frequently, the device can stay longer in low-power sleep mode and thus save energy. The variation in the individual network protocols is due to the different overhead to maintain the connection. We define a data acquisition interval of 100 ms, where we acquire 2 bytes of sensor data that needs to be transmitted. This interval is kept constant while increasing the communication period. As an example, for a communication period of 400 ms we will transmit 8 bytes of data. Figure 2 and Table 1 depict the results of the power consumption measurements for different communication periods, where they clearly show how power consumption decreases if the devices transmit the data less frequently. However, this decrease is not linear since the amount of data is increased for lower communication rates. Additionally, the overhead of the network protocol does not always scale with the update period. For example, the poll period in OpenThread for our configuration is constant and thus its part in the overall power consumption stays the same. Finally, even in low-power sleep mode, the power consumption is not zero.

Fig. 3 Floor plan of the relevant part of the building showing the orientation of the rooms and indicating the locations of the measurements



Also between the network protocols, a difference in the power consumption can be observed due to the different PHY layer and communication procedure. BLE is a highly-optimised low-power protocol and returns the lowest power consumption in our measurements. However, BLE only supports communication between two devices in the standard configuration. OpenThread and EPhESOS support mesh features with multiple nodes, though having a higher power consumption compared to BLE for our configuration. EPhESOS has a slightly lower power consumption and allows highly synchronised communication with multiple nodes, while OpenThread supports easy network authentication and higher network layer features like IPv6. UWB shows the highest power consumption, although we do not consider bidirectional communication. However, the results for the given use case are still comparable and UWB additionally provides high-accuracy localisation capabilities.

4 Available Energy in Real-World Scenarios

As already stated in Sect. 1, we have performed a measurement campaign inside an office building with solar cells. This section summarises the results, which are necessary to estimate the supply possibilities of the discussed communication protocols. The measurement campaign has been performed in Linz, Austria, at the coordinates N 48.335902 and E 14.322516 in July 2021 (summer). Figure 3 shows the four different locations for short-term measurements.

The measurements have been performed with a self-developed measurement device as presented in [22]. The connected solar cell is regularly characterised by measuring the voltage-current trace of the solar cell from open-circuit to short-circuit. Based on the recorded voltage-current trace, the maximum power point can be calculated. In this chapter, we are considering the three best performing solar cells, which are the YH-57 × 65 from Conrad components, with an active area of 30.87 cm² [23], SM141K10LV from IXYS, with an active area of 16.2 cm² [24], and SM141K09L from IXYS, with an active area of 14.49 cm² [25]. The maximum power point of the three solar cells has been averaged for the four measurement locations.

Fig. 4 Harvestable power density of the solar cells placed directly on the window at two rooms

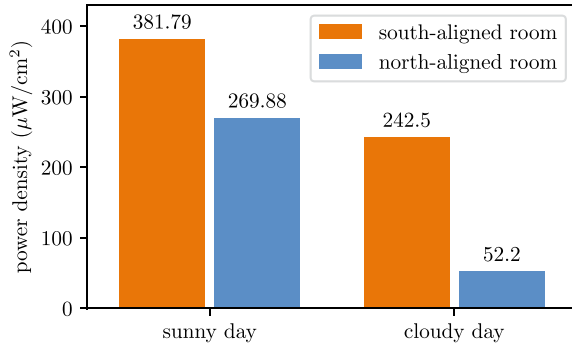


Fig. 5 Harvestable power density of the solar cells placed on the doorframe at two rooms

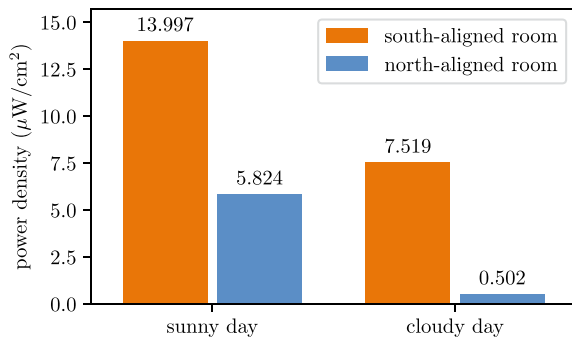


Figure 4 shows the harvestable power density of the solar cells mounted directly on the window glass. It compares the power in two different rooms in sunny and cloudy weather conditions. The results clearly show a difference in the harvestable power depending on the alignment of the room. Additionally, for changing weather conditions, e.g., cloudy weather, the degradation is more in the north-aligned room.

Figure 5 shows the harvestable power density of the solar cells mounted on the doorframe opposite the window side, though directed towards the window. The measurement compares the power in two different rooms in sunny and cloudy weather conditions. Although a similar ratio between the different conditions can be observed, the absolute values are considerably lower. While for the measurements on the window, we could achieve for the best condition a harvestable power density of $381.79 \mu\text{W}/\text{cm}^2$, on the doorframe only $13.997 \mu\text{W}/\text{cm}^2$ could be achieved. This additionally shows the huge impact of the location within the considered rooms.

Table 2 summarises the measurement results, including a factor for each value comparing it to the maximum value measured on the south-aligned window. The maximum power density $S_{EH,max,loc}$ refers to the measured daily maximum harvestable power density and the average power density $S_{EH,avg,loc}$ refers to the average value which considers also periods with no illumination during the night. This average value is estimated with $1/3$ of the daily maximum harvestable power.

Besides the location and the weather conditions, also seasonal changes influence the harvestable power density. Depending on the geo-location, the variation between

Table 2 Summary of harvestable power density at four locations and two different weather conditions. The average power density refers to the daily average value considering also night. A factor is given referring to the maximum harvestable power density (on window at south-aligned room)

Location	Maximum power density $S_{EH,max,loc}$ ($\mu\text{W}/\text{cm}^2$)	Avg. power density $S_{EH,avg,loc}$ ($\mu\text{W}/\text{cm}^2$)	Factor (%)
Room (south), on window, sunny	381.79	127.26	100.00
Room (south), on window, cloudy	242.50	80.83	63.52
Room (south), on doorframe, sunny	14.00	4.67	3.67
Room (south), on doorframe, cloudy	7.52	2.51	1.97
Room (north), on window, sunny	269.88	89.96	70.69
Room (north), on window, cloudy	52.20	17.40	13.67
Room (north), on doorframe, sunny	5.82	1.94	1.52
Room (north), on doorframe, cloudy	0.50	0.17	0.13

different seasons is significant. In a long-term measurement campaign done in Graz, Austria, we have tracked the daily harvestable energy for almost two years [6]. The relevant results are summarised in Table 3 showing the harvestable energy on the south and north side of a building. For summer, the daily maximum harvestable energy, and for winter, the average harvestable energy (including the months December, January and February) is depicted. The measurements during summer are comparable with those previously presented based on sunny weather conditions. To consider the lower harvestable energy during winter, we calculate a factor between the daily maximum during summer and the daily average during winter. It can be seen that the difference between summer and winter is more significant at the south side. The reason is the direct sunlight during summer which is much stronger on the south side. For bad weather conditions in general, the clouds and fog act as giant diffusor which distribute the light almost equally.

In order to estimate the harvestable energy, both location and seasonal changes have to be considered. The seasonal changes presented in Table 3 also include weather variations, since maximum and minimum values of nearly two years of records are used. As the factor depends on a specific location, these numbers should only indicate the magnitude and only be used for a rough estimation. The harvestable power density S_{EH} can be estimated as follows:

$$S_{EH} = S_{EH,avg,loc} \cdot f_{loc,weather,season} \quad (1)$$

Table 3 Summary of daily harvestable energy density at two locations in winter and summer

Season	Energy density of one day (J/cm ²)	Factor (%)
South outside, daily maximum summer	152.56	100.00
South outside, daily average winter	19.75	12.95
North outside, daily maximum summer	59.58	100.00
North outside, daily average winter	9.40	15.78

Table 4 Estimation of the daily average harvestable power density at four different locations at the south and the north side of a building

Location	$S_{EH, avg, loc}$ ($\mu W/cm^2$)	$f_{loc, weather, season}$ (%)	S_{EH} ($\mu W/cm^2$)
Room (south), on window	127.26	12.95	16.480
Room (south), on doorframe	4.67	12.95	0.605
Room (north), on window	89.96	15.78	14.196
Room (north), on doorframe	1.94	15.78	0.306

Table 4 reports the numbers used for the estimation of the daily average harvestable power density at four different locations. This estimation considers the worst-case scenario of supplying a device during the winter months including December, January, and February without artificial light. Since the factor is based on the average value of three months, it includes also seasonal and weather effects. Please note that the device needs an energy storage device to bridge periods with insufficient illumination conditions.

5 Experimental Results

In Sect. 4 we evaluated how much energy can be harvested in a typical office environment using the available solar energy, while in Sect. 3 we evaluated how much energy is needed in common network protocols. Combining the results from both evaluations, we now investigate if the proposed wireless protocols are suitable to be operated by solar-based energy harvesting solutions. Specifically, we evaluate how big the solar cell needs to be at least to support the considered network protocols for the different update periods. This will help with the design and location choice of the sensor nodes.

Figure 6 depicts the needed solar cell area for each communication protocol for the deployment on the window in the south-aligned room. This evaluation also includes the factor for seasonal changes as depicted in Table 4. Depending on the update period and choice of network protocol the needed area for the solar cell can be estimated.

Fig. 6 Needed solar cell area in cm² of the different network protocols and update periods for the south-aligned room and window position. Here also the weather and seasonal factors are included

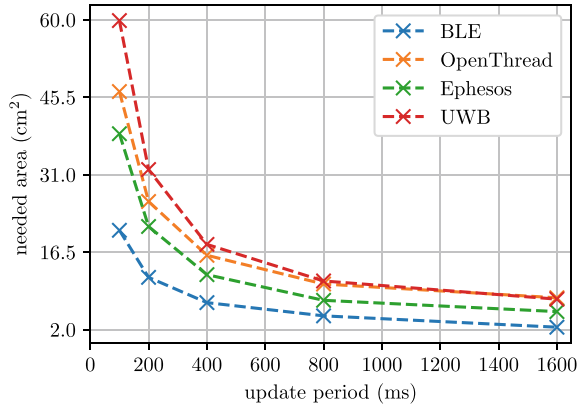


Table 5 Needed solar cell area of the different network protocols and update periods for the window position. Here also the weather and seasonal factors are included

Period (ms)	Needed area south (cm ²)				Needed area north (cm ²)			
	BLE	Thread	Ephesos	UWB	BLE	Thread	Ephesos	UWB
100	20.58	46.56	38.66	59.84	23.89	54.05	44.88	69.47
200	11.77	25.97	21.29	31.95	13.66	30.15	24.72	37.09
400	7.03	15.90	12.27	17.94	8.17	18.45	14.25	20.82
800	4.56	10.52	7.47	11.09	5.29	12.21	8.67	12.87
1600	2.43	7.94	5.36	7.69	2.83	9.22	6.22	8.92

Tables 5 and 6 illustrate the remaining results for the window and doorframe location, respectively, with Fig. 6 showing the experimental results listed in Table 5, with regard to the needed area in the south collection area (highlighted in gray color in Table 5). These presented results can be used to properly select a wireless communication depending on the specific application requirements as well as to estimate the shortest possible communication period if a certain solar cell area is given. The results also include location-dependent influences which may be of great interest for certain applications. The main outcome is that a supply of wireless devices is easily possible at well-illuminated locations on or near a window independent of its orientation. At a location with bad illumination conditions inside a room, the supply of wireless devices is more challenging and the solar cells must be significantly larger considering only natural light. However, also here a supply is possible if the communication periods are long enough.

Table 6 Needed solar cell area of the different network protocols and update periods for the doorframe position. Here also the weather and seasonal factors are included

Period (ms)	Needed area south (cm ²)				Needed area north (cm ²)			
	BLE	Thread	Ephesos	UWB	BLE	Thread	Ephesos	UWB
100	560.56	1268.18	1053.05	1630.09	1108.30	2507.35	2082.02	3222.89
200	320.51	707.40	579.93	870.38	633.69	1398.62	1146.59	1720.85
400	191.62	433.04	334.25	488.62	378.85	856.17	660.86	966.06
800	124.20	286.53	203.40	302.07	245.56	566.50	402.15	597.24
1600	66.33	216.27	145.91	209.35	131.14	427.60	288.48	413.90

6 Conclusion

This chapter discusses four well-suited communication protocols for wireless devices and presents measurement results of their power consumption at different communication periods. It further summarises the most relevant measurement results of the harvestable power at different locations in a building using solar cells including also the evaluation of weather and seasonal changes. Both measurement campaign results are combined in order to estimate the needed size of a solar cell depending on the location, the used communication protocol, and the selected communication periods considering only natural light. As presented, the supply using solar-based energy harvesting is easily possible at locations with good illumination conditions on or near a window. At locations with bad illumination conditions it is more challenging but also possible by using solar cells with reasonable size and adapted communication parameters.

References

1. Sisinni, E., Saifullah, A., Han, S., Jennehag, U., Gidlund, M.: Industrial internet of things: challenges, opportunities, and directions. *IEEE Trans. Ind. Inform.* **14**(11), 4724–4734 (2018)
2. Stoyanova, M., Nikoloudakis, Y., Panagiotakis, S., Pallis, E., Markakis, E.K.: A survey on the internet of things (IoT) forensics: challenges, approaches, and open issues. *IEEE Commun. Surv. Tutor.* **22**(2), 1191–1221 (2020)
3. Bal, M.: Industrial applications of collaborative wireless sensor networks: a survey. In: 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE), pp. 1463–1468 (2014)
4. Bernhard, H.P., Springer, A., Berger, A., Priller, P.: Life cycle of wireless sensor nodes in industrial environments. In: IEEE International Workshop on Factory Communication Systems—Proceedings, WFCS, vol. 7 (2017)
5. Kim, H.S., Kumar, S., Culler, D.E.: Thread/openthread: a compromise in low-power wireless multihop network architecture for the internet of things. *IEEE Commun. Mag.* **57**, 55–61, 7 (2019)
6. Hörmann, L.B., Buchegger, T., Steger, C.: Optimizing the energy supply of autonomous wireless sensor nodes. In: Microelectronic Systems Symposium (MESS), pp. 1–6 (2014)

7. Janek, A., Trummer, C., Steger, C., Weiss, R., Preishuber-Pfluegl, J., Pistauer, M.: Simulation based verification of energy storage architectures for higher class tags supported by energy harvesting devices. *Microprocess. Microsyst* **32**(5), 330–339 (2008); dependability and Testing of Modern Digital Systems. <https://www.sciencedirect.com/science/article/pii/S0141933108000379>
8. Hörmann, L.B., Berger, A., Pötsch, A., Priller, P., Springer, A.: Estimation of the harvestable power on wireless sensor nodes. In: *IEEE International Workshop on Measurements & Networking (M&N)*, pp. 1–6 (2015)
9. Kansal, A., Potter, D., Srivastava, M.B.: Performance aware tasking for environmentally powered sensor networks. *SIGMETRICS Perform. Eval. Rev.* **32**(1), 223–234 (2004). <https://doi.org/10.1145/1012888.1005714>
10. Hörmann, L.B., Hölzl, T., Kastl, C., Priller, P., Bernhard, H.-P., Peterseil, P., Springer, A.: Evaluation of solar-based energy harvesting for indoor IoT applications. In: *European Test and Telemetry Conference (ettc2022)*, pp. 190–198 (2022)
11. Connectivity Standards Alliance: Matter Specification Version 1.0 (2022)
12. Bluetooth SIG: Bluetooth Core Specification, v 5.2 (2019)
13. Karoliny, J., Blazek, T., Springer, A., Bernhard, H.-P.: Predicting the channel access of bluetooth low energy. In: *2023 IEEE International Conference on Communications (ICC): IoT and Sensor Networks Symposium (IEEE ICC'23 - IoTSN Symposium)*, Rome, Italy (2023) [PREPRINT]. <https://doi.org/10.48550/arXiv.2301.08109>
14. Alliance, Z.B.: ZigBee specification R22 1.0 (2019)
15. Kim, A.N., Hekland, F., Petersen, S., Doyle, P.: When HART goes wireless: understanding and implementing the WirelessHART standard. In: *IEEE International Conference on Emerging Technologies and Factory Automation*, pp. 899–907 (2008)
16. Berger, A., Holzl, T., Hormann, L.B., Bernhard, H.P., Springer, A., Priller, P.: An environmentally powered wireless sensor node for high precision temperature measurements. In: *SAS 2017—2017 IEEE Sensors Applications Symposium*, Proceedings, vol. 4 (2017)
17. Coppens, D., Shahid, A., Lemey, S., Herbruggen, B.V., Marshall, C., Poorter, E.D.: An overview of UWB standards and organizations (IEEE 802.15.4, FiRa, Apple): interoperability aspects and future research directions. *IEEE Access* **10**, 70 219–70 241 (2022)
18. Nordic Semiconductor: NRF52840 Product Specification v1.1 (2019)
19. Nordic Semiconductor: Power Profiler Kit II—User Guide v1.0.1 (2020). https://infocenter.nordicsemi.com/pdf/PPK2_User_Guide_v1.0.1.pdf
20. Karoliny, J., Hörmann, L.B., Peterseil, P.: InSecTT WSN power consumption dataset (2023). <https://doi.org/10.5281/zenodo.7762712>
21. Nordic Semiconductor: nRF Connect SDK v2.0.2 (2023). <https://github.com/nrfconnect/sdk-nrf>
22. Hörmann, L.B., Hölzl, T., Kastl, C., Priller, P., Springer, A.: Evaluation of energy harvesting devices for industrial applications. In: *European Test and Telemetry Conference (ettc2020)*, pp. 31–37 (2020)
23. Conrad Components SE: Product specification - solar cell module type YH-57X65. <https://asset.conrad.com/media10/add/160267/c1/-/en/000-191321DS01/datasheet-191321-conrad-components-yh-57x65-solar-panel.pdf>
24. IXYS KOREA LTD: SM141K10LV - IXOLAR™ High Efficiency SolarMD. <https://ixapps.ixys.com/DataSheet/SM141K10LV.pdf>
25. IXYS KOREA LTD: SM141K09L—IXOLAR™ high efficiency SolarMD. <https://ixapps.ixys.com/DataSheet/SM141K09L.pdf>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Location Awareness in HealthCare



Frank van de Laar and Karin Klabunde

Abstract Waiting for patients and looking for or losing assets in healthcare costs millions/year and may cause considerable annoyance for caregivers and additional health risks for patients. Localization solutions for healthcare may help to reduce these significantly. This paper provides an overview on localization for Healthcare. It starts with an overview of applicable technologies for both indoor and outdoor localization that may be applied in healthcare logistics solutions. The next section describes various aspects that needs to be considered for an end-to-end localization solution. It is followed by a section on use-cases for healthcare with different operational requirements that may deploy technologies and methods as presented. The final section is on building a use-case concept demonstrator taking into account the technologies and considerations addressed in the previous sections. Security and privacy aspects are also considered to comply with EU healthcare regulations, especially when dealing with personal location data.

Abbreviations

AoA	Angel of Arrival
AoD	Angel of Departure
AP	Access Point
API	Application Programming Interface
BLE	Bluetooth Low Energy
BSSID	Basic Service Set Identifier
CVSS	Common Vulnerability Scoring System
CNN	Convolutional Neural Network
CMX	Connected Mobile Experiences
CTE	Continuous Tone Extension
D2C	Device to Cloud

F. van de Laar (✉) · K. Klabunde
Philips Research, Eindhoven, Netherlands
e-mail: frank.ida@kpnmail.nl

FHIR	Fast Healthcare Interoperability Resources
GDPR	General Data Protection Regulation
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPX	GPS Exchange Format
HIPAA	Health Insurance Portability and Accountability Act
IAM	Identity and Access Management
IoT	Internet of Things
IPS	Indoor Positioning System
ISM	Industrial, Scientific, and Medical
JSON	JavaScript Object Notation
LAN	Local Area Network
LoS	Line of Sight
LTE	Long Term Evolution
LPWAN	Low Power Wide Area Network
MCI	Mass Casualty Incident
ML	Machine Learning
MNO	Mobile Network Operator
IR	Infra-Red
LOCI	Localization Infrared
NFC	Near Field Communication
OSM	Open Street Map
QR	Quick Response (code)
PoE	Power over Ethernet
PDR	Pedestrian Dead Reckoning
RF	Radio Frequency
RFC	Request For Comment
RSSI	Received Signal Strength Information
RTK	Real Time Kinematics
RTLS	Real Time Location System
RTT	Round Trip Time
TLS	Transport Layer Security
ToF	Time of Flight
TTFF	Time To First Fix
TWR	Two Way Ranging
UWB	Ultra-Wide Band
WFA	Wi-Fi Alliance
XML	Extensible Markup Language

1 Terminology and Technology

A state-of-the-art topology of a location awareness solution for healthcare is shown in Fig. 1. It may involve various RF technologies with different characteristics for indoor localization and GPS for outdoor localization of assets and people. These technologies as well as non-RF technologies will be addressed in the following subsections.

1.1 Positioning, Localization, Tracking and Navigation

Before addressing technologies it is important to distinguish some basic terminology as used regarding location awareness in this paper:

Positioning is about determining the absolute or relative position of an asset on a map or a floorplan in a coordinate system. Positioning can be done using 1, 2 or 3 dimensional coordinates or simply by indicating a zone number or area code. The latter is sometimes referred to as 0-dimensional positioning.

For a geographical map a position typically will be expressed in geo-coordinates (latitude, longitude, altitude). For a local map or floorplan coordinates (x,y,z) can be expressed as the distance (x,y) with a certain unity (for example meters or feet) to a reference (corner) position and optionally a floor level number (z).

Localization is about visualizing the physical location of an asset on a map or floorplan to a user.

Tracking is about tracing the positions or locations of an (moving) asset over time.

Navigation is about getting guidance on a map to find the shortest, easiest, or fastest route to a required asset or specified destination.

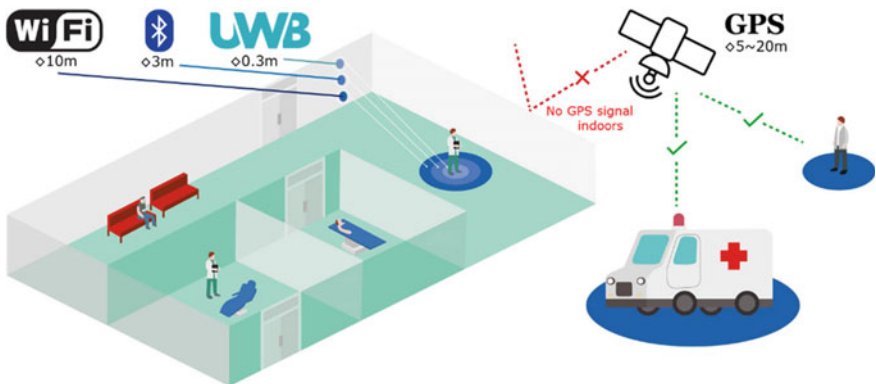


Fig. 1 Major RF technologies for indoor and outdoor localization in healthcare

1.2 RF-Based Indoor Localization Technologies

A typical RF-based Real Time Location System (RTLS) is made up of multiple receivers, often called “beacons” or “anchors” on the ceiling/walls and transmitters, called “tags”, on the assets to be tracked. The tags send signals to the anchors that are connected to a network. The received tag signals are sent to a back-end server that calculates the position of the tags in real-time using proximity (strongest signal), trilateration (estimating distances based on received signal strength) or triangulation (received signal direction based) as shown in Fig. 2. The server can provide the tag position(s) to an application in order to show the location(s) on a dashboard to the user. The tag itself does not know its position.

Indoor Navigation Solutions (also sometimes referred to as Indoor Positioning System (IPS)) work in the opposite way. A device (usually a smartphone) will receive signals from “beacons” on the ceiling/walls. The device will then calculate its position and present it to the user for localization or navigation purposes.

Location fingerprinting can be used to identify a position more reliably by matching a received signals profile to predetermined location reference profiles using Machine Learning [1]. Fingerprinting typically requires a time-consuming survey of the RF environment and is prone to errors due to changes of this environment.

Various RF-based wireless technology options are available for a positioning system such as Bluetooth, Wi-Fi, UWB. These are described in more detail below. Various non-RF based methods are described in the section here after.

Different state-of-the art technologies that can be used for indoor localization are described in the following subsections:

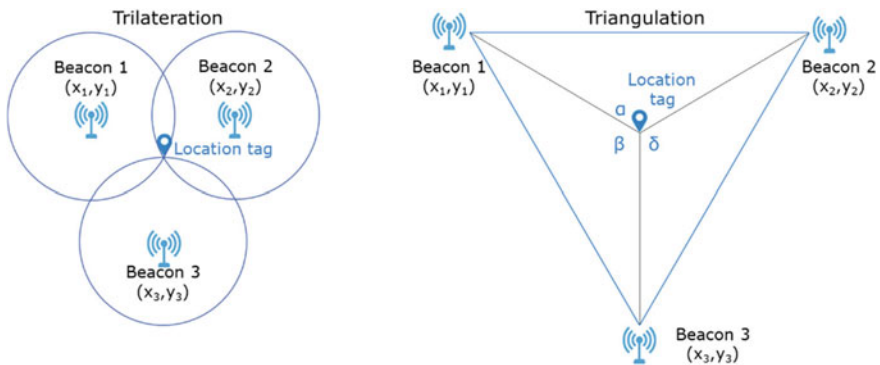


Fig. 2 Trilateration (RSSI based) versus Triangulation (direction based) position estimation

1.2.1 Bluetooth

Bluetooth Low Energy (BLE) can be used to do indoor localization using simple BLE beacons as tags and multiple receivers measuring RSSI to estimate the tag position. This provides a basic low cost, low energy solution with a low accuracy. It can be widely used in hospitals by reusing network infrastructure with BLE-enabled Wi-Fi Access Points (AP). If needed the accuracy can be improved using fingerprinting technologies.

Bluetooth 5.1 has RF provisions for Angle of Arrival (AoA) or Angle of Departure (AoD) detection that allows more accurate localization with proper beacon installations. The 5.1 specification adds a Continuous Tone Extension (CTE) to a Bluetooth packet to enable a receiver to calculate the position of its transceiver from the RF signal [2]. BLE tags can be both low-cost and low energy, but a network gateway function is required to make the data available on the internet. Note that Time-of-Flight (ToF) for accurate distance finding between beacons is not in the scope of Bluetooth 5.1, so the distance between beacons should be known in advance or determined less accurate using RSSI. Whether AoA or AoD should be used depends on the use-case:

- In an AoA direction finding asset tracking system (such as a RTLS) the tag uses a single antenna and conventional Bluetooth LE SoC to send Bluetooth 5.1 packets with CTE. The main computation occurs on the multi-antenna locator side of the system where the signal data gathered by the locator is fed to a location engine that runs the direction-finding algorithms.
- In an AoD direction finding indoor positioning system (IPS) the fixed beacons use antenna arrays to send Bluetooth 5.1 packets with CTE. The main computation occurs on a mobile device, such as a person’s smartphone (Fig. 3).

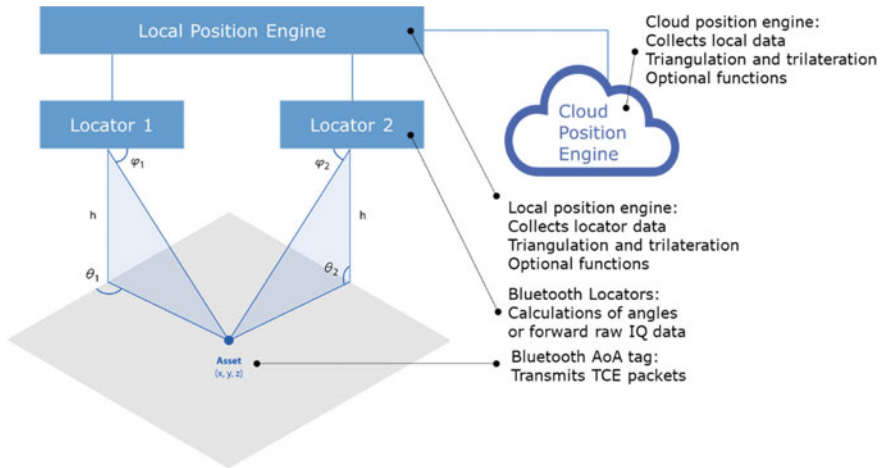


Fig. 3 Bluetooth 5.1 solution for asset tracking based on AoA

1.2.2 Wi-Fi

The big advantage of using Wi-Fi for indoor localization is that the wireless infrastructure is already there in enterprise environments like hospitals and offices. Unfortunately, it is not reliable as it typically depends on trilateration. This also means that the physical AP distribution needs to be considered accordingly. Using signal-strength based location fingerprinting may help to improve the accuracy at the cost of creating (offline) a floorplan based signature list and the risk of mismatches in a changing environment.

As example for a Wi-Fi-based locating solution, the Cisco Connected Mobile Experiences (CMX) [3] Detect and Locate service enables viewing and tracking of APs and clients (including Wi-Fi tags) in an enterprise Wi-Fi deployment (Fig. 4). Typical accuracy is about 10 m.

The CMX service has recently been integrated into Cisco Spaces that includes additional features for i.a. behavior metrics, location analytics and partner integration. Other Wi-Fi network suppliers (e.g., Aruba, Extreme Networks, Alcatel Lucent) provide similar solutions for device localization.

The Wi-Fi Alliance has recently developed Release 2 of Wi-Fi Location using RTT (Round-Trip-Time) to estimate distance between a device and an AP [4]. It is based on IEEE 802.11az (successor of 802.11mc) and offers the following features:

- Improved accuracy, about 1 m for enterprise positioning and less than 1 m for proximity detection.
- Improved operation in Non-Line-of-Sight (NLOS) conditions, by leveraging Wi-Fi 6 Multi-User MIMO technology.



Fig. 4 Screenshot from CMX system showing the location of a Wi-Fi tag (red dot) and nearby APs (blue dots) in a test setup.

- MAC/PHY Security, including prevention against eavesdropping and impersonation and protection against false sense of distance.

Reference chipsets are available from Broadcom and NXP. Google provides an API for Android Wi-Fi Location ranging with RTT.

Wi-Fi can also be used to roughly determine the locations of assets outside the hospital using the signals strength of received APs with known locations. The location accuracy depends on the size and actuality of the AP database. Google, Here and others (including open source) maintain such databases and offer APIs to get a geographical location based on a list of detected APs.

1.2.3 UWB

Ultra-Wideband (UWB) also provides a reliable and accurate way of asset tracking using tags attached to or integrated with assets [5], but it requires a dedicated infrastructure for the UWB receivers.

UWB solutions are based on Time of Flight (ToF) method for calculating the distance between two transceivers by multiplying the ToF of the RF signal by the speed of light. The signal pulses used to measure the ToF can be short (typically 2–3 ns) due to the ultra-wide RF bandwidth (>500 MHz). The short UWB pulses prevent interference from reflected signals, resulting in a much higher ranging accuracy than non-wideband solutions.

UWB ranging can be done using either Two-way ranging (TWR) or Time difference of arrival (TDoA) as shown in Fig. 5. The TWR method requires two-way communication between a tag and three or more anchors, avoiding the need for accurate synchronization between the anchors at the cost of a higher energy consumption for the tags. TDoA requires accurate synchronization between the anchors, but the energy consumption for the tags can be lower without the need to receive. Angle of arrival (AoA) detection may optionally be added to the anchors to improve accuracy or to reduce the number of required anchors.

For asset tracking BLE and Wi-Fi may be more cost effective, but with respect to either data privacy, communication security, and location accuracy UWB technology may be a better choice. BLE and Wi-Fi tracking accuracy in combination with its wide scale use on modern smartphones is also considered as a privacy concern w.r.t. potential stalking and surveillance. However, UWB based tracking of many assets in a crowded and noisy RF (hospital) environment may significantly degrade its reliability and accuracy due to a lack of LoS [6].

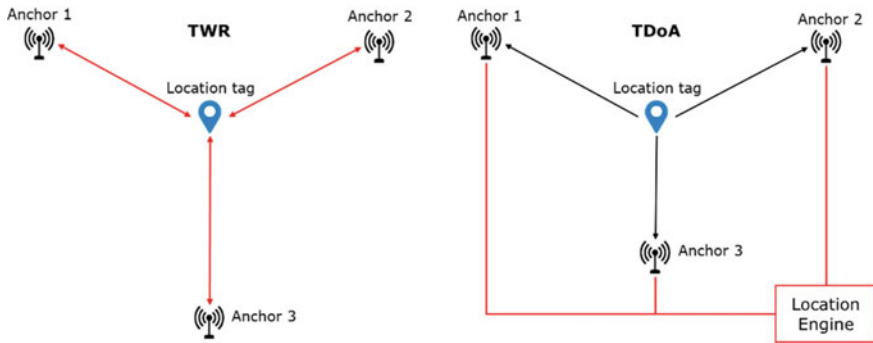


Fig. 5 UWB using either TWR or TDoA for distance measuring. TWR requires bidirectional communication between anchors and tags whereas TDoA requires anchor synchronization

1.3 Non-RF Based Localization Technologies

1.3.1 InfraRed (IR)

Infra-red systems provide a reliable way of asset tracking using tags attached to, or integrated with assets, but they require a dedicated infra-structure for the IR receivers/transmitters (both directions are possible: either tag receives IR signal and transmits to server or the other way round). Unlike RF technologies, IR signals do not pass-through walls and ceilings, which is an advantage for having room level and even bed-level accuracy and certainty. An example of a healthcare IR solution [7] is shown in Fig. 6.

The following operational steps are involved in using this solution:

- IR beacons are installed in all areas where assets need to be tracked. All beacons send out regularly (e.g. every 1.5–3 s) an IR code. Areas can be grouped to define a so-called zone.



Fig. 6 IR tag-based localization solution

- A communication infrastructure of hubs is set-up which build the communication network for all components of the IR solution. Hubs are the only cabled components of the solution (with Power over Ethernet, PoE).
- A person or asset is given a battery-powered IR tag.
- The tag is always active. It scans for IR signals e.g. every 1.5–3 s (configurable).
- The tag receives an IR zone-id from the beacons and emits a (900 MHz) RF signal with its tag-id and the zone-id to communications hubs.
- The Communication Hubs send the IR tag-id and IR zone-id to a central server using a LAN.
- A dashboard program on a computer connected to the server collects and displays the location data.

1.3.2 QR Codes

A very simple way of obtaining one’s location may be by using QR tags with the geolocation or area/zone code, where it is placed, stored on it. By scanning the QR tag with an appropriate application it is clear where the person or asset is [8]. An advantage of such a solution is that it may also work in situations where the network infrastructure is down or not available. A disadvantage is that it typically requires manual actions and scanning is not a low energy operation (Fig. 7).

2 Pedestrian Dead Reckoning (PDR)

PDR on a smartphone may be used for tracking or navigation of people in a hospital when starting from a known position. PDR systems typically rely on an accelerometer for measuring displacement, a magnetometer for orientation and a gyroscope for

Fig. 7 QR location tag example with geoJSON data



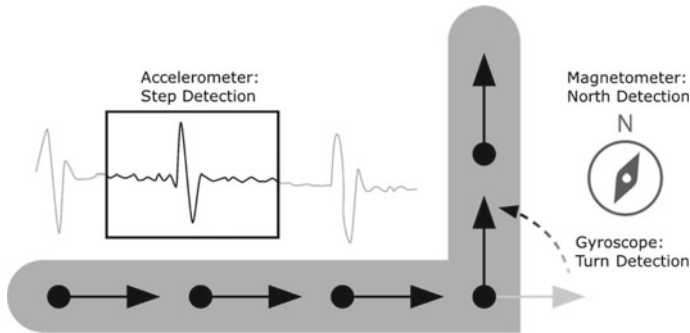


Fig. 8 Accelerometer based PDR relies on step detection

detecting turns (Fig. 8). Its accuracy will be low since an accelerometer (as the name implies) cannot measure speed and step counting is not a reliable method to determine displacement as step sizes are variable. It is also not applicable to assets on wheels tracking as no steps will be detected probably.

3 Others

In addition to the location solutions described above it is also possible to determine an asset position without having a tag associated with it. This can be done using cameras or Passive Infrared Sensors (PIR) [9] extended with object or people detection. Many of these solutions rely on an existing infrastructure for surveillance purposes. Extending those with localization features will allow for situational awareness in emergency situations but does not allow individual localization as no UIDs are available. For example, in case of a threat such a system will allow to provide a quick overview of how many people are involved, in which areas they are and in which direction they are moving.

3.1 Outdoor Localization Technologies

3.1.1 GNSS/GPS

Using GNSS (Global Navigation Satellite System) or more specifically GPS (Global Positioning System) is the obvious way for outdoor asset positioning [10]. Accuracy for GPS smartphone is up to about 10 m in the open field, but the accuracy may reduce up to 30 m near buildings, bridges, and trees. The accuracy may be improved using filtering and applying map information. PDR (Pedestrian Dead Reckoning) based

on accelerometer and gyroscope sensor data may be used to track slowly moving devices.

Getting an initial position may take a considerable time. Each GPS satellite sends a “legacy” (L1 C/A) navigation frame every 30 s. This message consists of two main components:

- Ephemeris data: used to calculate the position of each satellite in orbit and valid for 2–4 h.
- Almanac data: information about the time and status of the entire satellite constellation and it is valid for about 2 weeks.

Only a small portion of the Almanac is included in one GPS frame. It takes 25 frames (12.5 min) to get the full Almanac. The full Almanac (15,000 bits) is needed before a GPS fix can be obtained. The Time To First Fix (TTFF) is a measure of the time required for a GPS receiver to acquire satellite signals and navigation data and calculate its position (called a fix).

During a cold start, that is when a GPS module has been off for a few days and does not have actual data in its memory. The full Almanac is required for processing. If the GPS module has clear line of sight to all satellites, the shortest time for TTFF is 12.5 min.

In a warm start scenario, the GPS module has valid Almanac data, is close to its last position (100 km or so) and knows the time within about 20 s. This approximate information helps the receiver estimate the range to satellites. The TTFF for a warm start can be as short as 30 s (1 frame), but typically it is a couple of minutes.

A receiver that has a current almanac, ephemeris data, time and position can have a hot start. A hot start can take from 0.5 s to 20 s for TTFF. Smartphones use Assisted GPS (A-GPS), this allows them to download the ephemeris and almanac data (size 1 ~ 3 kB) over the cell network which greatly reduces the TTFF (Fig. 9).

Two new methods have emerged recently to reduce energy consumption: transmission of pseudo-ranges for remote position determination, and snapshot reception [11].

The snapshot method for IoT solutions [28] uses only a very short (≥ 4 ms) interval of the received GNSS signal that is processed in the cloud (including A-GPS support) to reduce energy consumption at the device side at the cost of a reduced accuracy (Fig. 10). Using a reception front-end with sampling frequency equal to 4 MHz,



Fig. 9 Assisted GNSS or GPS improves TTFF in adverse signal conditions

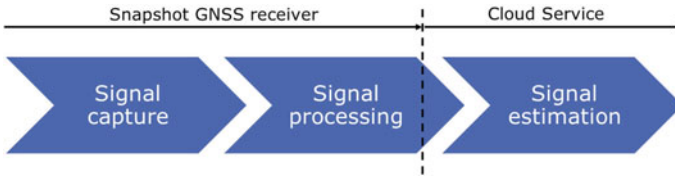


Fig. 10 A snapshot receiver uploads GNSS data to the cloud to obtain position, velocity, and time (PVT)

Table 1 Major characteristics of various GNSS implementation methods

Method	Capture	Accuracy	Processing	Download	Upload
GNSS/GPS	>30 s	<20 m	On-device	NA	Position
Assisted	<30 s	<20 m	On-device	Almanac/Ephemeris	Position
Snapshot	<0.1 s	<20 m	On-cloud	NA	Snapshot
DGPS	<30 s	<1 m	On-device	RTCM data	Position
SRTK	<0.1 s	<0.1 m	On-cloud	NA	Snapshot

a 1-bit ADC and a snapshot length of 4 ms, the size of an upload packet would be about $(4000 \text{ kHz} \times 4 \text{ ms}/8 =) 2 \text{ kB}$. The energy saved from normal GNSS processing (typically requiring 1–30 s of data) on the device will outweigh the energy consumption from uploading the snapshot data.

GNSS position accuracy can be improved by using Differential GPS (DGPS) or Real Time Kinematics (RTK), both using correction (and ephemeris) data from a static local GNSS reference station. The combination of RTK with snapshot methodology is called snapshot RTK (SRTK). For healthcare applications this higher level of accuracy will probably not be needed.

Table 1 shows the major characteristics of each method:

Refinements or combinations of the methods above allow for further optimizations on processing, communication, and accuracy tradeoffs.¹

3.2 Technology Overview

The table below provides an overview of the indoor positioning methods described above and including GNSS/GPS for outdoor. The tag-based technologies are most suitable for asset tracking, whereas the camera and PIR based solutions are more suitable for situational awareness. Combining technologies may be useful to improve the accuracy or range of the solution.²

¹ Rough estimates based on a battery capacity of about 3000 mWh, a message payload of 25 bytes, sending 1 message/hour and a 5uA sleep current.

² Assuming roughly 1 k devices, 12 location messages/day, www.emnify.com.

Table 2 Real Time Location System (RTLS) technology comparison

Technology	Tag based	Accuracy	Presence	Position	Energy	Range [m]	Limitations
BLE (RSSI)	✓	m	✓	Grid	Low	3–5	BLE 4.0 +
BLE (direction finding)	✓	m	✓	Grid	Low	0.1–2	BLE 5.1
Wi-Fi	✓	m	✓	Grid	Medium	1–50	Interference
UWB	✓	cm	✓	Grid	Low	1–50	Line of Sight
InfraRed	✓	m	✓	Zone(s)	Low	1–5	Line of Sight
Camera	×	cm	✓	Grid	High	1–10	Line of Sight
PDR	×	m	×	Relative	High	1–100	Cumulative Error
PIR	×	dm	✓	Grid	Low	1–10	Line of Sight
QR codes	✓	m	×	Zone(s)	Medium	NA	Smart Scanner required
GNSS/GPS	✓	m	×	Absolute	Medium	∞	Outdoor only

4 Designing an End-To-End IoT Solution

When designing an end-to-end solution based on localization or logistics tags using the technologies as described in the previous section, various aspects need to be considered. These include commissioning, reliable low power wireless area network (LPWAN) communication and battery lifetime of tags, handling transitions from indoor to outdoor, available APIs for providing location and map visualization services on a user dashboard and application specific security and privacy issues. These are described in the following subsections.

4.1 Commissioning

A localization tag needs to be associated with a unique person’s or device’s identity and be securely connected to an IoT network. The process to do so is referred to as commissioning. To do so each localization tag will have its own unique id (UID) associated with each message, which can also be stored on an NFC tag or shown on a sticker with a QR or bar code. A commissioning device (typically a smartphone) that is already securely connected to the network may perform the following steps:

Table 3 Most used LPWAN technologies

	LoRa	NB-IoT	LTE-M
Range	5 km	>10 km	>10 km
Maximum coupling loss	155 dB	≥ 164 Db	156 dB
Throughput (uplink)	0.3–50 kbps	4.8–62.5 kbps (NB1) ≤ 160 kbps (NB2)	≤ 375 kb/s (M1) ≤ 2600 kb/s (M2)
Battery lifetime	10 years	5 years	2 years
MNO support	No	Yes	Yes
Latency (uplink)	<10 s	<10 s	<1 s
Peak current @3 V	<50 mA (14dBm)	<200 mA	<300 mA
Security	AES-128	LTE (128 bits key block cipher)	LTE (128 bits key block cipher)
Commissioning/ pairing	Custom	(e)SIM based	(e)SIM based
Mobility (velocity)	<40 km/h	<100 km/h	<300 km/h
Positioning accuracy	<200 m	<100 m	<50 m
Network coverage	National	Mainly EU + Asia	Mainly US + EU
Network capacity	<1 k devices/cell	>50 k devices/cell	>50 k devices/cell
Firmware OTA	Difficult	Limited	Yes
Native IP support	No	Yes	Yes
Suitable for streaming data	No	No	Yes
Max. message size (bytes)	51 (SF > 9)	1600	NA
Subscription cost/ device	12€/year?	12€/year	12€/year

- Scan this UID (Fig. 11) and register the IoT device for use by the Application(s).
- Optionally configure the IoT device. For a localization tag this may for example be the reporting period or a geofence area.
- Provision the device for IoT network access (depending on its connectivity technologies). In case of Wi-Fi the “Easy Connect” technology may be used for onboarding IoT devices.
- Link the device to a person’s or asset’s identity that has been previously registered into the system. Obtaining the person’s or asset’s identity may be done by an additional UID scan operation or by selection from a UI list.

If needed multiple tags (supporting different technologies or functions) can be associated with one person or device by either commissioning both tags separately as described above or by assigning the same UID (if configurable) to both tags.

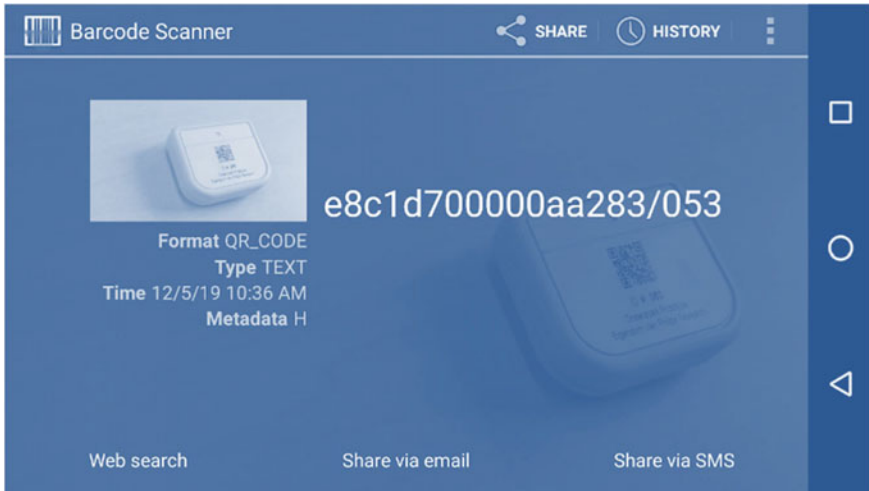


Fig. 11 Scanning UID from QR code App

4.2 Low Power Wide Area Networks (LPWAN)

Cellular IoT is a mobile network technology that connects IoT devices to the internet using the same cellular network as used by smartphones. It has been optimized for low energy operation with limited data. For healthcare IoT applications requiring operation anytime, anywhere without intermediate devices it will be the preferred LPWAN technology, allowing global operation using Mobile Network Operators (MNO) support. The disadvantage is that a subscription cost will be involved. The table below compares major LPWAN technologies:

One interesting LPWAN feature is that IoT device positioning can be done by the network provider(s) based on trilateration providing an accuracy as listed in the table. Although that may not be accurate enough for many use-cases, it may provide a rough position for any IoT device. For emergency services, a reference criterion may be the FCC's E911 mandate in the US, which requires location of emergency callers to be provided. In summary the 2020 requirement for 2D accuracy for a given set of measurements for indoor calls is that 70% has an accuracy of <50 m. Cellular solutions should preferably fit these accuracy requirements. The LTE Positioning Protocol (LPP) describes various ways to improve accuracy for cellular technologies [29]. The OTDOA (Observed Time Difference of Arrival) method using a standardized Positioning Reference Signal (PRS) allows for up to 27 m accuracy in rural areas.

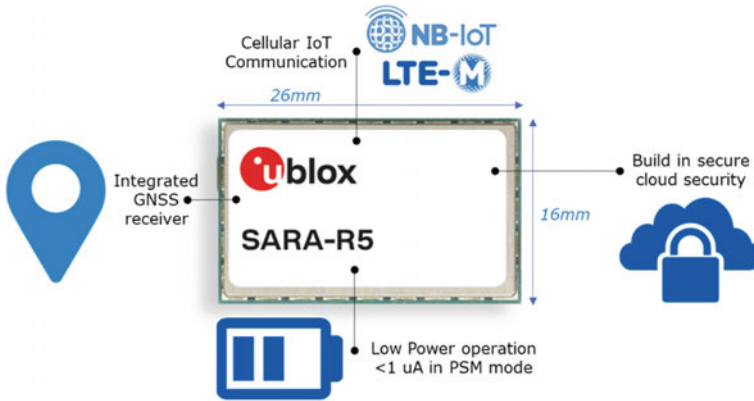


Fig. 12 Integrated cellular module with both GNSS and cellular IoT

4.3 Battery Lifetime

One important aspect to consider is the battery lifetime of tags, especially for the asset tracking use-case. These location tags typically operate on a battery that should last a year or longer to prevent the logistics burden for replacing batteries of hundreds of devices weekly or monthly. For indoor localization tags using IR, BLE or Wi-Fi this is feasible. The technologies are already quite energy efficient and by switching to a low energy mode when a tag is not moving (as detected by an accelerometer) battery lifetimes of over a year are achieved in practice. For BLE tags a battery free solution is even possible as demonstrated by the Wiliot IoT pixels devices that harvests energy from local radio waves.³ Within the EU Madras project, a solution is investigated to geolocate assets using UWB technology without a battery, also harvesting energy from surrounding UHF waves.⁴

Cellular technologies as required for outdoor localization (Fig. 12) require much more energy and obtaining a battery lifetime of 1 year or more is quite a challenge. Specifying a specific lifetime is complicated as many variables influence energy consumption (Cell coverage, network load, RF propagation, message size, power saving modes). Using LTE-M and **assuming good coverage** (CL = 154 dB, a reasonable assumption for outdoor location reporting), a (3Wh) battery lifetime of one year or more is feasible with a report interval of 1 h or more and a message size of 84 bytes [12]. Battery lifetime will scale almost linear with the reporting interval but increasing the message size up to 274 bytes will hardly decrease battery lifetime. Tests done the with MCI Logistics tags (Fig. 18) confirm these observations. Apparently, the LTE-M protocol overhead is the main determining factor here. The size of a geoJSON point type message (with empty properties) is about 120 bytes.

³ <https://www.wiliot.com/product/iot-pixel>.

⁴ <https://madras-project.eu/development-of-A-flexible-battery-free-geolocation-tag-based-on-advanced-materials/>.

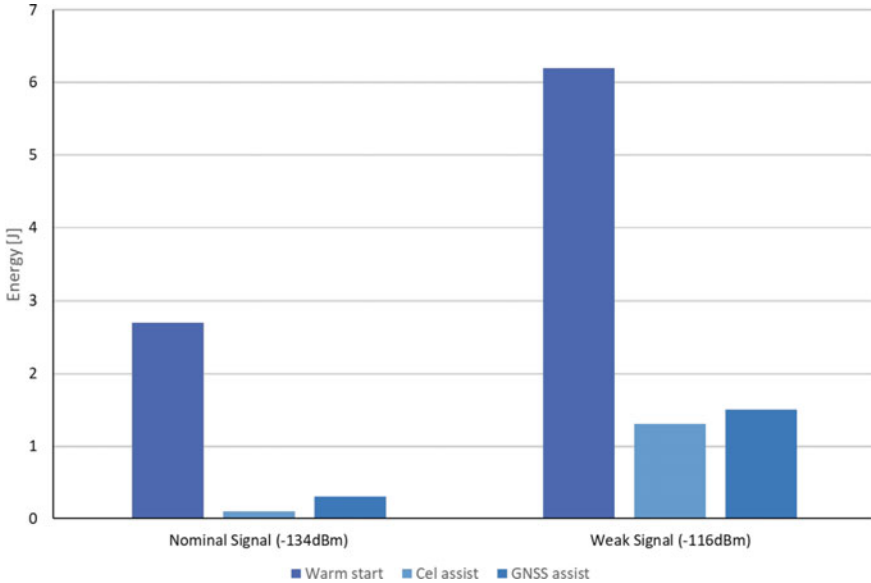


Fig. 13 Energy needed for first GPS fix (uBlox M8)

GPS energy consumption adds up significantly to the cellular device for outdoor localization. Assuming good satellite signal reception the energy needed for an hourly position fix is about 0.75 mWh (Fig. 13, uBlox M8) [13].⁵ Assuming again a 3Wh battery, 1 message/hour and a battery lifetime of 365 days for a cellular device (0.34 mWh), the battery lifetime with GPS will reduce to $3000 / (24 \times 0.75 + 24 \times 0.34) = 115$ days. When reasonably assuming an outdoor usage percentage of 50% and an indoor battery lifetime of 1 year the expected battery lifetime will be $3000 / (12 \times 0.75 + 24 \times 0.34) = 175$ days. This is well below the 1-year requirement. Further extending battery lifetime is possible by preventing outdoor location updates when the tag is not displaced, using A-GPS or snapshot positioning⁶ to reduce acquisition time(s) significantly, using solar energy and/or increasing the battery capacity. Confirmation of the above assumptions and estimates is needed with a state-of-the-art geolocation tag.

One other aspect to consider w.r.t. battery lifetime is that NB-IoT and LTE-M technologies use a high peak current for transmitting data, which cannot be provided by an almost empty battery. This will reduce the effective battery capacity by 10 ~ 20%.

When using LoRa instead of cellular transmission the required tag energy can be reduced significantly, the SemTech RL1110 geolocation chip uses 81uW or less for GPS localization [14] and LoRa transmission, resulting in a 3Wh battery lifetime of

⁵ Assuming a TTFF of 30 s and considering that the ephemerides data is only valid for a few hours, see UBLOX Hot Fix Times (u-blox.com).

⁶ Extend Battery Life with an Ultra-Low Power GPS Solution | Bench(mouser.com).

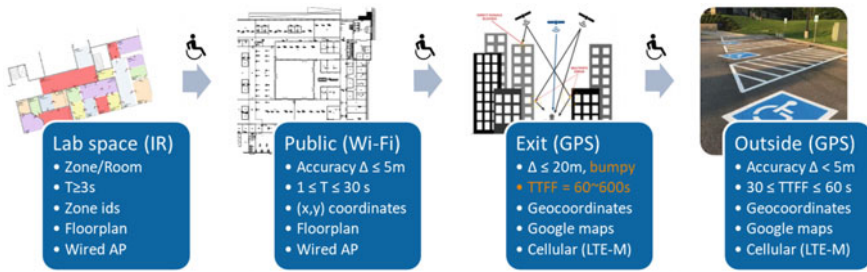


Fig. 14 Localization of assets from the lab space to the parking lot (for example with a wheelchair) requires multiple technologies

>4 years.⁷ Major disadvantage of using LoRa is the lack of global network provider support. Nevertheless, there is a huge LoRa network available using private LoRa APs operating in internationally reserved Industrial, Scientific, and Medical (ISM) bands.

4.4 Going from Indoor to Outdoor

RTLS solutions for medical device asset tracking are available from various vendors. Most solutions support indoor localization only. Room level accuracy would be preferable in most situations, but the technologies providing this (IR, BLE, UWB, ...) typically require a dedicated and thus costly infrastructure. Wi-Fi infrastructure thus would be preferred but does not provide this level of accuracy (yet). A hybrid solution may provide the room level accuracy in specific areas (such as lab spaces) and an acceptable accuracy in public spaces using Wi-Fi. Expanding localization coverage to the hospital campus or beyond with a reasonable accuracy will require GPS (Fig. 14).

Localization tags should preferably support all required technologies to prevent the logistics burden of commissioning and battery replacement of multiple tags. For the same reason battery lifetime should be 1 year or more. This can be easily achieved for the low energy and short-range technologies; it is more difficult for Wi-Fi and GNSS and cellular will be the real challenges here. To maximize battery lifetime such tags should always use the lowest energy technology and only switch to the next level when that signal is lost. Evaluation tests show that when going from indoor to outdoor in the shade of high buildings (“a hospital”), it takes considerable time to acquire a GPS position for reasons as explained in the previous section. When moving away from the building(s) the GPS location is acquired more easily and accurate.

⁷ LR1110 chip: one solution for LoRa and GNSS tracking—IoT Blog (irnas.eu).

4.5 APIs for Location Services

Multiple Technologies may be combined to improve accuracy, add features, or reduce energy consumption. For example, a location tag with both BLE and Wi-Fi may support higher accuracy localization in building sections with a BLE 5.1 RTLS system installed, whereas Wi-Fi will be used in the remaining building sections to allow for localization with less accuracy. A location tag with both GPS and Wi-Fi may use Wi-Fi to get a more accurate or quicker location coordinates in the presence of buildings with a known Wi-Fi network Basic Service Set Identifier (BSSID, typically the MAC address of the AP).

There are various API services available that provide geolocation estimates based on cell towers and Wi-Fi Access Points (AP) information as detected by the localizing device. Typical cell towers information that needs to be provided include cell id, mobile country code (mcc), location area code (lac), mobile network code (mnc) and received signal strength (RSSI). This type of information will be available at the receiver's RF module after connection to a cellular network (Fig. 15). Wi-Fi AP information typically contains the AP's MAC address (BSSID), RSSI and the Wi-Fi channel number. A Wi-Fi connection is not needed to acquire this information, detection of the Wi-Fi network already provides this information at the device side. A successful API location request will return a JSON response with a geolocation and an associated accuracy (radius) estimate. The geolocation data returned is based on location information from a large database with cell tower locations and locations of APs all over the world, acquired using public smartphone data.

For the InSecTT use-case several location APIs have been considered:

- Google's geolocation API.⁸ This is the API that is used by most Android smartphones when localization is enabled. It supports both Wi-Fi and cellular, it is well documented and is backed by a large and actual location database. For commercial use, a fee is required of about \$5 per 1000 requests.
- Here Network Positioning API.⁹ This API supports 2G, 3G, 4G, and WLAN measurements data. For cellular it supports neighboring cell measurements to improve position accuracy. Registration is required to get an API access key with options to get started for free or a flexible pay-as-you-grow pricing.
- RadioCells openbmap API,¹⁰ shown in Fig. 16. It is compatible to Google API and free to use, but their AP location database is much smaller. Unfortunately, it has recently ceased operation.
- Polte location API.¹¹ This location API claims to deliver the best cellular location possible for 4G and 5G Massive IoT devices. Their API comes in two flavors CoreRes and SuperRes. The SuperRes API requires embedded firmware support

⁸ <https://developers.google.com/maps/documentation/geolocation>.

⁹ <https://developer.here.com/documentation/positioning-api>.

¹⁰ <https://radiocells.org/geolocation>.

¹¹ <https://www.polte.com/solutions/polte-location-api/>.

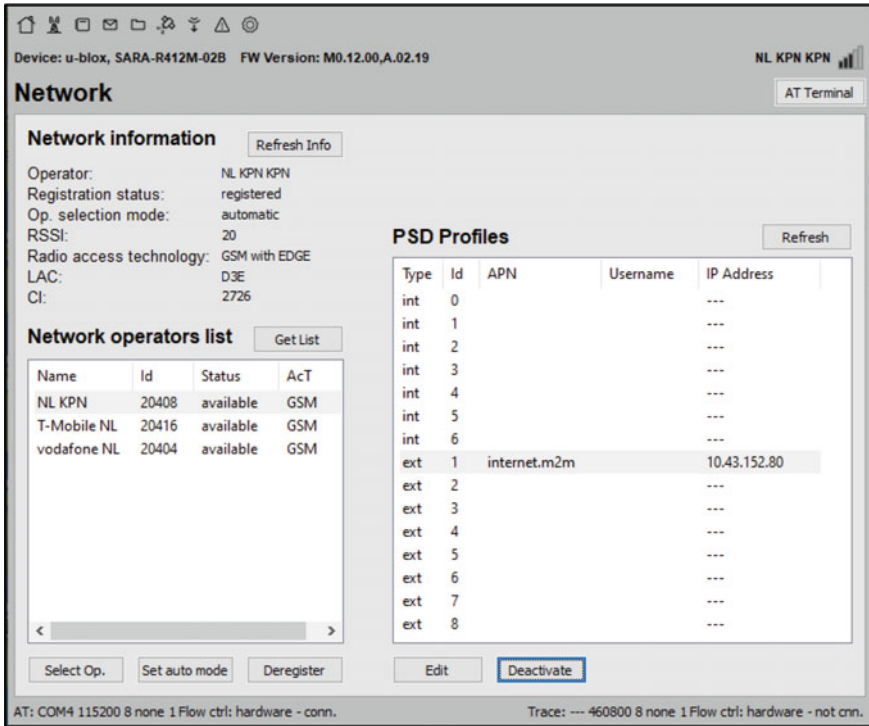


Fig. 15 Screenshot from uBlox m-center tool connected to the prototype logistics device (Fig. 18), showing the required network parameters (left) for the location API

from a cellular device and provides the highest possible accuracy for cellular localization. Unfortunately, the Polte location API does not support Wi-Fi networks localization and has recently ceased operation.

Evaluation tests show that outside localization based on cellular is up to 100 m accurate, which is not accurate enough for many applications and thus GPS functionality will be required in addition. However, the cell(s) based location (and time) may assist a GPS module to reduce TTFF. Wi-Fi network-based localization on the other side will not be available in many outdoor locations but may help to improve location accuracy in urban areas with many known Wi-Fi networks that may compensate for scattered GPS reception.

4.6 Visualizing on a Map

Visualizing the locations on a dashboard of assets both outdoor and indoor can be done with the Google maps API for Android, iOS or JavaScript (Fig. 17). A disadvantage

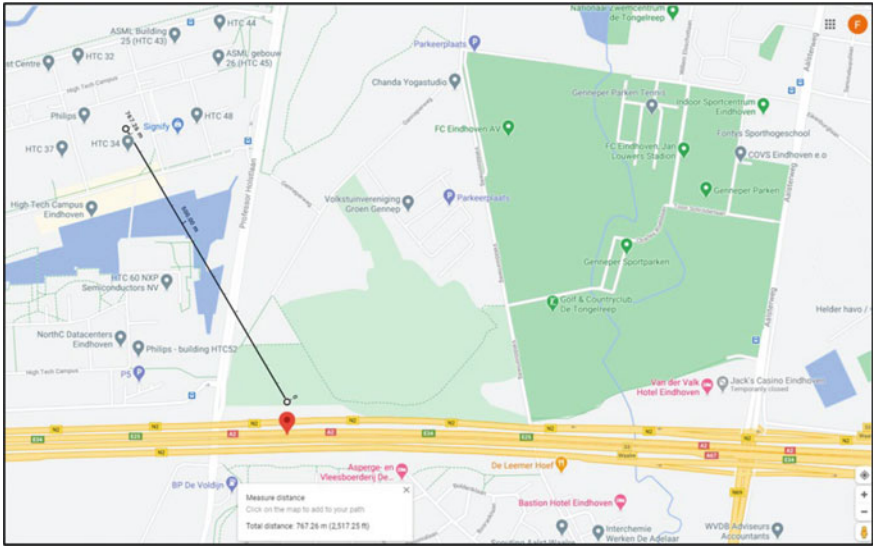


Fig. 16 Map showing the cell location as reported by the RadioCells openmap API with 1000 m accuracy, actual distance to device’s location was <800 m

of this API is that a subscription cost is involved. For the maps SDK this is about \$14 for 1000 requests. Indoor floorplans can either be uploaded to Google maps by which they become public or by overlaying them locally. Adding floorplans locally can be done in both bitmap (JPG or PNG) or vector (geoJSON linestring) format. When overlaying bitmap floorplans, the Mercator projection used by Google maps shall be taken into account. The JavaScript version of the API of public maps does not support floor level selection, so local overlays will be needed for buildings with multiple floors.

Creating a geoJSON floorplan can be done using geojson.io, a quick, simple tool for creating, viewing, and sharing maps. Floorplans can be drawn manually, but it can also import various file formats (like KML and GPX). Alternatives to Google maps are the HERE Maps API or using the less familiar but free OpenStreetMap (OSM) API.

A dashboard may also need functionality for finding geocoordinates from location names or the other way around. This can be done using a (reverse) geocoding API.

Table 4 shows the major features and costs of the above-mentioned APIs:

4.7 Security and Privacy Aspects

The EU General Data Protection Regulation (GDPR) considers health related data as a special category and provides a definition for health data for data protection

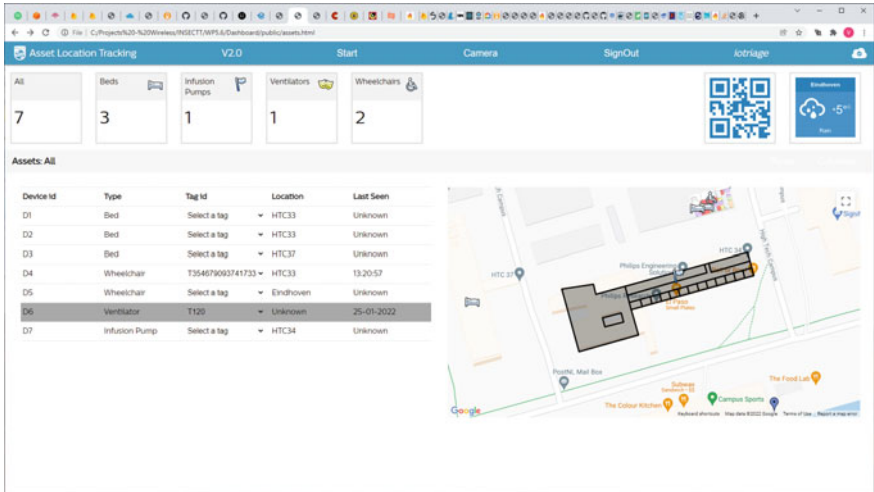


Fig. 17 Screenshot of a web dashboard showing clinical assets locations using the Google maps API with overlays for indoor floorplan

Table 4 Major features and costs of commonly used map APIs

Feature	Google maps API	HERE maps API	Open street map
Map request cost	€7/1 k	€0.06/1 k	€0
Overlay support	Yes	Yes	Yes
geoJSON support	Yes	Yes	Yes
Geocoding API	€5/1 k	€0.60/1 k	Nominatum.org

purposes [15]. Specific safeguards for personal health data are being addressed by the GDPR. Research and development activities to improve healthcare, such as clinical trials or mobile health need to comply with these data protection safeguards to maintain the trust and confidence of individuals that their data is protected. In the US, the HIPAA Privacy Rule establishes national standards to protect individuals’ medical records and other individually identifiable health information. In general location data is considered as very private sensitive information as demonstrated by the news message below.

40 Attorneys General Announce Historic Google Settlement Over Location Tracking Practices

November 14, 2022

LANSING – Michigan Attorney General Dana Nessel announced that Michigan, along with 39 other attorneys general, has reached a **\$391.5 million** multistate settlement with Google over its location tracking practices relating

to Google Account settings. This is the largest multistate Attorney General privacy settlement in the history of the U.S.

...

Location data is a key part of Google’s digital advertising business. Google uses the personal and behavioral data it collects to build detailed user profiles and target ads on behalf of its advertising customers. **Location data is among the most sensitive and valuable personal information** Google collects. Even a limited amount of location data can expose a person’s identity and routines and can be used to infer personal details.

5 Healthcare Use-Cases

Within the InSecTT project various use-cases have been considered requiring location awareness for healthcare. These use-cases involve the technologies and end-to-end aspects as described in the previous sections.

5.1 Asset Tracking

This is the simplest application because the tags used for localization of critical equipment themselves are not categorized as clinical devices and hence do not need the associated medical device certifications. Also, there are limited privacy concerns as no personal information is involved. Asset tracking may offer the following advantages [16]:

- Improved equipment efficiency (typically 30–40% utilization use) by keeping track of equipment use and locations.
- Improved process workflow as less time will be needed to find nearby equipment. On average nurses take 20 min/shift for finding required equipment.
- Improved equipment lifetime by automating registration of actual use and keep track of required maintenance.

Localization technologies used for asset tracking can also be used for tracking of patients and personnel, but in that case privacy issues need to be carefully considered. Asset tracking solutions for healthcare are already available from various suppliers, but typically for indoor localization only. Extending these solutions with outdoor localization technologies as described above will allow for a wider scope of applications.

Fig. 18 Logistics tag for MCI demo in the InSecTT project



5.2 Mass Casualty Incident (MCI)

In case of mass casualty incidents localization technologies can be used to track locations and triage status of casualties [17]. Traditionally this is done using a paper solution with triage cards attached to casualties and reporting triage status and casualty locations to an incident commander. Properly digitizing the paper solution will provide a better incident overview in chaotic situations and result in better and faster hospital transfers of casualties. Possibly even saving lives (Fig. 18).

5.3 Bed Management

Automated bed management is a specific asset tracking use-case in which not only the location of hospital beds is registered, but also the status of the beds (Unoccupied, occupied, closed, housekeeping, isolated and contaminated according to the FHIR standard, see below). Manually entering and updating this information in the HIS as typically done in most hospitals takes time and leads to inconsistencies between the actual situation and the registered situation [18]. This may be prevented by an IoT solution that updates status and location of beds with a role-based swipe of a badge (Fig. 19).

5.4 Hospital Wayfinding

Hospitals lose efficiency due to late or missed appointments related to patients arriving late because of traffic delays, parking problems or losing their way around. A navigation (and appointment scheduling) smartphone Application supporting hospital floorplans and indoor localization infrastructure for navigation may prevent these inefficiencies and annoyances [19]. QR codes as described previously may be

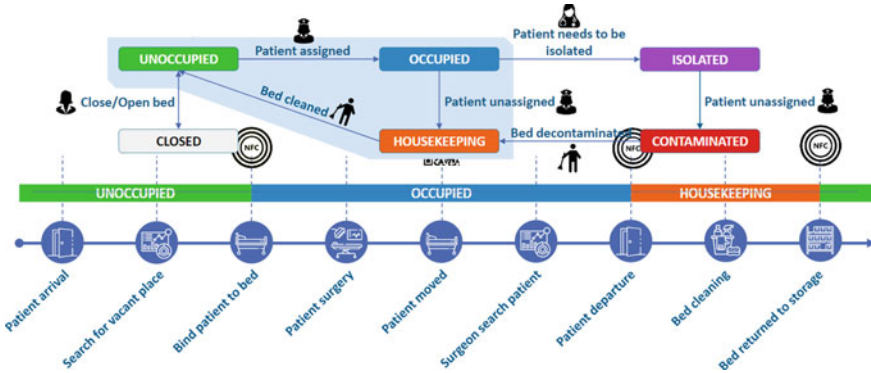


Fig. 19 Workflow for bed management using FHIR defined bed states

used for low-cost navigation. Providing patients with a location tag on arrival allows re-use of existing RTLS for navigation purposes.

6 Use-Case Concept Demonstrator

A generic architecture for an IoT logistics solution that supports the use-cases as described above is presented. It has been implemented using cellular localization devices, a geoJSON server with secure authentication, and a dashboard with map visualizations (as shown in Fig. 17). Compatibility with FHIR and location privacy aspects have been considered for the actual implementation.

6.1 Architecture

Figure 20 show a simplified architecture for an autonomous localization solution as used in the InSecTT project for tests and demo. The heart of the system is a secured REST API server with geoJSON data running on the Philips Health Suite Digital Platform (HSDP) using CloudFoundry as a service [20]. Registered location tags can upload their geoJSON data to this server and dashboard client(s) can show the reported locations on a map. This geoServer was developed in JavaScript (ES6) using the node.js framework extended with express for API routing and Postgres SQL database access. A KeyCloak server is used for Identity and Access Management (IAM) providing OAUTH 2.0 authentication and verification. Functional REST API testing was done using POSTMAN. A security assessment of the geoServer was done using the industry standard Common Vulnerability Scoring System (CVSS) [21] to remove all (known) vulnerability risks.

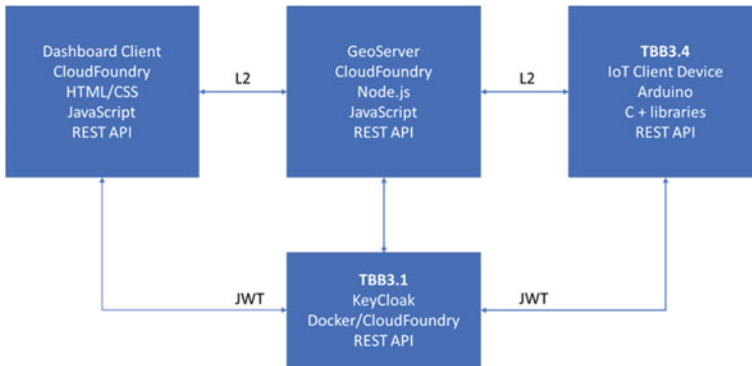


Fig. 20 Simplified architecture of an IoT location solution based on the geoJSON standard for data and KeyCloak for OAuth 2.0 authentication

Dashboards for the asset tracking (Fig. 17) and the MCI use-cases were build using HTML/CSS and JavaScript (ES6) to access the geoJSON and Google maps server APIs.

An Arduino SARA-R4 board (Fig. 21) has been used for prototyping and testing the GPS and cellular functionality for outdoor localization. For this purpose, the cellular Sodaq R4X library¹² has been updated to support HTTPS (port 443). Due to firmware (header size) limitations OAUTH 2.0 authentication could not be implemented, so basic authentication is (temporarily) used on the device side. At this point it should also be noted that mutual authentication (using client token generation and TLS server certificate validation) can be a considerable burden for constrained IoT devices, demanding for secure but more lightweight alternatives [22]. For constrained IoT devices TLS overhead may be reduced using session resumption using a client id or a ticket [30, 31]. TLS 1.3 [32] defines a new mechanism 0-RTT (zero round-trip time) that can be considered to reduce TLS overhead,¹³ but its value to IoT devices is not yet clear.

With a new generation of GPS/cellular localization devices currently entering the market, the IoT device choice is reconsidered, taking into account the requirements and concerns that have been identified from our prototype testing.

6.2 GeoJSON Server

GeoJSON is an open standard JSON format designed for representing simple geographical features, along with optional properties [23]. Features include points,

¹² https://github.com/SodaqMoja/Sodaq_R4X.

¹³ <https://crypto.stackexchange.com/questions/15209/compare-rfc-5246-sessionid-re-use-versus-rfc-5077-session-resumption>.

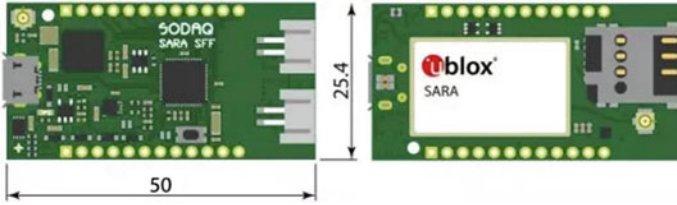


Fig. 21 Sodaq Small Form Factor (SFF) board as used for prototyping. The SARA-R4 module supports cellular IoT and a GNSS module (EVA-8 M) can be added. Software can be uploaded via the USB connector and runs on a 32-Bit ARM Cortex M0 + processor. A Tricolor LED is used for showing triage or asset status

line strings, polygons, and collections of these types. Thus, it can be easily used for any location tag reporting its geolocation as a point feature (Fig. 22).

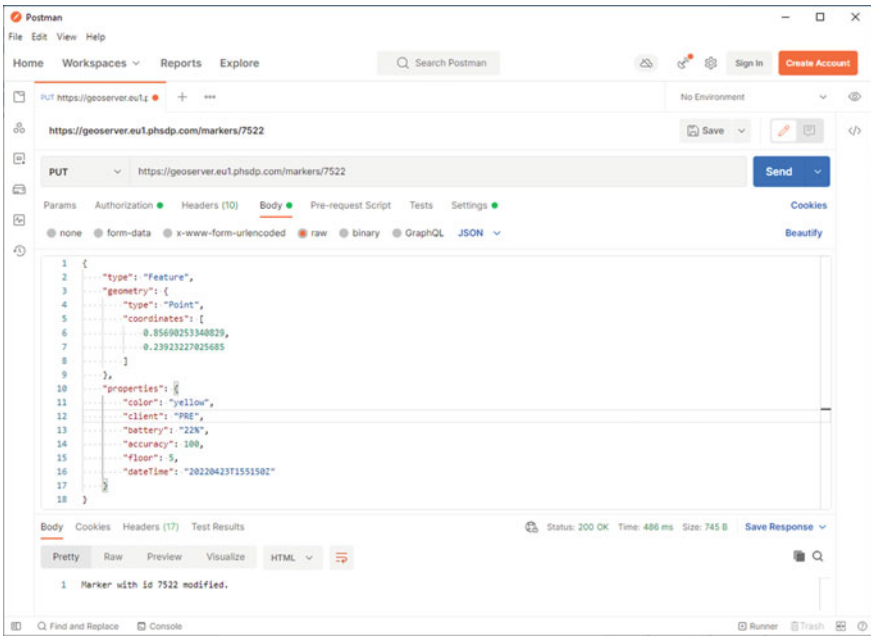


Fig. 22 Screenshot of a Postman client sending a geoJSON point with various properties (including a timestamp) and a unique id to the geoJSON backend server using a REST API

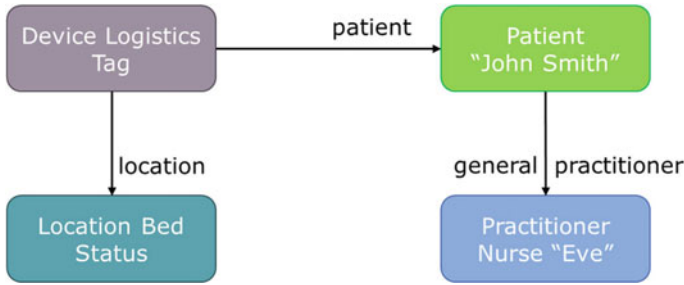


Fig. 23 FHIR resources and references involved in bed status reporting

6.3 Client Authentication

For security and privacy reasons two-way authentication is required for a location solution in healthcare [24]. When using a HTTPS REST API, the positioning device should support TLS with server certificate validation. Due to IoT device constraints Device to Cloud (D2C) authentication may be done using tokens of sufficient length (≥ 128 bits) and a limited lifetime. Token generation and transmission requires additional energy on the device, so selecting the token length and its validity period will be a trade-off between battery lifetime and security.

6.4 FHIR Compatibility

FHIR (Fast Health Interoperability Resources)¹⁴ is an international standard for Healthcare Interoperability. FHIR solutions are built from a set of modular components called “Resources” in JSON or XML format. These resources can be combined in hospital systems to manage both clinical and administrative problems. The standard is widely used in mobile applications, cloud communications, EHR-based data sharing, and server communications across the healthcare industry. FHIR server implementations are available from major providers such as Google and Microsoft. Apple is using the (SMART on) FHIR standard to enable their users to download health records and share them with participating organizations. An example of FHIR resources involved in bed status reporting is shown in Fig. 23. The FHIR location resource supports both geographic and address locations, so conversion from geoJSON to a FHIR location (and v.v.) can easily be done.

¹⁴ <http://hl7.org/fhir/>.

6.5 *Location and Privacy*

Although localization solutions do not address privacy directly, location information of persons surely is personal and therefore needs to be handled very carefully and cannot be used without permission except in emergency situations. Geographical masking health data may be used to protect the confidentiality of location records while still allowing for geographically based analyses [25]. For these reasons the geoJSON server only stores the most recent reported location and supports an offset operation on geo-coordinates.

6.6 *Additional Features*

In this section several features are described that may be considered to support or extend specific healthcare use-cases.

6.6.1 *Geofencing*

Geofencing can be used to generate an alarm on the dashboard and/or the device if an asset is moving inside or outside a specific area or building. Boundaries could optionally be set at the dashboard side and uploaded to the server possibly using the geoJSON polygon feature type.

6.6.2 *Sensors*

Additional sensors could be added to a tag as needed. For example, a temperature or humidity sensor that can be used to check the environment for the device associated with the tag. A specific healthcare example could be monitoring a medicine box for transporting medication or tissue from a patient to a lab or v.v.

6.6.3 *Analytics*

Data gathered from a location tag can be used for statistical analysis on location, transport, usage. Additional sensors or integrated tags and integration with existing FHIR services may help to properly register device specific features, for example bed occupancy.

7 Conclusions/Next Steps

In this paper an overview of most widely used localization methods was presented and how these can be applied to healthcare logistics use-cases using an IoT architecture with standardized data formats like geoJSON and FHIR with a secure REST API.

Visualizing the locations of assets or people both outdoor and indoor can be done with the Google maps API for Android, iOS, or JavaScript. A disadvantage of this APIs is that a variable subscription cost is involved. Indoor floorplans can either be uploaded to Google maps by which they become public or by overlaying them locally. When using the JavaScript version of the API floor level, selection of public maps is not possible, so using local overlays will still be needed. An alternative to Google maps is using the lower cost Here API or the less familiar but free OpenStreetMap API. A floorplan only mode (without overlay) may be included to avoid API subscription cost at all for indoor visualization.

For an RTLS solution supporting outdoor localization of assets it can be concluded from analysis and tests done that battery lifetime strongly limits the period for GNSS position reporting using cellular IoT (LTE-M). This allows for outdoor localization but prevents real-time tracking of assets for a longer period. Also, it was observed that outdoor GNSS acquisition might be difficult in the shade of large buildings (as hospitals) when moving the asset from inside to outside even when using A-GNSS. New generations of GNSS receivers designed for IoT using snapshot technology may improve significantly on tag battery lifetime allowing 1 year or more of operation. When using LoRa such a long battery lifetime will be easier to achieve, but LoRa does not support global business operation.

Multimodal solutions including UWB/BLE/IR, Wi-Fi and GNSS are required to provide an overall cost-effective solution for asset tracking both indoor and outdoor on a healthcare campus.

These findings may help in a next step for setting up a larger scale trial for asset localization with outdoor GNSS tags that fit the established functional requirements. Such a trial may provide further insights into operational requirements and fulfilment of the quadruple aim of creating better health outcomes at lower cost, while improving patient and staff experience.¹⁵ A business plan is also needed to provide a.o. business requirements, a product or service roadmap, how to handle (cellular, A-GNSS and maps API) subscription costs and a budget for market introduction [26, 27].

References

1. Completely automated CNN architecture design based on VGG blocks for fingerprinting localisation. In: International Conference on Indoor Positioning and Indoor Navigation (IPIN), Shreya Sinha and Duc V. Le (2021)

¹⁵ <https://www.philips.com/a-w/about/news/archive/blogs/innovation-matters/20191107-five-ways-in-which-the-internet-of-things-is-transforming-healthcare.html>.

2. “Bluetooth Direction Finding—A Technical Overview”, Martin Woolley, Bluetooth SIG (2021)
3. White Paper “CMX Analytics”, Cisco/Meraki (2014)
4. “Wi-Fi CERTIFIED Location”, Wi-Fi Alliance (2022)
5. Ultra-Wideband (UWB) for the IoT—a fine ranging revolution? Andrew Zignani and Stephanie Tomsett, NXP and ABI research (2021)
6. Mohammadmoradi, H. et al.: UWB physical layer adaptation for best ranging performance within application constraints. In: Conference Paper (2018)
7. Sharif-Vakili et al.: A comparison of commercial and custom-made electronic tracking systems to measure patient flow through an ambulatory clinic. *Int. J. Health Geogr.* (2015)
8. Indoor navigation with a smartphone, Drago Torkar, Jožef Stefan Institute, Slovenia (2023)
9. Narayana, S. et al.: LOCI: privacy-aware, device-free, low-power localization of multiple persons using IR sensors. In: ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN) (2020)
10. GPS compendium “Essentials of Satellite Navigation”, uBlox (2007)
11. Whitepaper “Power-efficient positioning for the Internet of Things”, European GNSS Agency (2020)
12. Power Consumption Analysis of NB-IoT and eMTC in Challenging Smart City Environments, Pascal Jorke, Robert Falkenberg and Christian Wietfeld, Communication Networks Institute, TU Dortmund University, Germany, IEEE Global Communications Conference Workshops (2018)
13. Whitepaper “Low-power GNSS for tracking applications”, Bernd Heidtman, uBlox (2021)
14. LR1110 Application Note: LoRa Edge™ Advance Scan Location Performance Overview, Semtech (2022)
15. Regulation (EU) 2016/679 of the European parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)
16. “Top 6 Applications of Healthcare Asset Tracking - Asset Utilization Drives Operational Efficiency”, Whitepaper Airista (2021)
17. Chang, C.H.: Smart MCI tracking and tracing system based on colored active RFID triage tags. *Int. J. Eng. Bus. Manag.* (2011)
18. Bed Management Optimization, Dr Ramachandran Balaji, Mark Brownlee, Infosys (2018)
19. “Wayfinding Tailored to the Patient Experience”, Datasheet Everbridge (2022)
20. Brochure “Philips HealthSuite digital platform”, Philips (2020)
21. User Guide “Common Vulnerability Scoring System version 3.1”, first.org (2019)
22. “Addressing coverage concerns for Direct-to-Cloud wearables”, Ewout Brandsma, Paul Gruijters, Henk Huijgen and Jesus Gonzalez Tejeria, IEEE (2022)
23. The GeoJSON Format, Internet Engineering Task Force, RFC-7946 (2016)
24. “The State of Healthcare IoT Device Security”, A Cynerio Research Report (2022)
25. Armstrong, M.P., Rushton, G., Zimmerman, D.L.: “Geographically masking health data to preserve confidentiality”, *Statistics in Medicine* (1999)
26. Whitepaper PerformanceFlow Solution, “How to innovate mobile asset management in hospitals by providing actionable IoT insights based on real-time location data”, Philips (2021)
27. Top 6 Applications of HealthCare Asset Tracking – Asset Utilization Drives Operational Efficiency (2021)
28. Shafran, S.V., Gizatulova, E.A., Kudryavtsev, I.A.: “Snapshot technology in GNSS receivers”. In: 2018 25th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS), St. Petersburg, Russia (2018)
29. Mahyuddin, M.F.M., Isa, A.A.M., Zin, M.S.I.M., Afifah Maheran A.H., Manap, Z., Ismail, M.K.: “Overview of positioning techniques for LTE technology”. *J. Telecomm. Electron. Comput. Eng. (JTEC)* 9 (2–13) (2017)

30. “Transport Layer Security (TLS) Session Resumption without Server-Side State”, RFC 5077 (2008)
31. “The Transport Layer Security (TLS) Protocol Version 1.2”, RFC 5246 (2008)
32. “The Transport Layer Security (TLS) Protocol Version 1.3”, RFC 8446 (2018)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Driver Distraction Detection Using Artificial Intelligence and Smart Devices



Efi Papatheocharous, David Buffoni, Matthias Maurer, Anders Wallberg, and Gonzalo Ezquerro

Abstract Distracted driving is known to be one of the leading causes of vehicle accidents. With the increase in the number of sensors available within vehicles, there exists an abundance of data for monitoring driver behaviour, which, however, has so far only been comparable across vehicle manufacturers to a limited extent due to proprietary solutions. A special role in distraction is played by smart devices, usually used while driving, such as smartphones and smartwatches. They are repeatedly a source of distraction for drivers through calls, messages, notifications and apps usage. However, such devices can also be used for driver behaviour monitoring (like driver distraction detection), as current developments show. As vehicle manufacturer-independent devices, which are usually equipped with adequate sensor technology, they can provide significant advantages and opportunities. This work illustrates the opportunities in using smartphones and wearables to detect driver distraction. The overall architecture description of the concept, called Smart Devices Distracted Driving Detection, is presented together with a series of initial experiments of a proof-of-concept. Artificial Intelligence and more especially Machine Learning is used to assess driving distractions using smart devices in a comprehensive manner.

1 Introduction

Driver distraction through secondary tasks, i.e. phone usage, is one of the major causes of road accidents [23], while the avoidance of these has been a driving force to technological advances. Such secondary tasks, pull away the drivers' eyes off

E. Papatheocharous (✉) · A. Wallberg
Research Institutes of Sweden (RISE), Kista, Sweden
e-mail: efi.papatheocharous@ri.se

M. Maurer
Virtual Vehicle Research GmbH (VIF), Graz, Austria

D. Buffoni
Tietoevry, Stockholm, Sweden

G. Ezquerro
JIG Advanced Solutions, Logroño, Spain

the road, mind away from driving and hands away from the steering wheel. Consequently, the detection of driver distraction is a popular research topic, and vehicle manufacturers increasingly implement proprietary distraction detection systems to prevent accidents. With smartphone penetration (according to connections) continuing to rise to over 75% of the population [3], the number of messages received (via email, SMS, messenger apps, etc.) is steadily increasing. Studies show that it is accustomed to check our smartphones about 6 times an hour in our daily lives to see if we have received any new messages. Especially, young drivers check their smartphone on average 1.71 times per minute for various reasons while driving, e.g., to text, surf the internet, listen to music or watch videos, whereas those who are addicted to their smartphones use them dangerously while driving [22]. However, many distraction detection applications only register whether someone is actively answering a call or using a messenger app, but not whether they are just checking their status, which appears to be a more frequent activity.

Smartwatches could be categorised as a smaller version of devices, such as smartphones [8], causing a similar frequency of notifications and status checking. While their market growth has been steadily increasing (with penetration rate at approximately at 2.69% in 2021) [1, 8], they have been reported in studies as even more distracting than smartphones while driving [6]. Nonetheless, only a very small number of studies have been conducted on the impact of smartwatch usage on driving [6].

Nevertheless, smartphones are rising as important platforms for general mobile applications (including applications developed for smartwatches) and for the transport and mobility sector in particular, called smartphone-based vehicle telematics [35]. Some examples include application-based vehicle information systems [19] or fleet management applications, which, if used appropriately, can even lead to the prevention of risky driving behaviour [21]. Research has recently looked at the use of the smartphone and integrated smartphone sensors as well as smartwatches as a basis for the development of application-based driver monitoring systems (e.g., [6, 12]). Smartphone-based driver monitoring systems can provide an important added value as they can be used to retrofit older vehicles, since they do not rely on the vehicle's sensors and actuators. They make use of a large number of build-in sensors offered by the devices, use their processors that offer high computational ability, and also make use of their efficient means of wireless data transfer and communications. Moreover, the developer availability producing applications for them has seen an unprecedented growth, due to the fact that smartphones have a huge market, and smartphone-based solutions are generally more frequently updated than vehicle-based systems, and are therefore much more scalable, upgradable, and cheap [14]. Smartphone-based solutions for the transport and mobility sector (and others) can be seen as a natural way of providing instant driver feedback via audio-visual means enabling the smooth integration of notifications and driving activity. At the same time, limitations with respect to battery, sensor quality and the fact the built-in sensors might be decoupled with respect to orientation and positioning of the vehicle, can bring challenges in their uptake.

Overall, monitoring driving behaviour, and especially distraction detection, can have a strong impact on traffic safety, but also results in fuel or energy consumption, and gas emissions improvements. Recognising or preventing such behaviour, plays an important role in generating a safety score for a driver, which can increase overall safety and promote economical driving. In addition, monitoring driving behaviour can address the needs of multiple markets: vehicle manufacturers, car insurance, fleet management and fuel consumption/optimisation.

The aim of this book chapter is to propose solutions where smartphones and wearables such as smartwatches, can reduce risky driving rather than causing road accidents. In Sect. 2 we will present definitions and the background of the driver distraction area. Then, we will describe the design of our solution (Sect. 3) proposing three Machine Learning applications for detecting drivers' distractions (Sect. 4) and a web application (Sect. 5) presenting the information as a dashboard. Before concluding, we will report the related work (Sect. 6) to our work.

2 Definitions and Background

In [27] a general overview of the term “driver distraction”, what it means, how it relates to driver inattention, types of driver distraction, sources of driver distraction, factors that moderate the effects of distraction on driving, the interference that can derive from distraction, theories that seek to explain this interference, the impact of distraction on driving performance, and safety, and strategies for mitigating the effects of driver distraction, are described. Although some inconsistencies are reported in the definitions found in the literature, and different relations with respect to inattentive driving are discussed by the authors, the following key elements emerge, characterising driving distraction [27]: driver distraction seems to involve the diversion of attention away from driving, or away from activities critical for safe driving, toward a competing activity. This activity may either originate from inside or outside the vehicle, and it may be driving- or non-driving related. Driver distraction is moreover a subset of driver inattention, and is related to something (a task, object, or person) that distracts the driver's attention needed to perform the driving task adequately [29].

Driver distraction is a very broad topic. The focus of this chapter is exclusively on methods that use the smartphone and wearables, as a sensor device, to detect distraction and to communicate detected distraction to the driver. This chapter thus describes the conceptual framework which uses Smart Devices for Distracted Driving Detection. The overall architecture description of the concept and a series of initial experiments of a proof-of-concept is presented. Artificial Intelligence (AI) is used to assess driving distractions using smart devices in a comprehensive manner.

3 System Design

The aim of our system is to implement and test different approaches that could be used for detecting distracted driving events. The philosophy behind is that a smart device can be used to detect the distraction it is causing. For example, a smartphone used by the driver can be used at the same time to predict such events. Similar idea for a wearable such as a smartwatch. However, each device's predictions concern the usage of the device itself. In addition, we include a camera-based approach which could detect the driver's activity. The camera-based solution, using computer vision techniques on images, enables detecting distractions caused by wearables without being on the device itself and can be extended to other sources of distractions such as radio manipulation, talking to a passenger, etc.

Our conceptual framework makes use of smart devices for distracted driving detection at its core. An overview of its architecture is presented in Fig. 1. It is comprised of the following 4 main components:

1. Smartphone application for driving distraction detection developed by VIF
2. Smartwatch application for driving distraction detection developed by RISE
3. Camera-based system for activity labelling developed by Tietoevry
4. Dashboard application summarising the driving session developed by JIG

The first three components are based on Machine Learning and share a similar process such as data collection, data pre-processing, model training, testing, validation and deployment. For some of them, a dedicated app is developed to perform the inference on the device itself. The dashboard application offers an offline visualisation of a driving session where distractions have been detected by the three other components. The goal is to implement a collaboratively designed dashboard including the information provided by all the Machine Learning-based components.

In the following sections we will describe the process of building each component. As the process is similar for the Machine Learning-based components, we will combine them together in one subsection.

4 Machine Learning-Based Components

In this section, we present the three Machine Learning-based applications developed for the Smartphone, the Smartwatch and the Computer vision system. We first provide their definition and describe the architecture of the applications. Then, we present them according to the Machine Learning process which is composed of three steps:

- Data acquisition and pre-processing
- Machine model training and experimental results of model fitting
- Model deployment on smart devices.

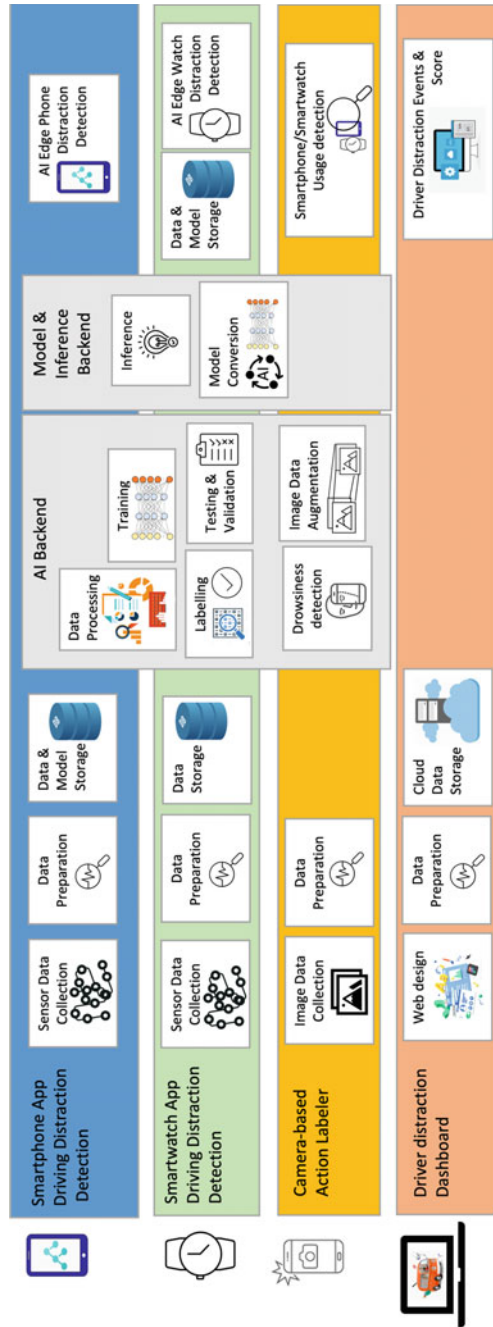


Fig. 1 Conceptual framework of the system design for distracted driver monitoring with smartphones and wearables

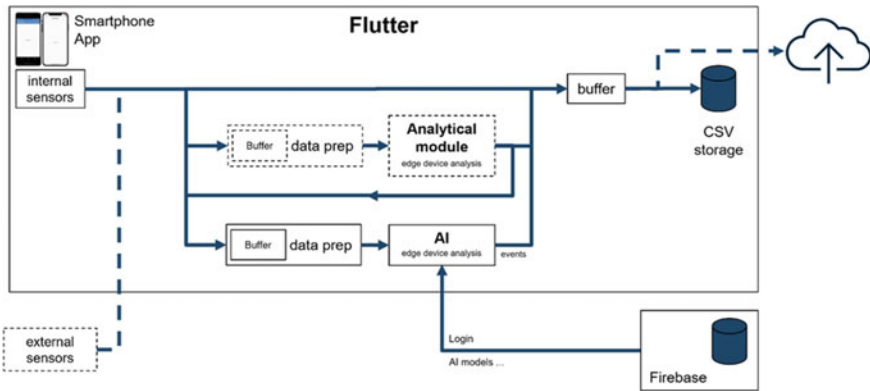


Fig. 2 Smartphone application sub-components architecture

4.1 Use Case Definition and Components' Architecture

4.1.1 Smartphone Application

This component is responsible for providing a smartphone application, capable of collecting sensor data, analysing and preparing the data for driving distraction detection caused by the use of the smartphone. Previous work of an author-centric literature review [20] showed that most driver distraction detection systems are based on proprietary hardware, and there is comparatively scarce research on how to use the smartphone as a sensor and to detect critical events on the edge device. While proprietary hardware often prevents data access and thus makes it difficult for researchers to compare various algorithms, accuracy and trustworthiness of the results, smartphones are increasingly powerful devices that are always connected to the internet. Therefore, the development for this component is targeted to offer an accessible, efficient and ordered method for data collection for research purposes. To achieve this a dedicated smartphone application is developed to detect potential smartphone usage by the drivers. Thereby, the application is capable of collecting smartphone sensor data, which later on is used in a Machine Learning model. Data collection includes data cleaning, pre-processing and storing. We explore to what extent it is possible to run Machine Learning models on smartphones, given the increasingly availability of powerful smartphones. As a privacy requirement, data may alternatively chosen to not leave the users' smartphone. Hence, driver phone usage is detected via smartphone sensor data classification using a model on the smartphone.

The smartphone application is implemented using Flutter¹ and Android as the target platform. The application's modules are shown in Fig. 2. Raw values from the smartphone sensors are collected and can be processed in three alternative ways: (i) pass-through: the raw values are transmitted to the storage without being processed,

¹ <https://flutter.dev/>.

(ii) analytical: the raw values are going through an analytical module where the raw values are modified by applying mathematical operations on them (e.g., apply rotation on Inertial Measurement Unit (IMU) values), (iii) edge device AI: the raw values are passed to an AI model and the output of the AI model is then stored. The described processing ways may also be combined. The following smartphone data is collected:

- Timestamp: absolute time of the sensor recording,
- IMU (gyroscope and accelerometer) with customisable frequencies ranging at 1–20 Hz (5 Hz steps),
- GPS coordinates with customisable frequencies ranging at 1–5 Hz (1 Hz steps),
- Screen state, and,
- Moving state (i.e., walking, running, biking, or driving).

The goal of the application is to be able to predict distraction and attention states based on smartphone data only.

4.1.2 Smartwatch Application

Smartwatches are becoming increasingly a source of driving distraction, as they can be used for calling, texting, and receiving notifications. At the same time, they can be used to detect driver smartwatch distractions given that they are fed with appropriate smartwatch distraction data. This component is responsible to perform exactly this, after collecting smartwatch usage data while driving and running Machine Learning models. Thus, the smartwatch application collects sensor data and shows basic trip information (like trip date, duration, and distance covered) along with distraction data. The development is targeted for data collection and driver training for research purposes. The smartwatch applications' sub-components and data flow are shown in Fig. 3.

The diagram supports the following scenario: the application for driver distraction is initiated and used to inform the driver about occurring distraction events, happening during trips. The driver is using a smartwatch application that collects sensor data of wrist movements and a companion smartphone application that collects sensor data from the smartphone. A video camera is used at the same time to record the trip. The recording is used for labelling purposes only. The video camera captures only the body parts of the person so that the person is not identifiable. Then, a labelled dataset is produced, which is then used for Machine Learning model creation, training, testing and validation. The trained model is then included in the smartwatch application for driver distraction detection. A sound is played when the application detects a distraction. The following smartwatch data is collected:

- Timestamp: absolute time of the sensor recording,
- Seconds elapsed: time expressed relative to the start of the data collection session,
- IMU (gyroscope, gravity and accelerometer)–with customisable frequencies ranging at 1–200 Hz,

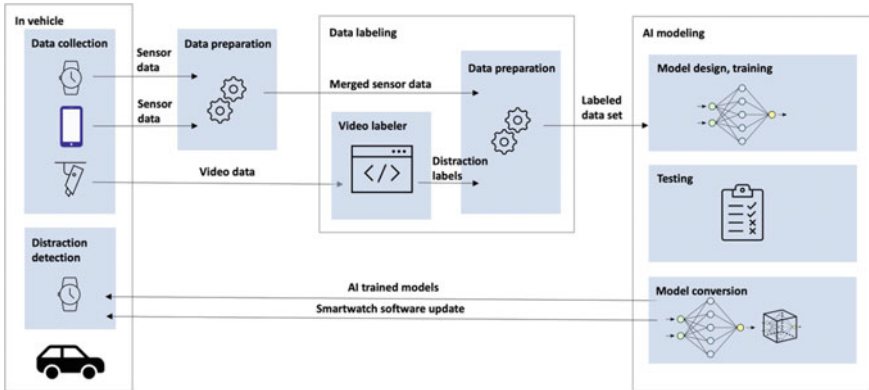


Fig. 3 Smartwatch application sub-components architecture and data flow

- GPS latitude and longitude,
- Activity: a value corresponding to an activity which might involve distraction or not,
- Distraction: a boolean value corresponding to a distraction or not (1–distracted/0–not distracted).

The application helps drivers detect distracted driving and summarises at the end of the trip the start time of the trip, duration and the times a distraction took place. A history of trips is also made possible to view. Finally, all trip data collected on the smartwatch can be at any time deleted by the user.

4.1.3 Computer Vision Application

One potential limitation of the two above applications is that they are dedicated for a particular smart device and a particular set of distraction events. Therefore, we propose an additional application of a computer vision solution, to extend the scope and include detection of distracted driving based on video recordings. These distractions are not necessarily limited to the usage of smart devices but can be generalised to other behaviours such as talking to a passenger, reaching for something behind the driver, etc. Our motivation relies on the automotive industry roadmap where vehicles embed more and more sophisticated systems to improve passengers’ safety. In particular, in the EuroNCAP roadmap [25], it is reported that an increasing level of road assistance is expected and systems are able to ensure that the driver remains engaged in the driving task such as having hands on the steering wheel and eyes on the road.

In case of implementing an efficient and accurate computer vision system for detecting drivers’ distractions, this application could also be used for labelling the data generated by the smart devices and be responsible for keeping such data in the

vehicle. Aim of this computer vision-based solution is, therefore, defined to build a Machine Learning algorithm able to classify the following 3 events: driving normally, using smartphone and using smartwatch based on drivers' images. To achieve that, we will follow a transfer learning procedure to train a computer vision model on a custom dataset of images, as explained below.

Training a computer vision model usually demands a lot of data. In our situation, taking into consideration that we want to execute the algorithms inside a vehicle where the computation resources would be limited, and thus we are restricted with limited resources, we combine an openly available dataset of images (the Statefarm dataset [18]) and one dataset that we created on our own. A lot of prior work exists in the literature where computer vision algorithms have been trained and tested on the Statefarm dataset. Most of them are able to perform well, frequently reaching to more than 96% accuracy [4, 15, 30, 31]. Based on that, we will report the performance of the computer vision state-of-the-art algorithm developed on our custom dataset.

4.2 Data Acquisition and Pre-processing

4.2.1 Smartphone Application

The smartphone data collection was conducted in two test drives where participant drivers performed certain actions. Each action is related either to a distraction state or an attention state. Each test drive took about 140 minutes and different cars and drivers participated. IMU data was collected with 50 Hz and GPS with 1 Hz. Furthermore, the distraction state was collected by a co-driver, being either distraction or no distraction. We only considered smartphone-induced distraction, more precisely phone calls and applications usage. We considered 4 classes, two classes when the smartphone is active (the smartphone is used to place a phone call or for app usage) and two other classes when the smartphone is resting (smartphone is in the middle console or on the smartphone holder).

Data pre-processing describes the process of combining the different data sources and unifying the data. In our case, each test drive generated two CSV-files (driver and co-driver data) which then were combined. Faulty data was removed and a combined time reference frame was created. The final dataset contains for each timestamp eight different measurements, i.e., three acceleration measurements, three gyroscope measurements, and two GPS measurements.

4.2.2 Smartwatch Application

The smartwatch data collection was carried out in three test drives. Each test drive took approximately 120 minutes and two persons (one male and one female) participated. Videos were recorded during the drives and were used for labelling the distractions (cf. Fig. 4). Similar rate in frequencies as for the smartphone were used

for the smartwatch sensors. Gyroscope, gravity, and accelerometer data was collected at 50 Hz and GPS at 0.1 Hz.

Once the sensor data were collected, human annotators start the labelling process based on the video recording of the events of a driving session. Then, for each driving session three CSV-files are generated, which were combined together (see Fig. 3). The first one from the time series of the wrist motion containing gyroscope, gravity, and accelerometer measurements, the second file contained the location at different timestamps and the third one with the label of the time period such as distraction or attention. The label distraction was used for those intervals when a person was looking at the watch screen, or interacting with the watch by tapping on the screen or using the scroll wheel on the side of the watch. This dataset is highly imbalanced where $<5\%$ of the events are annotated as distractions.

4.2.3 Computer Vision Application

The data used for the computer vision-based application is composed of two different datasets. The first dataset is an openly available dataset [18] which was used in a Kaggle competition: the Statefarm Distracted Driver prediction [2]. It is composed of 100, 000 images where 20, 000 of them are annotated and belonged to 10 classes, e.g., driving normally, manipulating radio, texting with left/right hand, calling with left/right hand, etc.

However, in this dataset, there was no class of events where the driver is distracted by a wearable device, such as a smartwatch. To overcome this limitation, we decided to augment the Statefarm dataset by collecting data ourselves by performing a dozen of driving sessions of 10–15 min each and use video recordings of these sessions. Each driving session was recorded using a smartphone installed on the passenger

Fig. 4 Video recorded during test drives for driver distraction



headrest of the seat, to capture at the same time, both the side profile of the driver and the steering wheel. This setup mimics the setup used in the Statefarm dataset. However, the type of camera, the resolution, the angle and the colour of the images are different which resulted in a heterogenous dataset (see Fig. 5). Two drivers, a male and a female, participated in these sessions. Images were extracted from the video recordings and labelled according to the three following categories: driving normally, using smartphone or using smartwatch.

The way we constructed our own new training dataset, based on the Statefarm one and our own recordings, is briefly explained. From the Statefarm dataset, we used the driving normally class, we combined the 4 classes related to the usage of a smartphone (texting left/right, calling left/right) into one class and we removed the remaining 5 other classes. Then, we added all the images of the same driver into this training set. We built a separate test set, where we manually annotated images from the Statefarm test set (using only the driving normally and using phone classes) and added the second driver from our own dataset, The statistics of this dataset are presented in Table 1.

We also adjusted the images to have the same resolution, so we resized the images from both the training and test sets into a 224×224 resolution.

The minority class, using smartwatch, has only a few hundred examples compared to the two others which are bigger. This may impact the performance of our Machine Learning-based models.



(a) Image taken from the Statefarm Distracted Driver Kaggle Competition [17]



(b) Image taken from our own driving sessions

Fig. 5 Example of two images from the training set showing the heterogeneity of the data. On the left, a picture from the Statefarm competition representing the pictures included in the Statefarm dataset [18]. On the right, an annotated image taken from a driving session. Original image size, angle, color and view are different

Table 1 Computer vision dataset statistics: number of images per category

Dataset	Driving normally	Using smartphone	Using smartwatch
Statefarm (training)	2490	9260	0
Own dataset (training)	296	253	182
<i>Total training set</i>	2786	9513	182
Statefarm (test)	31	70	0
Own dataset (test)	139	57	55
<i>Total test set</i>	170	127	55

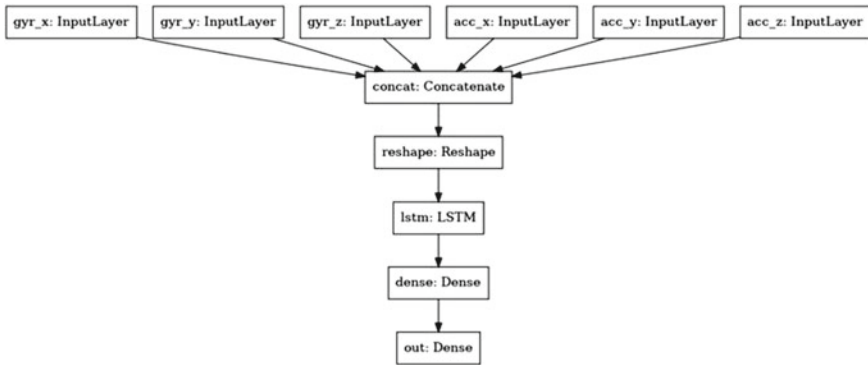


Fig. 6 The neuronal network architecture used for smartphone distraction detection

4.3 Machine Learning Model Training and Experimental Results

4.3.1 Smartphone Application

Fitting a model to the smartphone data was done by using a Long Short-Term Memory (LSTM) layer neuronal network (NN) design. An overview of the NN architecture can be seen in Fig. 6. It contains 6 neurons in the input, 32 neurons in the central, 16 neurons in the hidden and 2 neurons in the output layer. We train the model from scratch based on the collected data.

For the smartphone data, a 70%/15%/15% data split was used to create training, validation, and test data. The training data was used to adapt the model’s weights. The validation data was not directly used in the training process but was used to evaluate the current model’s performance independently of the training data. Based on the validation data, the training process was completed. The test data gives an unbiased evaluation of the model with not at all involved data in the training/validation process. Models are evaluated with the accuracy measure (see below).

We carried out experiments by fitting models to predict the right state (distraction or attention) based our collected IMU dataset. The best model achieved an accuracy

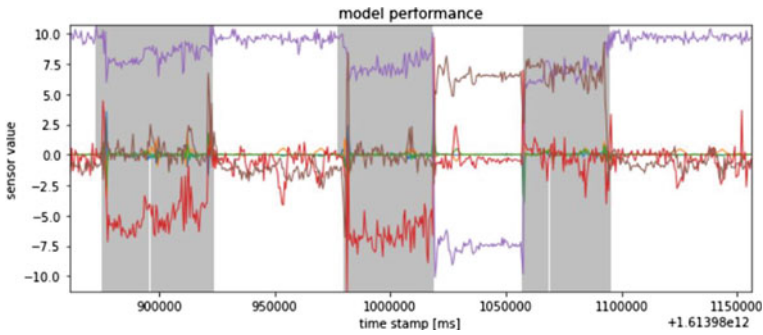


Fig. 7 Model performance of the AI model created for the **smartphone data**. Lines represent IMU measurements, the background the distraction state (whereas grey represents distraction). The top background represents the labels, the bottom background represents the model output

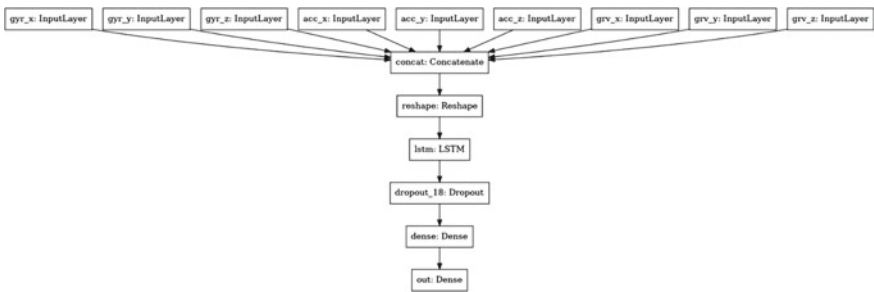


Fig. 8 The neuronal network architecture used for smartwatch distraction detection

of 94%. Thereby, especially the transitions between distraction and non-distraction state were responsible for the error, see Fig. 7.

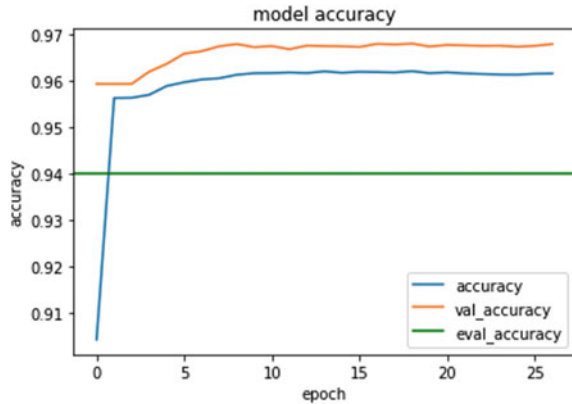
4.3.2 Smartwatch Application

In this application, as similar data were collected as in the smartphone application, we used a similar Machine Learning model as well. A LSTM based model has been designed and is presented in Fig. 8. To deal with the overfitting as our dataset is small, we added a Dropout layer after the LSTM one. The architecture comprises of 9 neurons in the input, 4 neurons in the central, 16 neurons in the hidden and 1 neuron in the output layer. Then we trained the model from scratch on our dataset to detect when a smartwatch distraction occurs.

Similar to the smartphone application, a 70%/15%/15% data split was used to create training, validation, and test data and the models are evaluated with the accuracy measure (see below).

We carried out experiments by fitting models to predict the right state (distraction or not) based on the dataset we collected during our driving sessions. The best model

Fig. 9 Model's performance for the smartwatch data. The training, validation and test accuracy is reported. After a few epochs the model has already converged



achieved an accuracy of 94% on the test set and an evolution of the performances is reported in Fig. 9.

4.3.3 Computer Vision Application

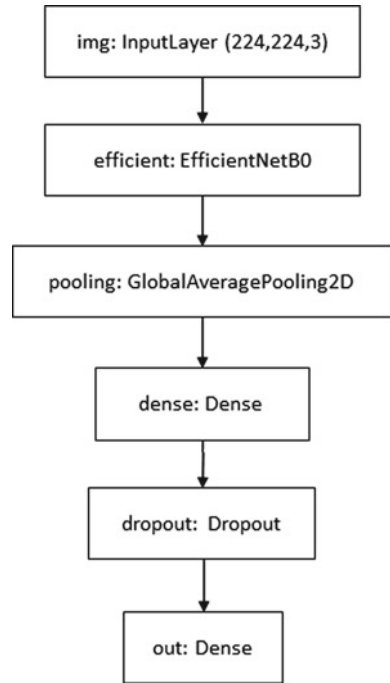
In our experiments, we tested various computer vision state-of-the-art algorithms such as VGG16 [32], Xception [7], MobileNetV2 [28] and EfficientNetB0 [34]. We opted for a transfer learning approach on our custom driver detection dataset with pre-trained² models. Based on the preliminary results we obtained, we focused on the model which was the best compromise between accuracy and size. As a recall, the end goal is to be able to train and run the model on an embedded system inside a vehicle where the computation resources would be limited. Models based on EfficientNetB0 appeared to offer the best compromise between accuracy and model size. An overview of the transfer learning model architecture can be seen in Fig. 10.

However, all models' performances were quite low, especially for detecting a smartwatch activity. Several reasons may explain this situation such as the fact that smartwatch activity is the minority class (with 2 orders of magnitude less than the majority class) and the images from this class were different from the original Statefarm dataset with only one driver represented (see Fig. 5).

To tackle this problem, we carried out experiments where we used data augmentation techniques to improve the robustness of our model. To do that, in each batch provided to the model, several geometrical transformations were randomly applied on each image:

² Models have been trained on the Imagenet dataset [9].

Fig. 10 The neuronal network architecture used for camera-based distraction detection



Zoom: [0.5, 2.5]

Brightness: [0.5, 2.5]

Rotation: [-30, 30]

Flip: {Horizontal}

*Shift*_{Height,Width}: [-0.3, 0.3]

We used 90%/10% splits of the training data where the first split was used for training and the second one for validation. We used an early stopping strategy when the validation loss was not decreasing for 3 consecutive epochs. Then, we evaluated the model on a separate test set, where we computed the F1 score for each class (Driving normally, Using Smartphone and Using Smartwatch) and the overall accuracy. Both measures are defined below:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$F1 = \frac{2 * TP}{2 * TP + FP + FN}$$

where:

- *TP*, the True Positives are the instances correctly predicted as Positive.

Table 2 Computer vision models' performances

Model	Accuracy	Driving normally (F1)	Smartphone (F1)	Smartwatch (F1)
EfficientNetB0	0.54	0.63	0.61	0.27
EfficientNetB0 + Data Augmentation	0.72	0.82	0.70	0.48

- *TN*, the True Negatives are the instances correctly predicted as Negative.
- *FP*, the False Positives are the instances wrongly predicted as Positive.
- *FN*, the False Negatives are the instances wrongly predicted as Negative.

The performances of a vanilla version of EfficientNetB0 and one using Data Augmentation are reported in Table 2.

Data Augmentation techniques helped the EfficientNetB0 model to outperform the vanilla version and overcome some of the challenges from the data imbalance and heterogeneity. However, to increase the performance of our model, more data should be collected. More specifically, additional data for the smartwatch class should be collected as it drastically impacts the performance of the model compared to the other two classes.

4.4 Model Deployment on Smart Devices

4.4.1 Smartphone Application

In our smartphone application, each model used for the purpose of training and inference can be selected and changed. This ensures to always serve a model which is performing well in production. To achieve that, we used Tensorflow Lite³ to deploy the AI model on the smartphone. The available models are stored on a Google Firebase⁴ storage which can be easily updated without redeploying the application, subject to the condition that the number and type of the input parameters are not changed. Whenever the application is started, the most recent models are fetched from Firebase and updated in the application. The deployed model takes the IMU data as input and then computes the predictions. It returns a distraction boolean as an output of either 1–distracted or 0–not distracted.

For demonstration purposes, a sound is played when the model detects a distraction.

³ <https://www.tensorflow.org/lite?hl=en>.

⁴ <https://firebase.google.com/>.

Table 3 Computer vision models' sizes and performances

Model	Size (MB)	Accuracy	Driving normally (F1)	Smartphone (F1)	Smartwatch (F1)
EfficientNet	19.9	0.72	0.82	0.70	0.48
Quantized EfficientNet	4.9	0.71	0.81	0.69	0.45

4.4.2 Smartwatch Application

In our smartwatch application, the goal is to deploy the trained models on the wearable devices. These devices have low computation and memory resources. To face this challenge, we proposed to convert the trained model to run efficiently on edge devices such as smartwatches. We opted to the approach of converting the model from being expressed in a TensorFlow format to a CoreML⁵ model, that is then added to the smartwatch during the compilation time of the application.

Currently, a beta testing service called Testflight⁶ is used, which is able to do over-the-air push updates of both the model and the smartwatch application to a group of beta testers of the application.

4.4.3 Computer Vision Application

Deep Learning computer vision models are usually computationally heavy and having the end goal to be able to deploy them on a low-resource device such as smartphone or a micro-controller inside the vehicle, makes the challenge even greater. Moreover, video/image processing approaches are greedy in terms of resources and different techniques exist to produce a lightweight model for inference.

We have optimised our computer vision model with Tensorflow Lite where the optimising parameters have been removed, the weights have been quantised using a lower-precision representation and then compressed. We evaluated the performance of this lightweight model and compared it to the original one (see Table 3).

As we can see, we were able to reduce the size of the model by a factor of 4 with relatively maintaining similar accuracy performances.

5 Dashboard Application for Driver Distraction

The three Machine Learning components already introduced offer drivers' distraction predictions based on data from smart devices' usage. To integrate these concepts into

⁵ <https://developer.apple.com/documentation/coreml>.

⁶ <https://developer.apple.com/testflight/>.

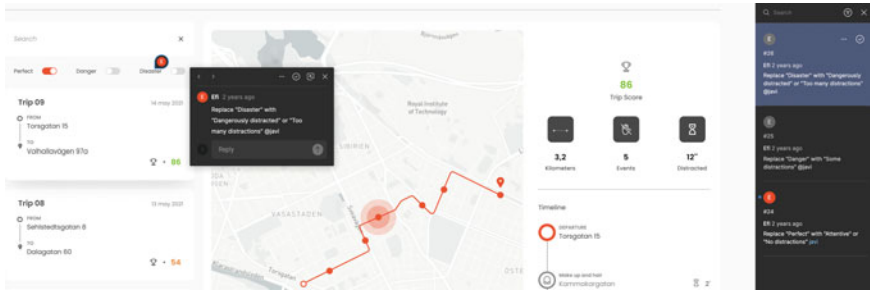


Fig. 11 A screenshot of the design using the Figma Design tool

one product, we implemented an end-user application that comprises of a dashboard summarising the distraction events occurring during a driving session.

To achieve this, we designed an application based on the Javascript framework VUE⁷ to be developed as a web application. Our development team comprises of members from different countries and companies. To this end, we analysed different collaborative development tools that would allow for a better work methodology that would lead to greater agility and ease of working together. We previously internally tested InVision⁸ and the Adobe XD⁹ set of services with the following results:

- Adobe XD: A lot of working power and many associated services but it is not comfortable to work collaboratively with other departments/companies. The acquisition cost of this service is expensive.
- InVision: Better quality/price ratio, and more ability to work together, but very focused on design, without much integration with development frameworks.

We finally chose to use Figma¹⁰, after comparing it with the other frameworks, since it has a high capacity for group work, supporting both internal and external team members, and does not need to be installed since it can be managed via a web browser. It also has the advantage of facilitating the implementation and evolution of the design to a greater extent than the rest of the options.

After this initial analysis, Figma has allowed us to carry out the entire design suggestion so that all team members involved, could see the changes and the design evolution in real-time. The team was also able to add comments and details in each of the parts of the prototype as can be seen in Fig. 11.

At the same time, Figma has facilitated the transition from design to code due to its power to control parts that until now were external to the design, such as the CSS or HTML coding of the visual proposals. This ease of implementing the layout of the components developed with the proposed design is possible because when a

⁷ <https://vuejs.org/>.

⁸ <https://www.invisionapp.com/>.

⁹ www.adobe.com.

¹⁰ www.figma.com.

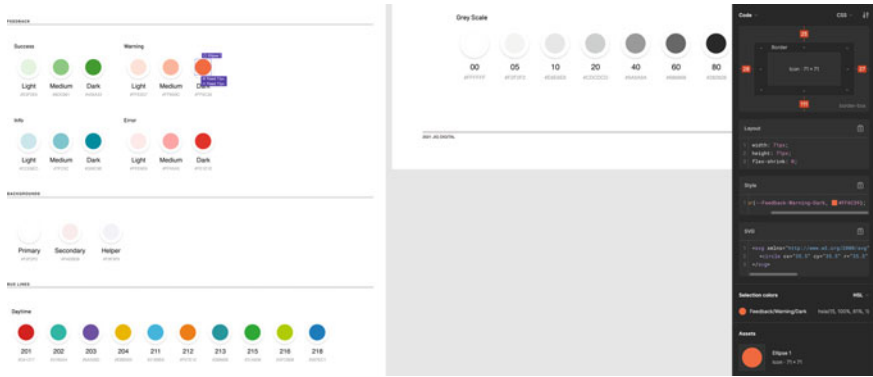


Fig. 12 Transition from design of the final dashboard layout to code

component is created in the system, its CSS code is created and exported directly to the developed code, as shown in Fig. 12.

Apart from the better joint collaboration and communication when proposing and developing the design, reducing the communication time through other means (i.e., sharing data in sharepoints, sending mails, etc.) the implementation of this application in a real environment was reduced to less than half, as Figma itself generates the code in the language of our choice (CSS, PHP, Java, etc.) making its use effective and straightforward.

In conclusion, the use of a framework such as Figma in collaborative projects has offered us greater speed, control, and efficiency in developing web applications from design to implementation.

6 Related Work

Using smart devices to recognise driver distraction from data extracted from wearables, smartphone and onboard diagnostics to obtain sensing information such as accelerometer, gyroscope, etc. is an active research area that has been gaining interest due to the increasingly computational power smart devices offer. This section summarises related recent works in the topic.

In [24] a 4-step methodological framework is presented for driving analytics to understand driving behaviour based on smartphone data. In [17] smartphone and smartwatch data is used to detect distraction while driving, like controlling the infotainment system, drinking/eating, as well as smartphone usage. Owens et al. [26] describe the coding efforts of an accessible dataset of driver behaviour and situational factors observed during distraction-related safety-critical events and baseline driving epochs. Data coding includes frame-by-frame video analysis of secondary task and hands-on-wheel activity, as well as summary event information. Deep learn-

ing on video and sensor data is proposed by [33], in a system called DarNet, capable of detecting and classifying distracted driving behaviour. To minimise privacy concerns, the system is using a distortion filter applied to the video data before processing the data.

Baheti et al. [5] use a dataset for distracted driver posture estimation and classifies images to the following 10 classes: driving, texting on mobile phones using right or left hand, talking on mobile phones using right or left hand, adjusting radio, eating or drinking, hair and makeup, reaching behind and talking to passenger. They use convolutional neural networks (CNNs) and report to achieve 96.31% on the test set.

Dua et al. [11] developed a Machine Learning-based system that uses the front camera of a windshield-mounted smartphone to monitor and rate driver attention by combining multiple features based on the driver state and behaviour such as head pose, eye gaze, eye closure, yawns, and use of cellphones. Ratings include inattentive driving, highly distracted driving, moderately distracted driving, slightly distracted driving, and attentive driving. The evaluation with a real-world dataset of 30 different drivers showed that the automatically generated driver inattention rating had an overall agreement of 0.87 with the ratings of 5 human annotators for the static data set.

Dua et al. [10] aim to identify driver's distraction using facial features (i.e., head-pose, eye gaze, eye closure, yawns, use of smartphones, etc). The smartphones' front camera is used and three approaches: in the first, convolutional neural networks (CNNs) are used to extract the generic features and then a gated recurrent unit (GRU) is applied to get a final representation of an entire video. In the second approach, besides having the features from a CNN, they also have other specific features, which are then combined using a GRU to get an overall feature vector for the video. In the third approach, they use an attention layer after applying long short-term memory (LSTM) to both specific and facial features. Their automatically-generated rating has an overall agreement of 0.88 with the ratings provided by 5 human annotators on a static dataset, whereas their attention-based model (third approach) outperforms the other models by 10% accuracy on the extended dataset.

Eraqi et al. [13] aim to detect ten types of driver distractions from images showing the driver. They use (in one phase) the rear camera of a fixed smartphone to collect RGB images, in order to extract the following classes with convolutional neural networks (CNNs): safe driving, phone right, phone left, text right, text left, adjusting radio, drinking, hair or makeup, reaching behind, and talking to passenger. Thereby, they run a face detector, a hand detector, and a skin segmenter against each frame. As results, first they present a new public dataset, and second their driver distraction detection solution performs with an accuracy of 90%.

Janveja et al. [16] present a smartphone-based system to detect driver fatigue (based on eye blinks and yawn frequency) and driver distraction (based on mirror scanning behaviour) under low-light conditions. In detail, two approaches are presented, while in the first, a thermal image from the smartphones RGB camera is synthesised with Generative Adversarial Network, and in the second, a low-cost near-IR (NIR) LED is attached to the smartphone, to improve driver monitoring under low-light conditions. For distraction detection, statistics are calculated if the driver

is scanning his/her mirrors at least once every 10 seconds continuously during the drive. A comparison of the two approaches reveals that, “results from NIR imagery outperforms synthesised thermal images across all detectors (face detection, facial landmarks, fatigue and distraction).” As a result, they mention a 93.8% accuracy in detecting driver distraction using the second approach, the NIR LED setup.

Most of the related works use smart devices to address driving distraction considering the wider view of driver behaviour and driver monitoring. Only a few of these works focus on distractions caused by the use of smart devices. Moreover, none of these works operate in real-time, gathering and detecting smart devices usage while driving, and neither rely on a wide range of driver distraction collection methods (utilising within one framework smartphones, smartwatches and camera sensors).

7 Conclusion and Future Work

Distracted driving due to smart mobile devices usage, like smartphones or smartwatches, increases the risk of accidents. There are directly related to these devices usage events of interest, like texting, browsing the web, calling, using applications, etc. To prevent distracted driving, many approaches focus on particular types of distractions.

This work demonstrates a concept called Smart Devices Distracted Driving Detection. The overall architecture description of the concept and a series of initial experiments of a proof-of-concept are presented. Artificial Intelligence and more especially Machine Learning is used to assess driving distractions using smart devices in a comprehensive manner. Based on the experiments we carried out, Machine Learning models running on smart devices demonstrated good prediction performance on test data. The computer vision model, aiming in detecting distractions by adding an external point of view of the smart devices, under-performs compared to the models on the smart devices. We also validated that the models can be deployed on the smart devices themselves without trading too much prediction performance. Moreover, a dashboard application was developed for showing to the user the occurring distraction events predicted by the models after the driving sessions.

As a future direction, collecting and annotating additional image data of distraction events would be a way of improving the prediction performances of the computer vision model. Then, labelling of driving distractions is a privacy-preserving and laborious task, if not done automatically. Computer vision algorithms may offer a complement to the predictions made on smart devices and could also be used to perform efficient data labelling. Moreover, as models have been deployed with success on smart devices, investigating the Federated Learning setting is a natural next step where data remain on the smart-devices instead of being transferred outside.

Acknowledgements This work is partially funded by the InSecTT project (<https://www.insectt.eu/>). InSecTT has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876038. The JU receives support from the European Union’s Horizon 2020 research and

innovation programme and Austria, Sweden, Spain, Italy, France, Portugal, Ireland, Finland, Slovenia, Poland, Netherlands, Turkey. In Austria, the work was also funded by the program “ICT of the Future” and the Austrian Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation, and Technology (BMK). The document reflects only the authors’ views and the Commission is not responsible for any use that may be made of the information it contains. Special thanks to Michael Spitzer for developing the smartphone application and Ifikhar Ahmad for suggesting improvements to the computer vision-based application. More especially, this is a collaborative effort between four organisations, VIF (Austria), RISE (Sweden), Tietoevry (Sweden) and JIG (Spain) taking part in the InSecTT Project.

References

1. Smartwatch penetration rate worldwide from 2017 to 2026. <https://www.statista.com/forecasts/1314341/worldwide-penetration-rate-of-smartwatches>. Accessed 17 June 2022
2. State Farm Distracted Driver Detection overview. <https://www.kaggle.com/competitions/state-farm-distracted-driver-detection/overview>. Accessed 15 Nov 2023
3. Association, G.: *The Mobile Economy 2022* (2022)
4. Baheti, B., Gajre, S., Talbar, S.: Detection of distracted driver using convolutional neural network. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (2018)
5. Baheti, B., Gajre, S., Talbar, S.: Detection of distracted driver using convolutional neural network. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2018-June, pp. 1145–1151 (2018). <https://doi.org/10.1109/CVPRW.2018.00150>
6. Brodeur, M., Ruer, P., Léger, P.M., Senecal, S.: Smartwatches are more distracting than mobile phones while driving: results from an experimental study. *Accid. Anal. Prev.* **149**, 105846 (2021)
7. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807 (2017)
8. Chuah, S.H.W., Rauschnabel, P.A., Krey, N., Nguyen, B., Ramayah, T., Lade, S.: Wearable technologies: the role of usefulness and visibility in smartwatch adoption. *Comput. Hum. Behav.* **65**, 276–284 (2016)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255 (2009). <https://doi.org/10.1109/CVPR.2009.5206848>
10. Dua, I., Nambi, A.U., Jawahar, C.V., Padmanabhan, V.N.: Evaluation and visualization of driver inattention rating from facial features. *IEEE Trans. Biom. Behav. Identity Sci.* 1–1 (2019). <https://doi.org/10.1109/tbiom.2019.2962132>
11. Dua, I., Nambi, A.U., Jawahar, C.V., Padmanabhan, V.: AutoRate: How attentive is the driver? In: *Proceedings-14th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2019* (2019). <https://doi.org/10.1109/FG.2019.8756620>. <https://ieeexplore.ieee.org/abstract/document/8756620>
12. Dumitru, A.I., Girbacia, T., Boboc, R.G., Postelnicu, C.C., Mogan, G.L.: Effects of smartphone based advanced driver assistance system on distracted driving behavior: a simulator study. *Comput. Hum. Behav.* **83**, 1–7 (2018)
13. Eraqi, H.M., Abouelnaga, Y., Saad, M.H., Moustafa, M.N.: Driver distraction identification with an ensemble of convolutional neural networks. *J. Adv. Transp.* **2019** (2019). <https://doi.org/10.1155/2019/4125865>
14. Handel, P., Skog, I., Wahlstrom, J., Bonawiede, F., Welch, R., Ohlsson, J., Ohlsson, M.: Insurance telematics: opportunities and challenges with the smartphone solution. *IEEE Intell. Transp. Syst. Mag.* **6**(4), 57–70 (2014)

15. Hossain, M.U., Rahman, M.A., Islam, M.M., Akhter, A., Uddin, M.A., Paul, B.K.: Automatic driver distraction detection using deep convolutional neural networks. *Intell. Syst. Appl.* **14**, 200075 (2022). <https://doi.org/10.1016/j.iswa.2022.200075>. <https://www.sciencedirect.com/science/article/pii/S2667305322000163>
16. Janveja, I., Nambi, A., Bannur, S., Gupta, S., Padmanabhan, V.: InSight: monitoring the state of the driver in low-light using smartphones. *Proc. ACM Interact. Mobile Wearable Ubiquitous Technol.* **4**(3) (2020). <https://doi.org/10.1145/3411819>
17. Jiang, L., Lin, X., Liu, X., Bi, C., Xing, G.: Safedrive: detecting distracted driving behaviors using wrist-worn devices. *Proc. ACM Interact. Mobile Wearable Ubiquitous Technol.* **1**(4), 1–22 (2018)
18. Kaggle: State farm distracted driver detection (2016). <https://www.kaggle.com/c/state-farm-distracted-driver-detection>
19. Kaiser, C., Stocker, A., Festl, A., Djokic-Petrovic, M., Papatheocharous, E., Wallberg, A., Ezquerro, G., Orbe, J.O., Szilagy, T., Fellmann, M.: A vehicle telematics service for driving style detection: implementation and privacy challenges. In: VEHITS, pp. 29–36 (2020)
20. Kaiser, C., Stocker, A., Papatheocharous, E.: Distracted driver monitoring with smartphones: a preliminary literature review. In: 2021 29th Conference of Open Innovations Association (FRUCT), pp. 169–176. IEEE (2021)
21. Levi-Bliech, M., Kurtser, P., Pliskin, N., Fink, L.: Mobile apps and employee behavior: an empirical investigation of the implementation of a fleet-management app. *Int. J. Inf. Manag.* **49**, 355–365 (2019)
22. Luria, G., et al.: The mediating role of smartphone addiction on the relationship between personality and young drivers' smartphone use while driving. *Transp. Res. Part F Traffic Psychol. Behav.* **59**, 203–211 (2018)
23. Maier, C., Matke, J., Pflüger, K., Weitzel, T.: Smartphone use while driving: a fuzzy-set qualitative comparative analysis of personality profiles influencing frequent high-risk smartphone use while driving in Germany. *Int. J. Inf. Manag.* **55**, 102207 (2020)
24. Mantouka, E., Barmounakis, E., Vlahogianni, E., Golias, J.: Smartphone sensing for understanding driving behavior: current practice and challenges. *Int. J. Transp. Sci. Technol.* (2020)
25. NCAP, E.: What's new for 2020? (2020). <https://www.euroncap.com/en/vehicle-safety/safety-campaigns/2020-assisted-driving-tests/whats-new/>
26. Owens, J.M., Angell, L., Hankey, J.M., Foley, J., Ebe, K.: Creation of the naturalistic engagement in secondary tasks (nest) distracted driving dataset. *J. Saf. Res.* **54**, 33.e29–36 (2015). <https://doi.org/10.1016/j.jsr.2015.07.001>. <http://www.sciencedirect.com/science/article/pii/S002243751500050X>. (Strategic Highway Research Program (SHRP 2) and Special Issue: Fourth International Symposium on Naturalistic Driving Research)
27. Regan, M.A., Hallett, C.: Driver distraction: Definition, mechanisms, effects, and mitigation. In: *Handbook of Traffic Psychology*, pp. 275–286. Elsevier (2011)
28. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: inverted residuals and linear bottlenecks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018)
29. Schaap, N., van der horst, R., Arem, B., Brookhuis, K., Regan, M., Lee, J., Viktor, T.: The relationship between driver distraction and mental workload. In: Regan, M.A., Lee, J.D., Viktor, T.W. (eds.) *Driver Distraction and Inattention: Advances in Research and Countermeasures*, vol. 1, pp. 63–80 (2013)
30. Shahverdy, M., Fathy, M., Berangi, R., Sabokrou, M.: Driver behavior detection and classification using deep convolutional neural networks. *Expert Syst. Appl.* **149**, 113240 (2020). <https://doi.org/10.1016/j.eswa.2020.113240>. <https://www.sciencedirect.com/science/article/pii/S095741742030066X>
31. Shahverdy, M., Fathy, M., Berangi, R., Sabokrou, M.: Driver behaviour detection using 1d convolutional neural networks. *Electron. Lett.* **57**(3), 119–122 (2021)
32. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Bengio, Y., LeCun, Y. (eds.) *ICLR* (2015)

33. Streiffer, C., Raghavendra, R., Benson, T., Srivatsa, M.: Darnet: a deep learning solution for distracted driving detection. In: Proceedings of the 18th acm/ifiip/userenix Middleware Conference: Industrial Track, pp. 22–28 (2017)
34. Tan, M., Le, Q.: EfficientNet: rethinking model scaling for convolutional neural networks. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the 36th International Conference on Machine Learning, Proceedings of Machine Learning Research, vol. 97, pp. 6105–6114. PMLR (2019)
35. Wahlström, J., Skog, I., Händel, P.: Smartphone-based vehicle telematics: a ten-year anniversary. *IEEE Trans. Intell. Transp. Syst.* **18**(10), 2802–2825 (2017)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Working with AIoT Solutions in Embedded Software Applications. Recommendations, Guidelines, and Lessons Learned



Christina Gratorp

Abstract This chapter aims to be a broad introduction for embedded systems professionals that wish to add machine learning to traditional embedded software. It briefly describes the foundation for a stable and secure IoT communication platform, touching on important areas such as the MQTT protocol and data extraction. The discussion is based on a case study for a digitalized marine vessel, and focuses on guidelines and recommendations for how to work with machine learning models in industrial embedded software applications.

1 Introduction

Technical development in the field of artificial intelligence (“AI”) and machine learning (“ML”) have the potential to hugely impact the industrial sector. For embedded software running in control loops, on edge devices and through various machine-interfaces, machine learning models is becoming an increasingly popular addition. Knowledge of AI and ML will therefore grow in importance as traditional embedded solutions are supplemented by additional algorithmic handling of system data in order to improve for example reliability in service, optimization of features and prediction of maintenance. Working with ML in embedded systems requires knowledge of specialized software tools and frameworks developed for resource constrained environments.

To adapt to a changing industrial sector, Realtime Embedded AB (“RTE”) has been a long time partner in pan-European/ECSEL projects such as SCOTT, DEWI, PaPP and SMECY. The goal has been to participate in research activities in close proximity to industry-driven use cases. RTE is a consultant company that focuses on embedded systems and IoT solutions for the Swedish industry, with customers in lines of business ranging from life science and medical technology to the automotive industry.

C. Gratorp (✉)
Realtime Embedded AB, Stockholm, Sweden
e-mail: christina.gratorp@miljo.lth.se

To discuss ML in embedded software applications, this chapter takes as vantage point a three-part collaborative effort carried out within the InSecTT project, in which RTE serves as middle partner. The use case is centered on a digitalized marine vessel located in the port of Gdansk, and the use case's focus is to perform predictive maintenance for various on board systems. In short, the project includes digitalization and data logging (overseen by Gdansk University of Technology, GUT), algorithmic data management through ML modeling (performed by RTE) and visualization (implemented by Vemco). The focus of this chapter is the middle part: ML as developed as part of an embedded system that targets industrial business clients, with special emphasis on recommendations, guidelines and lessons learned. Still if the use case is specific, the discussion is—when applicable—broadened to a more general level.

The remainder of this chapter is organized as follows: the first sections covers overall project description and goals, and describes the project on an organizational level. The next sections discusses project design on a technical level, including both general and case specific aspects. The technical design section is followed by two short introductions to cloud verses edge computing and IoT security, and last the reader will find a short concluding discussion.

2 Project Description and Goals

The three-part collaborative project, from here on referred to as the Tucana project,¹ is based on vast sampling of operational data from a marine vessel. The onboard sensors are connected with the plug-and-play marine communications standard NMEA 2000. For RTE, the project aims to develop an algorithm that can detect behavioral discrepancies in order to avoid a complete malfunctioning in one or more of the onboard systems. This is commonly referred to as *predictive maintenance*. Given the economical implications for such diagnostic tools, predictive maintenance has grown to be a sought after application within the field of ML.

All in all, eleven sensors that report electrical battery voltages, course over ground, heading, geographical position, rate of turn, speed over ground, engine revolution and rudder angle are deployed on the vessel. For all sensors that collect operational data, see Table 1. The data is stored in a time series database and subscriptions to all data can be set up through a message broker, as shown in Fig. 1. Thus, historical data is fetched from the time series database and used as training data for the ML model, while during runtime, the ML model is invoked using real-time data samples received either through the broker or by requests from the time series database. After realtime analysis using ML, the result is translated into a corresponding status message and forwarded in accordance with a third party API. The API describes four possible messages that mirror vessel status. This status is integrated in a visualization software that shows vessel whereabouts and condition on a map of the port area. In addition to the forward signaling, a message describing a prediction of a possible

¹ Named after the marine vessel “Tucana”.

Table 1 List of sensors deployed on the Tucana vessel

Measurement
Battery 0 voltage
Battery 1 voltage
Battery 9 voltage
Course over Ground
Heading
Position (longitude and latitude)
Rate of Turn
Speed over Ground
Port revolutions
Starboard revolutions Rudder angle

need for maintenance can be sent back to the vessel personnel. This message will be sent only if such a result is the conclusion of the ML analysis. For full architectural overview, see Fig. 1. Please note that the concept “real-time” in this use case is used in a broader context than in traditional Real-Time computing scenarios. Due to the sampling frequency of the sensors and latency in connectivity, subscription data can be up to some seconds old, but will none the less be referred to as live or real-time data. Older data is referred to as historical data.

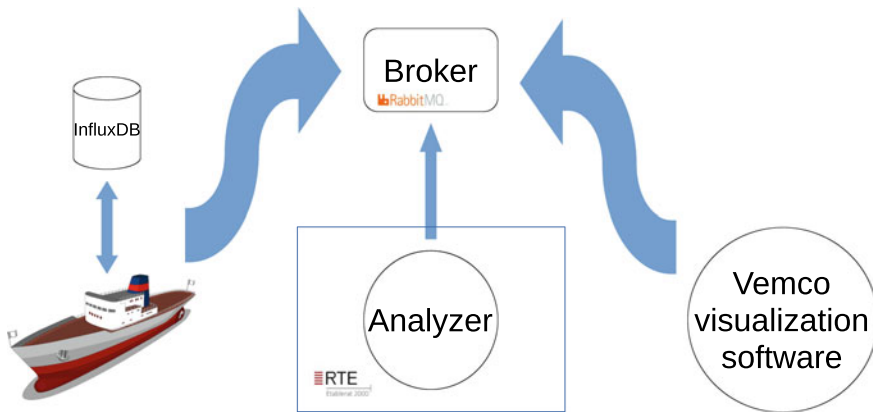


Fig. 1 Architectural overview

3 Project Design

This section covers the design of the software developed by RTE, that serves as middle part in the complete Tucana project setup. The software consists of a communication platform which includes an analyzer that integrates one or more ML models, and tools for creating a training data set. Additionally, a test environment to simulate the third party software is designed for verification purposes.² The design is described through its two phases. In stage 1, a simple ML model is used as a proof of concept in order to test the platform and data flow. In this stage, live data is fetched through queries to the time series database. In stage 2, the analyzer is generalized to be able to invoke different kinds of ML models and live data is retrieved by setting up subscriptions through a message broker. Both stages of the design use pre-trained ML models that does not learn beyond a defined moment in time. Alternatively, a learning model that is continuously retrained as new data become available could be used. This hypothetical scenario is discussed briefly in the section *Alternative model setup*. During stage 1. the complete RTE software is run on a regular PC with a Linux installation. However, since all software is written in Python, it is not platform dependent.

During design stage 2, the analyzer part can also be run on an edge device, in this case a Nitrogen6X board from Boundary Devices, with a 4 core, 1 GHz ARM Cortex-A9 CPU. For device environment, see Appendix A.

4 Machine Learning in Embedded Systems

Working with ML in embedded systems requires specific knowledge of software tools and frameworks developed for resource constrained environments. In this context, an embedded system is defined as a system dedicated to a specific set of functionality that has limited hardware resources, such as processor power or operating system capabilities (cf. Haigh et al., 2015). Embedded software is low-level software, tightly integrated with the specific hardware's design. These systems are often run through various predefined machine interfaces and therefore need to handle legacy. In the Tucana project this is reflected by the choice of InfluxDB and RabbitMQ message broker, that was made during the previous digitalizing part of the project, carried out by GUT.

Performing near real-time predictions in a live system affect choice of algorithms, communication protocols and overall software design. Embedded designs need to be reliable and include a sufficient amount of error handling, which extends to the parts that implement ML. Hence, the choice of tools, frameworks and standards chosen for the Tucana project have been made with embedded applications in mind, although the ML part itself is platform independent. In stage 1 of the project, the ML model is a sequential neural network, implemented by using the TensorFlow framework.

² Simulating a Vemco subscriber.

The model is saved as a protobuf (.db) file, which makes it easy to deploy on different hardware devices.

Stage 2 includes training a neural network in a cloud environment and export it as a TensorFlow Lite (.tflite) model. TensorFlow Lite is optimized for on-device machine learning and has multiple platform support, covering Android and iOS devices, embedded Linux and a range of microcontrollers. The second development stage also includes an exploratory approach, in which an unsupervised ML model is trained (clustering).

The edge device used in the second development stage is an iMX6 microcontroller, equipped with four ARM Cortex-9 CPU kernels. The software is written in Python, which provides easy access to libraries for socket communication as well as modules that are developed especially for ML purposes. For a list of enclosed software and versions, see Appendix A.

5 Communication Platform

5.1 Design Layout

The Tucana project's communication platform consists of four independently deployable processes in a client-server arrangement, see Fig. 2. The controller unit acts as server and handles all interprocess messaging. Three clients—an analyzer, an MQTT consumer and an InfluxDB client—can connect to the controller by using unique identification. The connection is a standard socket connection bound to a port number for the TCP layer to identify the application communication. The database client implements a periodic query to the Influx database, the MQTT consumer sets up message subscriptions to the broker and the analyzer handles system data processing and predictions.

Each client is run as a separate process that connects to the controller in a client-server socket connection setup. Using sockets allow clients and server to be run on different machines, should that be desired. In the Tucana project, all processes are run on the same machine. See Appendix B for sample code of a client controller class' connect and disconnect methods. In case a client is disconnected, the controller will keep serving remaining clients. If the controller shuts down, clients will try to reconnect until a new connection is established. Neither shut down of server nor clients should result in uncontrolled crashes. In the controller, each connected client is appended to a client list. The controller loops through all connected clients to handle incoming and outgoing messages.

Six inter-process messages are handled by the controller, see Table 2.

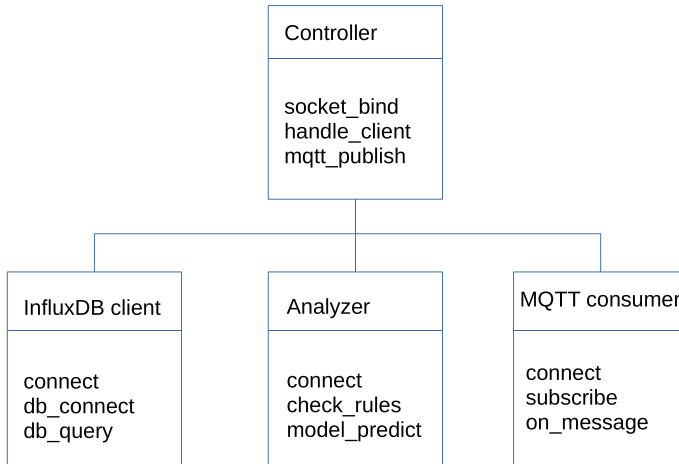


Fig. 2 Design overview

Table 2 Inter-process messages

Message name	Message no	Description
MSG_IDENTITY	1	Used by clients when connecting to controller
MSG_ANALYZE	2	Sent from controller to analyzer with live system data
MSG_ALARM_STATUS	3	Sent from analyzer to controller with analysis result
MSG_TUCANA_DATA	4	Sent from influx client to controller with live system data
MSG_MQTT_BODY	5	Sent from consumer to controller when MQTT message is received
MSG_TEST	100	Inter-process test message

5.2 Message Queuing with RabbitMQ

Message Queuing Telemetry Transport (MQTT) and Advanced Message Queuing Protocol (AMQP) are open-source protocols used for asynchronous messaging. Both are binary protocols and work on top of TCP/IP. They allow messages between applications irrespective of underlying software stack and is widely deployed for IoT services. MQTT is more light-weight and generally deployed in embedded systems, whereas AMQP is a more complete message protocol that is often used in larger systems and originates from the banking industry. For applications and small edge devices operating on minimum bandwidth, MQTT would likely be a preferable starting point, although security requirements, network reliability and scale might impact choice of protocol. Although MQTT and AMQP are common within IoT, they are still less commonly used in industrial control environments.

For the Tucana project at RTE, the choice to use RabbitMQ as message broker and AMQP over SSL is a prerequisite. It serves as a flexible solution that is both scalable and modular. The connectivity service is implemented as middleware and the design in large follows the strategy laid out by [4]. The design aims at separating protocol handling and control logic so that they can be independently deployed. For this purpose, we chose to implement the message queuing using Pika. Pika is a pure-Python client implementation of the AMQP 0-9-1 protocol for RabbitMQ. Several other Python libraries are similarly easy to use, for example Paho, however not as straight forward combined with RabbitMQ. Since RabbitMQ has an MQTT plugin that transparently translates between MQTT and AMQP, the Tucana project design depicts its message queuing client as MQTT.

The Pika implementation is made as simple as possible. It creates a channel connection and starts to consume messages from a queue dedicated to Tucana messaging. The `channel.basic_consume` method binds messages for a specific consumer tag to a callback. The consumer tag is automatically created should a specific tag not be declared. If there is a message with this tag in the Tucana queue, all subscribers will be notified. On message, the callback handles the AMQP message and forwards its payload in an inter-process message to the controller, as described in Figs. 3 and 4. For a list of all inter-process messages, see Table 2. In order to remove the AMQP message from the queue, the client finishes by sending a `channel.basic_ack`. For sample code of message queuing using the Pika client, see Appendix B.

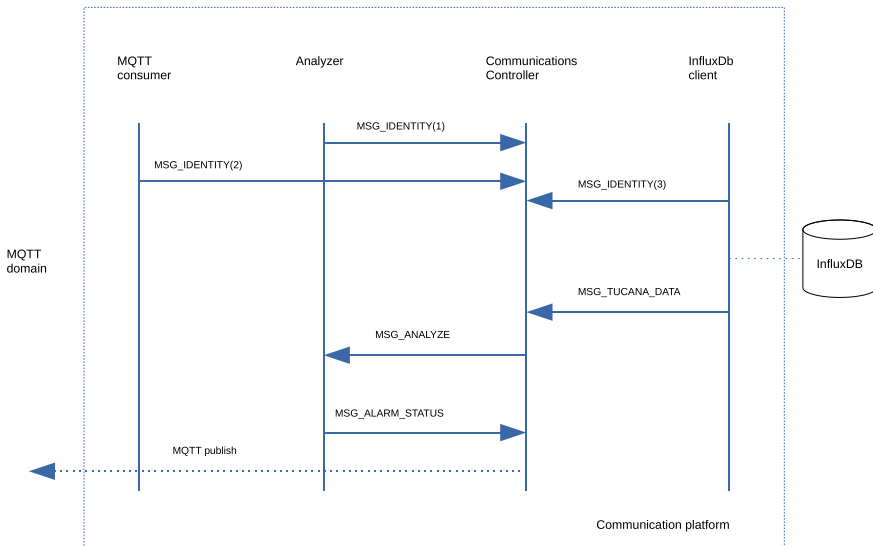


Fig. 3 Stage 1 system messaging

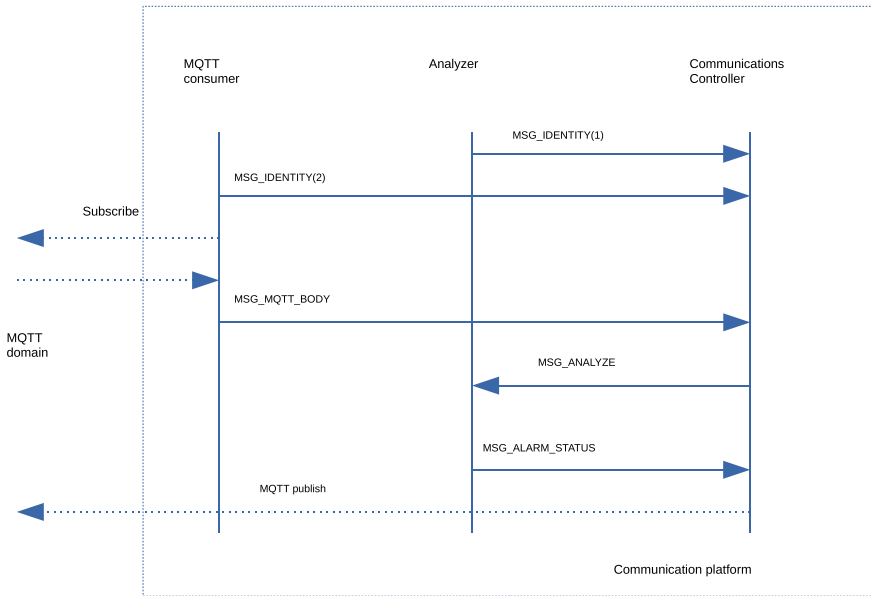


Fig. 4 Stage 2 system messaging

5.3 Inter-Process Messaging

In design stage 1, vessel data is fetched from the time series database with an interval of 20 s. In stage 2, subscriptions to live data is setup through the MQTT broker and the interval for performing an analysis is configurable. The data is routed to the analyzer through the controller. The analyzer always starts an analysis by checking that all sensor values pass basic rules and thresholds. After this initial sequence, it invokes the ML model in order to classify the present status of the marine vessel’s onboard systems. Should the model predict a state of needed maintenance, an MQTT status message is published to a dedicated queue. Any other software with the adequate credentials, within or without the Tucana project, can subscribe to such messages. MQTT publishing is handled directly by the controller. Please note that for a larger project, a dedicated MQTT publisher might be a preferable option. For an overview of all inter-process messaging, see Figs. 3 and 4.

6 Data Extraction

This section discusses data and data extraction. These are central topics to any embedded software project that aim to integrate ML methods. The rapid development of machine learning over the past decade and its applications within the industrial

sector brings forth new questions of data, data extraction and data representation connected to this field.

Three issues of general character quickly present themselves when dealing with data driven development: (1) what kind of data is accessible, (2) how is data accessed and (3) what kind of interpretations is that data open to? For the Tucana project, eleven sensors connected through the NMEA 2000 system are deployed on the marine vessel. The data is accessible via InfluxDB queries or as MQTT subscriptions, and the goal of the project is to interpret the sensor data to predict a need for onboard maintenance. Therefore, an ML model will have to recognize system anomalies. This means that the model has to be trained either by using labeled anomalies or by settings up model boundaries for how to interpret model output.

If the system in question is operating as a live system in an industrial environment, as is the case in the Tucana project, real anomaly data might be hard to come by. The vessel cannot be taken out of service for experimental purposes, and the onboard equipment is expensive. Putting high pressure on valuable industrial equipment to provoke errors that lead to anomaly data may be impossible. Instead, synthesizing anomaly data can be a way forward. To interpret and synthesize data accurately, thorough data exploration of historical data is necessary. Data to its nature is not solid, long lasting pieces of knowledge that makes universal sense, but should rather be looked at within the context of its creation. However contra intuitive a word like 'creation' might appear in this regard, it serves as a reminder of the complex process of interpreting data, as it captures the significant degree of human-machine interference that data extraction necessitates.

According to conventional wisdom around so called Big Data, more data is equated with a more accurate result. However, in embedded systems—as well as in general—relying on data analysis also requires a process of interpretation and translation in order to reveal “truth-telling” knowledge about a system. Sensor data and measurements might vary with temperature, air pressure or computational load, they might depend of geographical position, or they are perhaps only interpretable during a limited period of time. To address the question of the representative nature of data, even the term “data” itself has been questioned as a misnomer, as its etymological definition is that which is ‘given’ [5]. Instead, the term ‘capta’ has been suggested, meaning what is ‘taken’ [2]. This mirrors that which data to use is a choice.

Albeit the possible difference between the very fluid character of cultural data and technical measurements reported by for example digital sensors, keeping this discussion in mind has proven helpful during the development of the Tucana project. Here, system data is strictly bound to one specific vessel, rendering the statistical analysis closely tied to certain environmental circumstances. However, in order to make the platform more general, the Tucana project strives for a modular design, where the ML model can easily be swapped for another. The statistical analysis is also divided in two separate parts: a set of threshold rules that can be configured in accordance with the origin of the data, and a machine learning part that is trained on vessel specific data. Should the platform code be deployed for another vessel, the ML model most likely would have to be retrained.

Retraining for another vessel would, using the same model design and corresponding amount of training data, take approximately three to four hours on a standard laptop.

For this purpose, the project has developed a special toolchain for creating a training set database. For a more detailed description of this, see the next section. Depending on latency or restricted access issues, mirroring databases from customer site to development site can also constitute good practice to facilitate the development work.

7 Training Data Set and Model

7.1 Design Stage 1

This section describes a simple classifier model used for design stage 1, and the construction of a training set database. Using neural network based ML applications can be particularly challenging for embedded systems with limited resources. A number of methods exist to address these difficulties, including efficient model design, customized hardware accelerator designs and hardware/software co-design strategies [6].

In this proof of concept example, the model input vector is constituted of twelve data points, and the output is 1 or 0, meaning that the vessel is estimated to be located either inside or outside the port area. It is not an elaborated model, but it gives some information about the data itself and can tell if the vessel machinery behaves differently depending on its geographical location. However, in order to anticipate more detailed misbehavior, a more complex model must be deployed.

The training data build up is based on a so called “base measurement”. For the Tucana project, this is represented by the rate of turn measurement. Historical data is fetched from the time series database (InfluxDB), which stores measurements with individual timestamps. Therefore, data from different sensors will only in rare cases carry the same timestamp. Fetching the latest of all eleven measurements thus means the timestamps will differ depending on each measurement’s update frequency. Since rate of turn has the lowest update frequency, choosing this as base measurement means that the other measurements can be selected as “close” as possible to the rate of turn-values. To start extracting a training data set, a script tool reads entries from a base measurement file, see Fig. 5. As shown, twenty minutes worth of data is fetched for each entry. This results is one json file per entry, containing the stated time series for the rate of turn measurement. In this example this means that 12 json files containing time series for the rate of turn values will be created.

The script tool then continues by fetching the closest possible values for all other measurements. All measurements’ time series are stored in separate files. Thus, the final database will contain 11×12 files, or 12 chunks of data, each with a varying number of timestamp entries depending on the length of the respective time series.

```
[
  ["2022-04-20T10:00:00", "2022-04-20T10:20:00", "navigation.rateOfTurn", "value", "none", "port"],
  ["2022-04-19T09:00:00", "2022-04-19T09:20:00", "navigation.rateOfTurn", "value", "none", "port"],
  ["2022-03-22T11:30:00", "2022-03-22T11:50:00", "navigation.rateOfTurn", "value", "none", "port"],
  ["2022-04-09T09:35:00", "2022-04-09T09:55:00", "navigation.rateOfTurn", "value", "none", "port"],
  ["2022-04-06T08:20:00", "2022-04-06T08:40:00", "navigation.rateOfTurn", "value", "none", "port"],
  ["2022-04-11T15:30:00", "2022-04-11T15:50:00", "navigation.rateOfTurn", "value", "none", "port"],
  ["2022-04-12T10:20:00", "2022-04-12T10:40:00", "navigation.rateOfTurn", "value", "none", "fishing"],
  ["2022-04-21T08:50:00", "2022-04-21T09:10:00", "navigation.rateOfTurn", "value", "none", "fishing"],
  ["2022-04-11T13:20:00", "2022-04-11T13:40:00", "navigation.rateOfTurn", "value", "none", "fishing"],
  ["2022-03-22T10:20:00", "2022-03-22T10:40:00", "navigation.rateOfTurn", "value", "none", "fishing"],
  ["2022-04-06T08:41:00", "2022-04-06T09:01:00", "navigation.rateOfTurn", "value", "none", "fishing"],
  ["2022-04-13T08:10:00", "2022-04-13T08:30:00", "navigation.rateOfTurn", "value", "none", "fishing"]
]

rate_of_turn_1.json          rate_of_turn_2.json          . . .          rate_of_turn_12.json
battery_voltage_1.json     battery_voltage_2.json     . . .          battery_voltage_12.json
heading_1.json             heading_2.json             . . .          heading_12.json
.                           .                           .              .
.                           .                           .              .
.                           .                           .              .
```

Fig. 5 Base measurement json file (top) and resulting data chunks (bottom). The first and second column in the base measurement file specify between which timestamps the vessel has been located inside or outside the port area. The third column shows name of the measurement, the fourth and fifth specify what key(s) to look for in the database query and the last column sets the label. The label “fishing” is used as a metaphor for when the vessel is located outside the port area

When all data chunks are ready, the script tool finishes by reading all the data into an input matrix. The data values are normalized between 0 and 1, shuffled and split into one training set (70%) and one validation set (30%). The training set is used to train the model, and the validation set is used for evaluation. The model itself then splits the training set in one training suite and one test suite while fitting the model, see Fig. 6. In Fig. 7, validation of the neural network model is represented by true and false positives (tp, fp), true and false negatives (tn, fn), as well as precision (P) and recall (R):

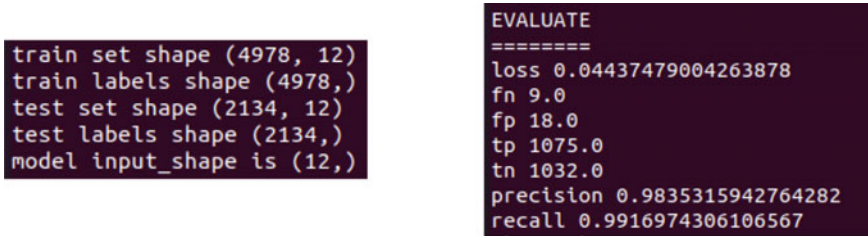
$$p = tp / (tp + fp) \quad R = tp / (tp + fn)$$

The neural network used in this example is a simple Tensorflow sequential model with 12 input nodes, one hidden layer of 6 nodes and one output layer with a sigmoid function, see Fig. 6. The reason that the number of input nodes are 12 and not 11 (as in the number of onboard sensors) is that the position sensor reports both latitude and longitude, which are represented by one node each.

```
model = tf.keras.Sequential([tf.keras.layers.InputLayer(input_shape=input_shape),
                             tf.keras.layers.Dense(6),
                             tf.keras.layers.Dense(1,activation="sigmoid")])

model.fit(x_train_norm, y_train, batch_size=128, epochs=100, verbose=1, validation_split=0.33)
```

Fig. 6 Sequential model. The model splits the training data set into training data (0.67%) and test data (0.33%)



```

train set shape (4978, 12)
train labels shape (4978,)
test set shape (2134, 12)
test labels shape (2134,)
model input_shape is (12,)

EVALUATE
=====
loss 0.04437479004263878
fn 9.0
fp 18.0
tp 1075.0
tn 1032.0
precision 0.9835315942764282
recall 0.9916974306106567

```

Fig. 7 Data dimensions (left) and evaluation metrics (right)

Using the base measurement as shown above, the training data, test data, label data and model will end up with the dimensions shown in Fig. 7. A batch size of 128 and 100 epochs evaluate the model with a precision value of 0.98 and recall of 0.99. These metrics clearly show that the ML model with a high degree of certainty can determine if the vessel is inside or outside the port area, which of course is not surprising given that the position data is included in the input vector. Excluding position data renders the output less accurate, with P approximately 0.83 and R approximately 0.85. Although these P and R values are not as good as when position data is included, the result is still useful to some extent.

7.2 *Design Stage 2*

In the second development stage, the idea was to use a clustering method to determine if any identifiable clusters would show up in the data. First, the plan was to use a clustering algorithm such as KMeans or DBSCAN. However, after the number of dimensions in the data was reduced from 12 to 2, two different areas for the data observations could be identified and used as clusters. After trying a number of other clustering methods, this method turned out to have the best outcome. The result is presented here in a series of visualized steps. The numbers of feature dimensions was reduced using Principal Component Analysis (PCA). Using two components proved to cover approximately 63% of the data variability, see Fig. 8, and we settled for this dimensional reduction. As shown in Fig. 9, two different data point areas appear in the 2D variability scatter plot: one “inner” area and one “outer” for label 0 and 1 respectively. Figure 10 shows a suggestion for how to determine if an observation should be classified as label 0 or 1.

7.3 *Alternative Model Setup*

As discussed above, the Tucana project design integrates an ML model that does not learn beyond a fixed point in time. This means that data reported after this

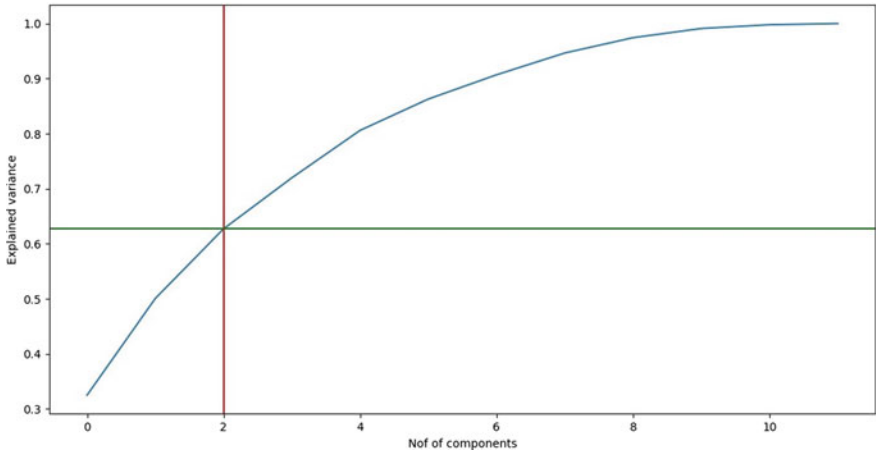


Fig. 8 Cumulative variance for the number of components. Two components cover 62.71% of the variability

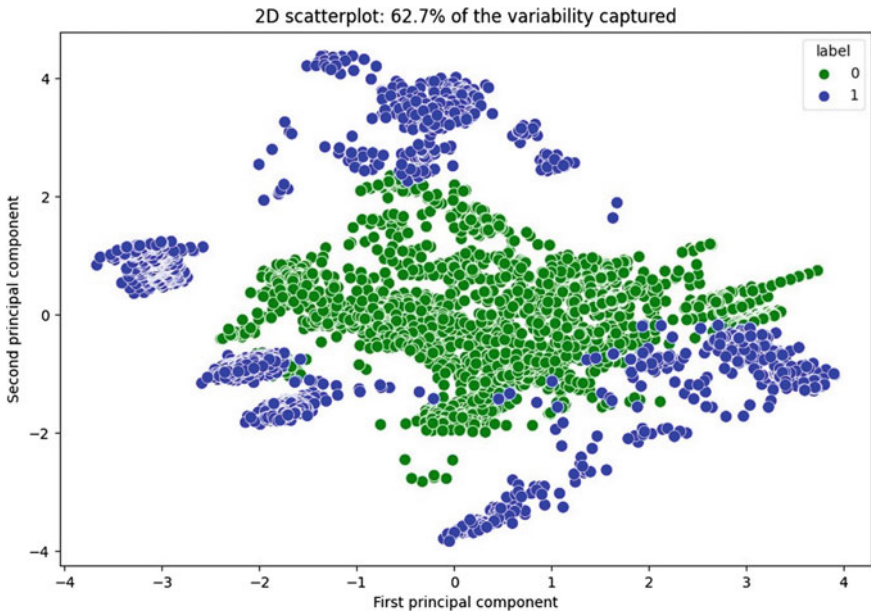


Fig. 9 Explained variance for two components. Label 0 corresponds to the vessel being inside the breakwater area, and 1 corresponds to the vessel being outside

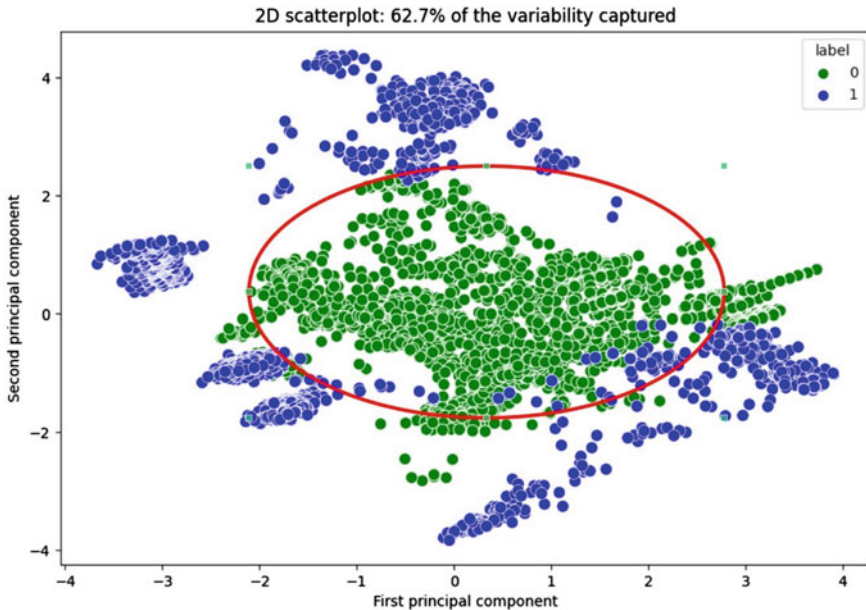


Fig. 10 Example of how an inner cluster can be separated from the outside area

moment is not included in the training data set, and the model, once deployed, is not retrained. As an alternative to this setup, a model that is continuously retrained can be used. Our suggestion for such an arrangement is to add another client process, refer to Fig. 2. This process should handle extraction and storing of new data as well as trigger retraining sessions of the model. New data chunks of course have to be labeled, why the process is preferably initiated on user command in combination with which label to use. This can be done in a number of ways: through a message subscription sent from a web application, from the command line interface or other. In case more training data become available, retraining the model can also be done automatically on a configurable interval, if the data is accurately labeled. For an example of a dynamically retrained framework architecture for embedded AI, see Fig. 11 (proposed by Brandalero et al.).

8 Cloud or Edge?

For a project such as the Tucana project, both cloud and edge computing can be a suitable choice. Cloud computing means that model training and data analysis is performed in the cloud, and edge computing means that these processes are handled on a small network device with less capacity. On one hand, system data is time-driven. Additionally, connectivity on the marine vessel could be limited, if the vessel

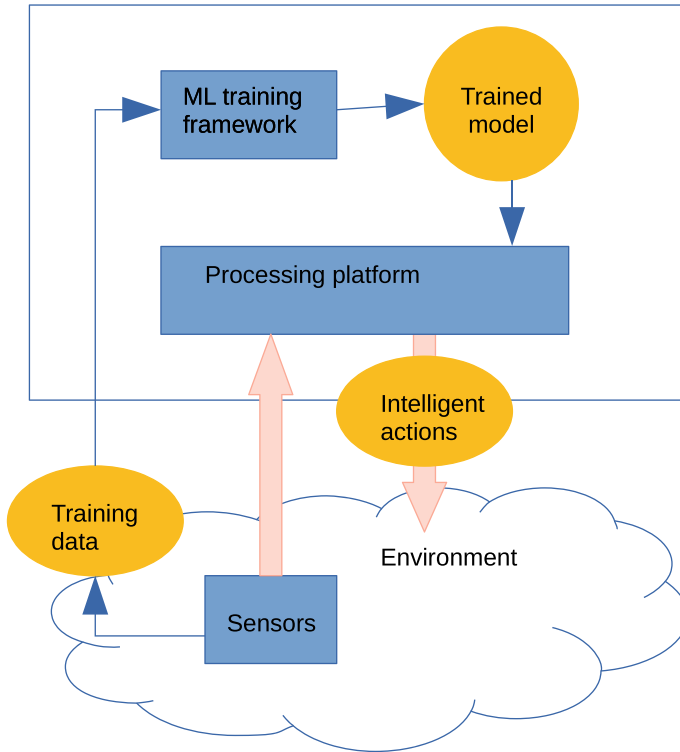


Fig. 11 AITIA Framework for Embedded AI as proposed by Brandalero et al. (2020)

is far from any cell tower. To avoid latency, these circumstances point towards edge computing (that the computing unit is located onboard, for example). On the other hand, the ML algorithm can potentially be modeled with large amounts of input data, and although the input data is time-driven, the predictions made by the ML model are not very time-sensitive. Maintenance work on a marine vessel will not start immediately, and decisions made by the crew or the shipping company will most likely not depend on the latest prediction. This does not mean that one single prediction could not make a difference, but decisions relying on model output will not have to be made within a very short period of time. Since the task of the ML algorithm is to *predict* a need for more acute maintenance, a likely scenario is that the vessel will undergo maintenance procedures by the end of the day, or when it returns to port. These circumstances allow for cloud computing. A mixed scenario with edge computing onboard and cloud computing when possible is also possible. Due to practical reasons during the development phase, the Tucana software is executed on a PC located far from the vessel. Whether the computational part will later on be deployed using edge or cloud computing is at the time of writing not decided. The ML model itself can be trained in a cloud environment, on a PC or on an edge device, as long as it is exported in a suitable format.

9 Security

This section serves as a very brief introduction to IoT security. The purpose is not to provide a detailed review, but to direct attention to this important matter. Many IoT devices are not designed with security in mind, but for industrial purposes security is often a central aspect. Thus, developing IoT systems that are scalable, error tolerant and easy to advance is not enough. An industrial online system also has to be secured from breaches and hostile take over. Therefore, the Tucana software is developed to fit a secure cloud based reference architecture developed by RTE during the SCOTT project (Secure Connected Trustable Things).³ For a general architecture overview, see Fig. 12. The secure architecture is based on a Google Cloud design. In many aspects, the three major cloud services Amazon Web, Google Cloud and Microsoft Azure are roughly equivalent, but easy access to API:s and an intuitive user interface resulted in the choice of Google Cloud. For the Tucana design, the IoT unit symbolizes the marine vessel and the Tucana software, as described in this document, would run on the gateway device. Note: A gateway can also act as an IoT unit, and there is no clear definition that tells them apart. For a smaller IoT unit, the communication link between the IoT units and the gateway can be realized with for example BLE (Bluetooth Low Energy), but for the Tucana design, the marine data is communicated using MQTT. Communication between the gateway and the cloud is setup using MQTT, and browser access is realized with HTTPS. All communication links are encrypted and communication between components in the architecture follows a pattern of authentication—to establish the sender’s identity—and authorization—to establish the senders right to access services supplied by the receiver. To handle this, each component is equipped with a public/private pair of keys. A dedicated server that stores security policies performs the authorization. Figure 13 shows an overview of different services that are integrated in the RTE secure platform. For further reading, please refer to online documentation for each component/service shown in Fig. 13.

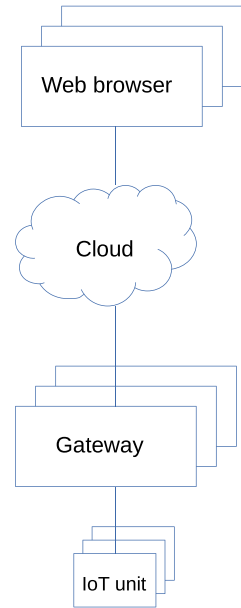
Important note: In August 2022, Google announced that Google Cloud’s IoT Core service would be discontinued within the year. For business customers and other professional users, this can potentially lead to big and time consuming migration projects. Thus, using cloud solutions that rely on services being maintained by other parties can cause extra work and should be part of a risk discussion with concerned clients and users. In this particular case, the discontinuation of IoT Core does not affect the Tucana project.

10 Conclusion

Working with ML in embedded systems require decisions that concern working with limited resources, security, real-time response requirements, high throughput-performance, cloud or edge computing, requirements of robustness, transparency and

³ <https://scottproject.eu/>.

Fig. 12 General reference architecture



so on. This chapter provides a number of short introductions to areas that are central to embedded systems that integrate AI solutions. We conclude that no ML algorithm will be more intelligent than data exploration and interpretation allow it to be, and hence we want to put extra emphasis on this domain. As the Tucana project proceeds, an important conclusion is also that ML can serve as a powerful tool during the development phase, but an ML model might not have to be included in the final product. As data exploration is a necessary part of all ML development, learning about the data itself, using for example ML technologies, can lead to other ways to integrate this knowledge in the final product. This can prove to be a more stable and robust software. Thus, ML can also serve as a feasible learning tool, and evaluation and validation of ML models will have to guide the development team along the way. Suggestions for further development in the Tucana project include for example completing a design that can combine cloud and edge computing depending on network availability, and implementing a dynamically retrained framework architecture for embedded AI.

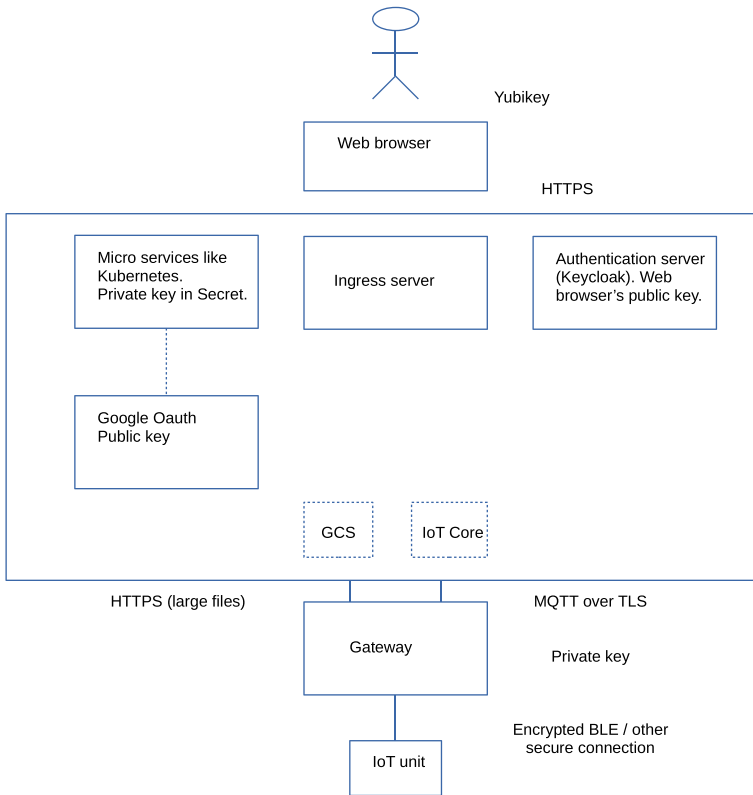


Fig. 13 RTE reference architecture for secure IoT

Appendix A

PC Environment

Keras 2.4.3

Pika 1.2.0

Python 3.8.10.

Tensorflow 2.5.0

Ubuntu 20.04.4 LTS.

Edge Device Environment

Linux image, supplied by Boundary Devices @ <https://boundarydevices.com/ubuntu-bionic-18-043-lts-for-i-mx6-7-boards-august-2019-kernel-4-14-x/>: 20,190,805-nitrogen-4.14.x_2.0.1_ga-bionicen_US-console_armhf.img.gz.

Uboot, supplied by Boundary Devices @ <https://boundarydevices.com/u-boot-v2018-07/>: uboot.nitrogen6q version 2018-07.

Appendix B

```
import socket
SERVER_PORT = 24230

class ControllerConnection(object):
    def __init__(self, sm):
        self.sm = sm // state
        machine, user defined
        self.wbuf = b'' // write buf
        self.rbuf = b'' // read buf

    def connect(self):
        print('Starting attempt to
connect to controllerd')
        s = None
        while s is None:
            s = socket.socket()
        try:
            s.connect(('localhost', SERVER_PORT))
        except:
            s.close()
        s = None

    time.sleep(1.0)

    print('Connected to controllerd\n')
    s.setblocking(False)
    s.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)
    self.sock = s

    def disconnect(self):
        self.sock.shutdown(socket.SHUT_WR)
        self.sock.close()
```

```

import pika
SERVER_PORT = 24230
MSG_MQTT_BODY = 5

amqps_URI = amqps://" amqp_authority [ "/" vhost
]parameters = pika.URLParameters(amqps_URI)
connection = pika.BlockingConnection(parameters)

def on_message(channel, method_frame, header_frame, body, userdata=None):
    if "Some string" in str(body):
        print(f"Received a message:
{str(body)}")
    else:
        ba = bytearray()
        ba +=
bytes([MSG_MQTT_BODY])
        ba += bytes(body)
        // forward mqtt message payload to controller

    # Send ack to remove msg from queue
    channel.basic_ack(delivery_tag=method_frame.delivery_tag)

channel = connection.channel()
channel.basic_consume('queue_name', on_message)
try:
    print("start
consuming")
    channel.start_consuming()
except KeyboardInterrupt:
    channel.stop_consuming()
connection.close()

```

References

1. Brandalero, M., Ali, M., Le Jeune, L., Hernandez, H.G.M., Veleski, M., da Silva, B., Lemeire, J., Van Beeck, K., Touhafi, A., Goedem, T., Mentens, N., Göhringer, D., Hübner, M.: International Conference on IEEE Language: English, Database: IEEE Xplore Digital Library 1–7 Aug, 2020 (2022)
2. Drucker, J.: Humanities approaches to graphical display. *Digit. Hum. Q.* **5**(1), 3 (2011). Accessed 10 Mar 2011
3. Haigh, Z.K., Mackay, A.M., Cook, M.R., Lin, L.G.: Machine Learning for Embedded Systems: A Case Study (2022). <https://www.cs.cmu.edu/afs/cs/user/khaigh/www/papers/2015HaighTechReport-Embedded.pdf>
4. Opačin, S., Rizvanović, L., Leander, B., Čaušević, A., Mubeen, S.: Developing and Evaluating MQTT Connectivity for an Industrial Controller. Submitted paper (2022)
5. Wasielewski, A.: Computational Formalism: Art History and Machine Learning. MIT Press, Cambridge (2023)
6. Zhang, X., Chen, Y., Hao, C., Huang, S., Li, Y., Chen, D.: Compilation and Optimizations for Efficient Machine Learning on Embedded Systems (2022). <https://arxiv.org/pdf/2206.03326.pdf>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Artificial Intelligence for Wireless Avionics Intra-Communications



Ramiro Samano Robles, R. Venkatesha Prasad, Ad Arts,
Mateusz Rzymowski, and Lukasz Kulas

Abstract This chapter presents a summary of the description and preliminary results of the use case related to the implementation of artificial intelligence tools in the emerging technology called wireless avionics intra-communications (WAICs). WAICs aims to replace some of the cable buses of modern aircraft. This replacement of infrastructure leads to: (1) complexity reduction of future airplanes, (2) creation of innovative services where wireless links are more flexible than wireline links, and mainly (3) a considerable weight reduction, which in turn leads to fuel consumption efficiency, increase of payload, as well as range extension. Therefore, WAICs is expected to have a large impact on the aeronautics industry, propelling a new generation of greener, more efficient, and less expensive aeronautical services. However, there are still several reliability, trust, interoperability and latency issues that need to be addressed before this technology becomes commercial. It is expected that AI will boost the applicability of this technology, contributing to the realization of the concept of “*fly-by-wireless*”.

R. S. Robles (✉)

Research Centre in Real-Time and Embedded Computing Systems, Polytechnic Institute of Porto,
Rua Alfredo Allen, 535, 4200-135 Porto, Portugal
e-mail: rasro@isep.ipp.pt

R. Venkatesha Prasad

Faculty of Engineering, Mathematics and Computer Science, Delft University of Technology,
Van Mourik Broekmanweg 6, 2628 XE, Delft, The Netherlands
e-mail: r.r.venkateshaprasad@tudelft.nl

A. Arts

NXP Semiconductors, High Tech Campus 46, 5656AE Eindhoven, The Netherlands
e-mail: ad.arts@nxp.com

M. Rzymowski · L. Kulas

Politechnika Gdańska, ul. Narutowicza 11/12, Gdańsk 80-233, Poland
e-mail: mateusz.rzymowski@pg.edu.pl

L. Kulas

e-mail: lukasz.kulas@eti.pg.gda.pl

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_18

Acronyms

Use the template *acronym.tex* together with the document class SVMono (monograph-type books) or SVMult (edited books) to style your list(s) of abbreviations or symbols.

Lists of abbreviations, symbols and the like are easily formatted with the help of the Springer Nature enhanced `description` environment.

AI	Artificial Intelligence
AFC	Active FLOW Control
AFDX	Avionics Full-Duplex Switched Ethernet
ARINC	Aeronautical Radio, Incorporated
ATM	Air Traffic Management
AWGN	Additive White Gaussian Noise
BL	Boundary Layer
CAN	Controller Area Network
DEWI	Dependable Embedded Wireless Infrastructure
DoA	Direction of Arrival
FDTD	Finite Difference Time Domain
IEEE	Institute of Electrical and Electronics Engineers
InSecTT	Intelligent Secure Trustable Things
IoT	Internet of Things
ISO	International Standards Organisation
ITU	International Telecommunications Union
MIMO	Multiple-Input Multiple-Output
ML	Machine Learning
PL	Power Line
SCOTT	Secure Connected Trustable Things
TTP	Time-Triggered Protocol
UMTS	Universal Mobile Telecommunication System
WAICs	Wireless Avionics Intra-Communications
WiFi	Wireless Fidelity
WSN	Wireless Sensor Networks

1 Introduction

The advent of the Internet-of-things (IoT) means that objects with embedded processing and networking capabilities will be exchanging information with cloud or edge infrastructure almost in real-time. It is expected that billions of devices will be connected to the cloud in the coming decades. Many of these connections are expected to be achieved by wireless links. In comparison with their wireline counterparts, wireless channels constitute much harsher and random propagation media. However, wireless technology provides several advantages, such as flexible deployment,

reduced implementation times, mobility, and the ability to cover locations difficult to reach with cables [1]. In addition, it enhances end-user experience improving automatic commissioning, management, and configuration. Aeronautics provides a framework where wireless and wireline will experience a complex coexistence. Wireless technology is becoming more competitive, and new releases are quickly catching up with wireline technology. But wireline technology evolves too with more efficient power-line communication and higher levels of cross-talk reduction technology. Therefore, it is expected that the decision to use either cables or wireless links will become a complex task due to the multiple factors and criteria needed by future applications.

In avionics, wireless technology is well known for air traffic management (ATM), telemetry, aircraft-ground control, satellite communication/localization, inter-aircraft communications and radar. By contrast, intra-avionics communications have just recently gained attention. Conventionally, the main concerns with the use of wireless links on aircraft are related to reliability/safety critical operations, security, and interference to on-board systems. However, recent studies suggest that commercial technologies show low impact to on-board systems. This has paved the way for interesting wireless applications for aircraft, which have been called wireless avionics intra-communications (WAICs).

One major potential advantage of using wireless technology in aeronautics is the reduction of cables [2]. According to [3], helicopters carry almost 2,000 pounds of wires. It is estimated that the use of wireless will bring a 12% reduction in terms of fuel consumption [4]. Cabling tasks cost nearly 2,200 dollars per kg of aircraft [5]. Estimated savings can reach 14–60 millions of dollars per aircraft [5]. Electrical wiring problems cause on average two in-flight fires every month as well as thousands of mission aborts and lost mission hours per year [6]. Each year, one to two million man-hours are spent by navy in finding and fixing wiring problems [7]. Damages on cables can affect not only the system related to the faulty wire, but also contiguous systems [8]. It is also estimated that 13% of an aircraft operation cost is related to maintenance, reparation and overhaul. Wireless nodes can also cover locations not easily reached by cables. Therefore, wireless is expected to bring considerable gains to the industry in terms of reduction of wires, flexible designs, and improved troubleshooting. This has been called “fly-by-wireless” [9].

An attractive use of wireless technology in aeronautics is sensing. Recent advances have provided a reliable, cost-effective, self-organized, ad-hoc, and flexible networking technologies [10, 11]. Furthermore, WSN technology provides with self-configuration, RF tolerance, and maintenance troubleshooting [8]. In critical applications though, wireless links cannot completely replace cables due to the high reliability requirement. However, they can act as redundant links, thus increasing reliability and flexible design.

Despite recent advances in the feasibility testing of this technology, WAICs is not yet commercial in the aeronautics industry. Multiple tests have been done proving that wireless can replace successfully cables and even the full buses of different types of aircraft in flight conditions too. However, the aeronautics industry has also very high standards, multiple testing and certification steps for new technologies to be

approved and eventually become commercial, particularly if these new technologies pose a threat or open vulnerabilities in the operation of an airplane. The development of this technology has great potentials but also major obstacles, including trust issues of technology selection/decision makers, etc.

The objective of the use case described in this chapter is to use artificial intelligence (AI) algorithms for the improvement of reliability, dependability and criticality of the new technology called wireless avionics intra-communications [12–19]. The improvement foreseen by the development of this use case lies in the lower layers of the communication stack, focusing also on the interaction with the internal aeronautics wireline network based on a deterministic and real time technology. The AI algorithms envisaged for WAICs aim to overcome the main degrading effects onboard an aircraft, mainly interference, fading, shadowing, metallic reflections, turbulence, etc. Several of the intended AI algorithms will use multiple antenna diversity which requires new modelling, prediction, and analysis in a different environment such as on board an operational aircraft.

Summarizing, the objectives of this use case are focused on overall improvement and development of WAICs by:

1. Improvement of reliability of WAIC to the same or similar level as the wired safety critical avionic networks of commercial aircraft.
2. Introduction of adaptive transmission to WAIC by means of artificial intelligence (AI) using spatial diversity and other signal processing algorithms for WAIC to reduce impact of potential interference, avoid jamming, increase spectral efficiency and support low latency.
3. Prediction of channel conditions in WAIC. AI will aim to predict channel conditions of an aircraft during different moments of a mission, including turbulent conditions and detecting changing patterns according to the type of aircraft and the movement of passengers inside the cabin.
4. Increase of Technology Readiness Level of WAICs by using a prototype demonstration in a representative scenario inside an aircraft.
5. Improvement of connectivity in an aircraft by employment of reconfigurable IoT antennas.
6. Increase explainability of AI decisions in the context of WAICs.

The organization of this chapter is as follows. Section 2 presents the objectives of the use case. Section 3 provide link to the building blocks of the project that are implemented in this use case. Section 4 provides a review of the state of the art. Section 5 provides the added value of AI and IoT. Section 6 presents the scenarios of the use case, while the remaining sections present details of the different building blocks of this use case and preliminary results.

2 Use Case Objectives

This use case addresses mainly the objectives of the use of AI algorithms for the improvement of wireless infrastructure by two means: an intrinsic improvement of

wireless technology, and the second way an improvement using the information transmitted by wireless technologies. The majority of the developments of this use case lie in the first class. The reason behind these objectives is that in the lower layers we find the main trust issue that currently affects WAICs to replace cable structures on board modern aircraft. The challenge is to show that WAICs are trustworthy for critical industrial applications in this domain. The developments also cover the issue to increase dependability and match the real-time behaviour of the critical internal aeronautical networks of operational aircraft. This means to make wireless avionics links as wireline-like as possible to reduce the mismatch between the two types of networks. This implies to minimize to a great extent the issues of interference, fading, shadowing, multi-path degradation, etc.

3 Link Between Scenarios and Building Blocks

The use case of AI for WAICs is constructed on a set of technical building blocks developed by different partners and which are integrated in the final use case demonstration. The philosophy of the project InSecTT lies on the concept of cross-domain reusability of building blocks. This means that many of the building blocks that constitute this use case can be reused in other use cases or applications with convenient adaptations. The mapping of building blocks to the scenarios of this use case is given in Table 1.

4 State of the Art

A WAIC system can be defined as a wireless transmission system where the network devices or nodes are located in the same aircraft [20]. Note that this definition explicitly excludes aircraft-to-ground or aircraft-to-aircraft communications. However, these systems must not be ignored in the design of a WAIC network, as interference can arise if contiguous/similar frequency bands are used, and also because some WAICs applications must interact with these other systems. A WAIC system can thus have very diverse purposes, such as [21]: wireless sensing [22, 23], cable replacement [24], structural health monitoring [25–27], remote control and maintenance [28], object identification [29, 30], fuel tank level monitoring [23], actuation (flaps), surveying, and avionics bus communications [31–33], etc.

Note that the term WAIC also applies to other types of aircraft such as smaller or larger aircraft, helicopters [34], aerial drones, etc. A good WAIC design must consider the optimum number and location of nodes, access points, or signal repeaters/relays, which ensure appropriate coverage where service is to be provided.

The design of a WAIC system must consider its integration into the existing communication infrastructure on aircraft. Avionics Full-Duplex Switched Ethernet (AFDX) is a data network for safety-critical applications that uses dedicated band-

Table 1 InSecTT sub-building blocks mapping for WAICS use case

TBB	Scenario 1	Scenario 2	Scenario 3
TB2.1			AI for multi-parameter sensing
TB2.2	AI for interference detection, mitigation, channel est., equalization, and prediction Interference detection and mitigation	AI model implementation and validation Multiple metrics assessment	AI for optimized sensing across aircraft
TB2.3			Sensor dynamics model
TB2.4	Metrics for V&V of AI tools for wireless systems	Metrics for V&V tools for wireless systems. Test generation, jamming attacks generation. Explainability of digital signal classification by an AI neural network	AI for multi-parameter sensing
TB2.5	Trustworthiness metrics for WAICs architectures	Trustworthiness metrics for WAICs architectures	
TB3.2	MIMO modelling, interference and propagation modelling	AI for interference detection, suppression channel estimation equalization and conflict resolution	AI for sensor management
TB3.4	Latency evaluation WAICs	Latency evaluation WAICs	
TB3.5	Security testing System level simulation. Multiple approach	Security testing System level simulation. Multiple approach	Security testing System level simulation. Multiple approach

width while providing deterministic Quality of Service (QoS) [35]. AFDX is based on IEEE 802.3 [36] Ethernet technology. Other bus technologies in aeronautics are CAN (Controller-Area-Network) [37], TTP (Time Triggered Protocol) and wavelength division multiplexing over optical fibre. The work in [31, 32] proposed a hybrid network based on the standard 802.11e (which is the flavour of WiFi technology with support for QoS) with the internal AFDX Ethernet network which is standard in modern passenger aircraft.

Standardization for WAICs has been materialized over the last few years. The initial ITU recommendation in [21] provided definitions and service desice recommendations in the ISM free frequency bands. However, subsequent recommendations and standardization have defined a specific band allocation and more details of physical layer design [12–18]. This PHY-layer of WAICs is particularly challenging because the physical configuration of an aircraft changes depending on the aircraft model and manufacturer. Outside the aircraft, wireless transmissions can suffer from reflections from the fuselage or moving metallic parts. While the aircraft is flying,

Table 2 WAICs scenario 1

Use case	AI-enriched wireless avionics resource management and secure/safe operation
Scenario	Interference detection and cancellation
Description	Use of signal processing tools for multiple antennas to detect intentional and non-intentional jamming interference on board aircraft that may harm operation of the emerging WAICs technology mitigation, channel est., & and validation across aircraft
Trigger	Designer of WAIC system running the test scenario
Flow of events	<p>Flow</p> <ol style="list-style-type: none"> 1. Entity model implemented in a given context with distances and obstacles, attackers, relevant to propagation and performance of the network 2. Propagation and interference models used to calculate link layer metrics 3. Link layer metrics are modified to account for advanced PHY layer and also to feed AI algorithms. Attacks are also actively generated 4. Decision on resource allocation and interference mitigation based on MIMO and other like adaptivity measures is implemented 5. Metrics of performance are collected over a number of simulation or demonstrator runs also feeding learning algorithms that will reuse the results in future runs <p>Normal flow</p> <ol style="list-style-type: none"> 1. Sensors nodes in the cabin of the aircraft send measured data via wireless network 2. Gateway nodes receive data and measure signals parameters 3. Direction of arrival of the signal and possible spatial information are calculated by the system <p>Exceptional flow</p> <ol style="list-style-type: none"> 1. Sensors nodes in the cabin of the aircraft send measured data via wireless networks and obstacles, attackers, relevant to propagation and performance of the network 2. A jamming signal is introduced in the cabin 3. Gateways try to receive data from sensors. Jamming signal is detected (but not yet classified) and measured 4. Direction of arrival of the signal and possible spatial information are calculated by the system 5. Alert is raised, as the system detects an interference anomaly
Input data	Physical layer multiple distributed antennas across the aircraftq
Output data	Identification of angle of arrival (AoA) of potential interferer or MIMO subspace definition. Interference compensation MIMO subspace or beamforming arrays that minimize distortion metrics
Infrastructure	PHY-layer simulation, system-level simulation and validation tools
TBB	BB2.2, BB3.2, BB3.4, BB3.5

Table 3 Scenario 2

Use case	AI-enriched wireless avionics resource management and secure/safe operation
Scenario	Verification and validation of WAICs
Actors	Transmitters, receivers and jammers in wireless communication, testing framework and the designer of WAIC system
Description	The goal of this scenario is to create a wireless testing framework capable of: Simulating wireless communication channel Simulating additional interferences (both intentional and unintentional) in the channel Applying real measured signals to the simulation Applying a real measured channel to the simulation Connecting real nodes to the simulated channel/interferences Handling multiple by standing devices in the simulation Overall, the framework should be able to simulate possible real-life situations, with capability to connect actual hardware mounted in avionics systems
Trigger	Designer of WAIC system running the test scenario
Flow of events	Flow 1. The WAIC system designer creates a test scenario 2. The WAIC system designer runs the test scenario 3. Channel is created, transmitter, receiver and jamming device are connect to the framework 4. Measurements are running 5. Test results are presented on UI
Input data	Signals from transmitter, signals from jammer, channel configuration
Output data	Measurements of received signal
Infrastructure	Depending on the scenario: receiver, transmitter, jammer, testing framework deployed on a server or a cloud
TBB	TBB 3.5: Verification, validation, accountability

turbulence conditions, high altitude, extreme temperature, vibration, etc. might also change propagation conditions. Inside aircraft, there are several challenges too. For example, passengers and crew moving constantly around cockpit and bays can modify propagation conditions; luggage with different materials could cause absorption or further reflection; windows can be transparent to radiation, thus being a source radiation loss and interference to outside networks or other planes; inside the aircraft radiation can cause interference to navigation and critical aircraft systems, etc. Multi-path propagation is likely to be increased too, mainly due to reflections on metals or signals travelling through different materials. Interference between contiguous systems can also be an issue in the future of WAICs.

The works in [38, 39] propose an electromagnetic modelling of cabins for WiFi, UMTS and Bluetooth technologies. The European project WirelessCabin [40–43] has addressed the measurement and propagation modelling of aircraft focusing on

Table 4 Scenario 3

Use case	AI-enriched wireless avionics resource management and secure/safe operation
Scenario	Devices, sensors inside the aircraft and sensors outside
Description	Building the devices that can harvest energy and also sense some information from passengers/systems Example, wind speed, push of a button, etc.
Trigger	The change in physical parameters
Flow of events	Flow 1. Design of devices 2. Testing them for the availability of energy 3. Increasing the reliability
Input data	Physical layer multiple distributed antennas across the aircraft
Output data	Identification of physical parameters or user driven information Wind speed information
Infrastructure	Embedded System/device
TBB	BB2.1 BB2.3 BB2.2, BB3.2

path-loss modelling, multi-path delay profile characterization, and frequency selectiveness. The work in [44] provided an electromagnetic propagation study based on FDTD (Finite-Difference Time-Domain) for the optimization of the number and location of access points inside the cabin of an aircraft. Multiple works exist on channel modelling for aircraft, inside the cabin or outside the cabin, or targeting interference to internal instruments and/or targeting the effects of passengers or multiple materials on propagation factors.

Testing of existing standards such as ZibBee, IEEE802-25-4 in all the different flavours, have been presented in different works, measuring different aspects from reliability, latency, emissions, and robustness to attacks (Security). The authors in [45] provide an extensive analysis of different types of security attacks using an adversary model, where the adversary can be internal or external and the attack can be passive or active. Security features of COTS technologies for wireless avionics have been also addressed in [46]. Other security and privacy considerations for wireless avionics communications can be found in [45, 47–49].

5 AI/IoT Added Value

AI is expected to bring several benefits to WAICS technology. In particular, the learning phases provided by AI can help us use data sets from different scenarios to refine the processing of signals in future or in current instances of the scenarios. For example, for interference, it can help us identify clearly multiple spatial signatures

that match previous issues or events included in our data sets. AI can be applied in this use-case at various points:

1. To improve early recognition of digital signals
2. To improve antenna beam-forming

In the context of avionics, AI will be mostly deployed to assist in solving complex RF transmission problems, where the application of AI will result in a better overall outcome (e.g. use of the frequency spectrum, cope with electronic RF disturbances) as traditional algorithms will. Due to the complexity of the problems addressed, clarification of the AI decision/classification process is desired by the experts using the system. Within the scenarios, various attempts shall be made to visualize the AI classification decisions for system experts. AI will be used to improve the reliability, latency and criticality (real-time support) of the wireless avionics' layers. The aim is to make the wireless links onboard the aircraft as wireline-like as possible in terms of performance. This is, the wireless network is expected to behave like the internal wireline network of the aircraft which is a real time technology. More specifically, the real time virtual links of the aeronautical bus standards are expected to be mapped transparently into the wireless side of the network. This involves a huge improvement in reliability and real time performance. AI is expected to help WAICs achieve this challenging ambition.

6 Scenarios

This section deals with the scenarios proposed by this use case to provide an instance example of how AI will improve the operation of WAICs.

6.1 Scenario 1: Interference Detection and Cancellation

One of the most challenging aspects of the technology WAICs to be highly reliable and trustable is the relatively small area of deployment and the complexity/density of all critical subsystems of an operational aircraft. This can be worsened by the highly unpredictable propagation channels and the extreme environmental conditions that can be found at different moments of a flight mission: take-off, taxi, landing, etc. Perhaps the most important issue on board an aircraft is the potential interference towards the WAICs network (coming from different directions internal or external to the aircraft) or the interference from the WAICs system to other on-board equipment. This scenario is focused on the modelling, measurement, detection and rejection/compensation of several sources of non-intentional or intentional (Jamming) interference on board the aircraft considering different positions of the interfering nodes or different strategies of the jamming attackers (see Fig. 1). Artificial intelligence tools will be designed to improve all the detection and compensation

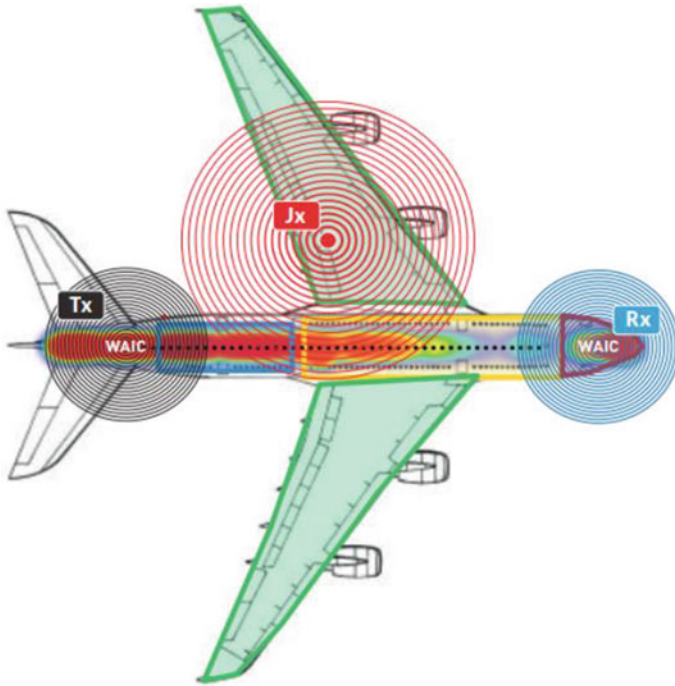


Fig. 1 Interference characterization for WAICs using AI

mechanisms and thus boost the reliability of WAICs systems under different aircraft conditions. More details of scenario 1 can be found in Table 2.

6.2 Scenario 2: Verification and Validation of WAICs

As wireless communication starts playing an important role in avionic intra-communication, a reliable testing environment becomes a necessity. The main objective of this scenario is to provide a framework for verification and validation of WAICs (Table 3).

6.3 Scenario 3: Battery-Less Devices

There is a need to reduce the wires and collect input/context information inside and outside of the aircraft. An initial passive switch developed in the project SCOTT is improved here in InSecTT to optimize communication performance, extend battery



Fig. 2 Example of a passive switches application

life and reduce interference on board aircraft (see Fig. 2). More details of scenario 3 are given in Table 3. The passive switch consists of a piezoelectric material that can produce a pulse of electromagnetic energy with a single deformation or pressing the switch by hand. This provides formidable communication capabilities without any battery support or extended battery life.

7 Performance Evaluation

This Section presents the details of the evaluation of some aspects of the scenarios described in previous sections. The core of the performance evaluation verification and validation of WAICs in this use case is the development of a system level simulator that contains all the details of synthetic and realistic data sets for channel power and interference location.

The process of verification, validation is conducted using a system level evaluation tool with an architecture shown in Fig. 3. The system level simulation can be defined as the detailed accurate representation of the environment where the system will operate. The simulation must include all the relevant processes either deterministic or random that play role in the performance on the system. In a wireless network, the channel model is a crucial component to recreate the environment where signals will propagate. In our case, the system level simulator considers a realistic virtual model of an aircraft and a synthetic channel geometry-base stochastic model. The details of the physical layer processes to be included in the system level simulator depends on the target and objective of the simulation. In our case, we intend to evaluate if wireless transmission assisted by artificial intelligence is viable up to the high standards of the industry in a difficult environment such as an operational aircraft (See symbol error rate performance results in Fig. 4).

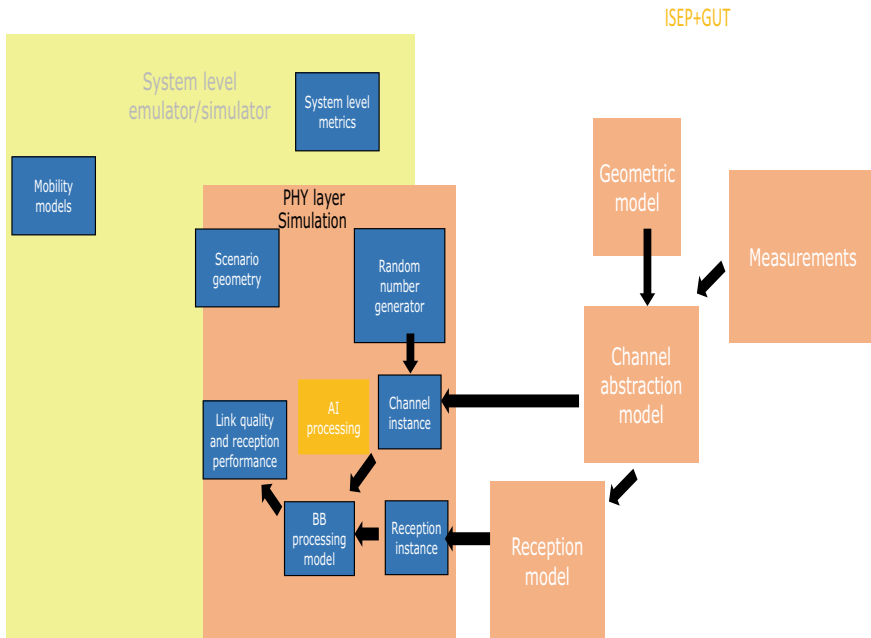


Fig. 3 Architecture of the system level simulation tool for the process of verification and validation of WAICs

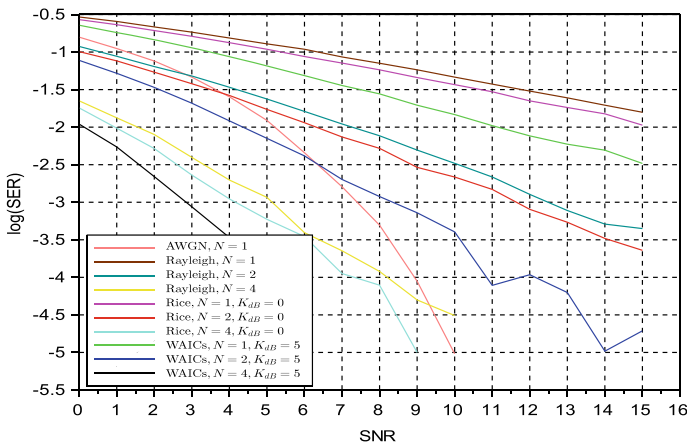


Fig. 4 SER (Symbol Error Rate) for different wireless links with different numbers of antennas N

7.1 Propagation Channel Modelling

A detailed channel model with multiple ray tracing and stochastic scatterers has been proposed for aircraft propagation (Table 4). The main idea is to calculate the

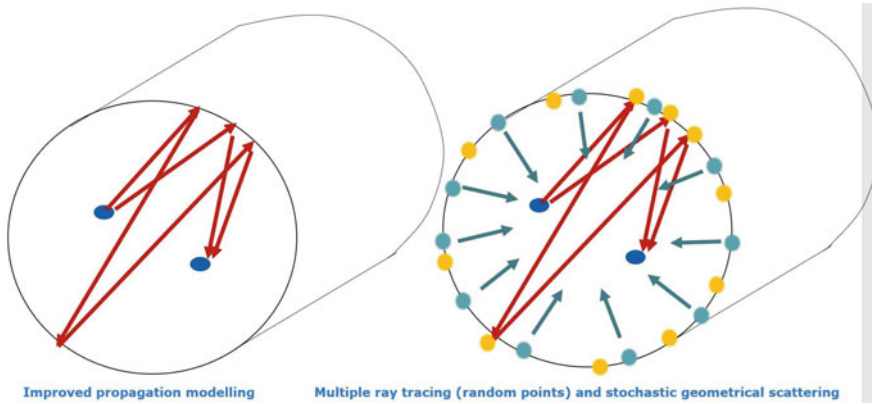


Fig. 5 Cylinder geometry for in-cabin channel modelling

reflections (deterministic) of multiple orders that occur inside a hollow structure of the aircraft. Different models have been used including a simple cylinder, followed by a cylinder and a solid floor inside as shown in Fig. 5. The channel model has also evolved to considered seats and passengers inside the aircraft (see Fig. 6). More recently, the channel model has considered two different profiles revolving around the central axes that define both the transversal section of the aircraft and its variations along the length of the aircraft (see Fig. 7). This modification allows us to consider any type of aircraft model and obtain a realistic virtual model of the internal mechanical parts of the aircraft.

Additionally, to address the evaluation of the coupling between internal and external networks, the channel models allows us to calculate the distribution over the windows of the aircraft. The electric fields can thus be used to calculate the radiation towards different directions in the external side of the aircraft, including the wings.

7.2 MIMO in Aeronautics

The implementation of MIMO in aeronautic commercial networks is in its infancy. For purposes of system level simulation the antenna elements will be considered as closely located and with the same set of scatterers impinging a random signal over the arrays. This leads to a natural correlation between the elements. However, our system level simulator has the option to consider additional electromagnetic coupling that naturally arises between radiating or passive elements located at close distances.

The simulator has the option to consider different processing schemes for MIMO, also enabled by AI. The advantage of MIMO is that due to spatial diversity, solutions

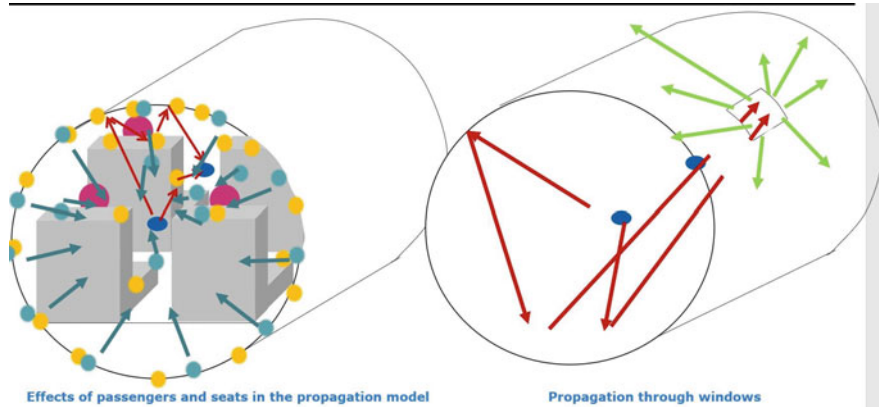


Fig. 6 Cylinder geometry for channel modelling including seats, passengers and radiation through windows

Fig. 7 Example of an aircraft profile for channel modelling



such as beam-forming and multiple antenna combining or precoding can help to reduce interference and fading, common issues in wireless networks.

The simulator has included MIMO beam-forming and direction of arrival estimation based on AI algorithms. The use of these two components is crucial to detect, track and counteract sources of jamming interference wither on board or outside the aircraft.

7.3 Wireless Measurements

A number of measurement campaigns of signal propagating inside a real aircraft cockpit have been conducted. The measurements consider positions of sensors across different locations of the aircraft as shown in Fig. 8. These measurements create a realistic data set to be used by AI algorithms. However, the data set does not cover all the potential positions and effects inside the aircraft. These measurements are therefore incorporated into the synthetic channel model presented in previous subsections. The parameters of the synthetic model such as Rice factor, profile decay factor, and multi-path distribution are adjusted according to the available measurements.

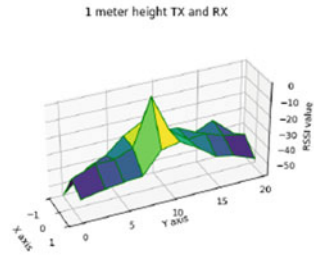
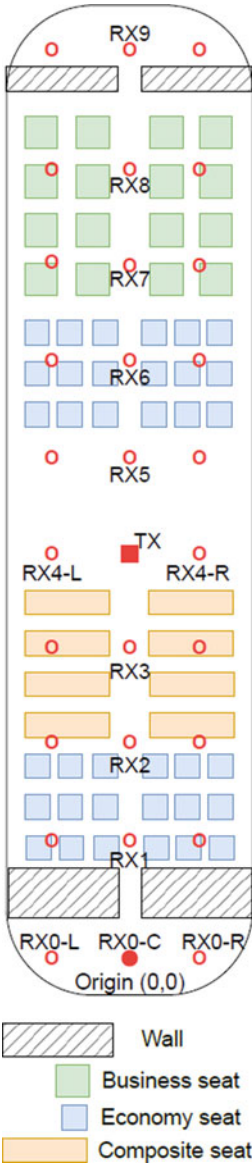


Fig. 2. 1 meter height TX and RX

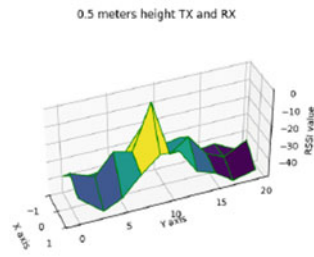


Fig. 3. 0.5 meters height TX and RX

Differential RSSI between measurement at 0.5m and 1m

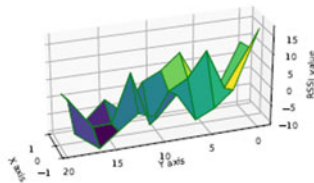


Fig. 8 Results of measurements campaign inside a Boeing aircraft

7.4 *Single and Multiple Link Results*

We consider a single wire-line link modelled as a channel with additive White Gaussian noise whose performance depends exclusively on the signal-to-noise ratio (SNR). The performance symbol error rate of a BPSK modulation is well known in the literature and shown in Fig. 4.

When a wireless channel model based on Rayleigh distribution is used, the performance is considerably degraded compared to the AWGN or cable performance. This is one of the main reasons behind the lack of reliability of wireless links compared to wire-line solutions.

The results using MIMO with different numbers of antennas, denoted by N show how the spatial diversity helps in reducing the effects of fading channels and get close to the performance of a wire-line system, which in this case is represented by the AWGN solution. The results have considered also different values of the Rice factor, and also the channel model proposed in previous subsections, which is labelled as “WAICS”.

7.5 *Active Flow Control Simulation with Jamming Interference and Node Misbehaviour*

To test the operation of WAICs, we use the example of an active flow control (AFC) system enabled by a dense network of sensors and actuators over the wings of the airplane (see Fig. 9). The objective is to track the formation of the turbulent flows or the boundary layer transition between laminar and turbulent flows. This tracking needs to be in real time and with low latency to be able to enable the model of the actuation that have some known consequences on delaying the transition from laminar to turbulent flow. The performance of this system is directly related to how accurate the central computer on board the aircraft has a real time perspective of the turbulent layer transition. Delays in the network as well as attacks, and other impairments such as fading and multi path considering the channel models previously studied have been considered. The results in Fig. 10 show that when machine learning is used to predict the boundary transition layer and to detect potential interference attacks or deep fades of the communication channel, it is possible to track with less errors and in real time the boundary transition layer, which in the end would make the system more efficient. Each figure shows on the right hand side two curves, one that belongs to the real-time boundary layer, and the other curve is the curve as received by the central server. The second curve is prone to transmission errors due to fading, jamming attacks, or node misbehaviour. ML prediction helps detect parts of the curve where an error potentially occurred and therefore there is a need to compensate with a predicted value based on a training model previously calculated over similar turbulence and network conditions. All the results have been calculated using the channel models previously described over a hypothetical WAICs network

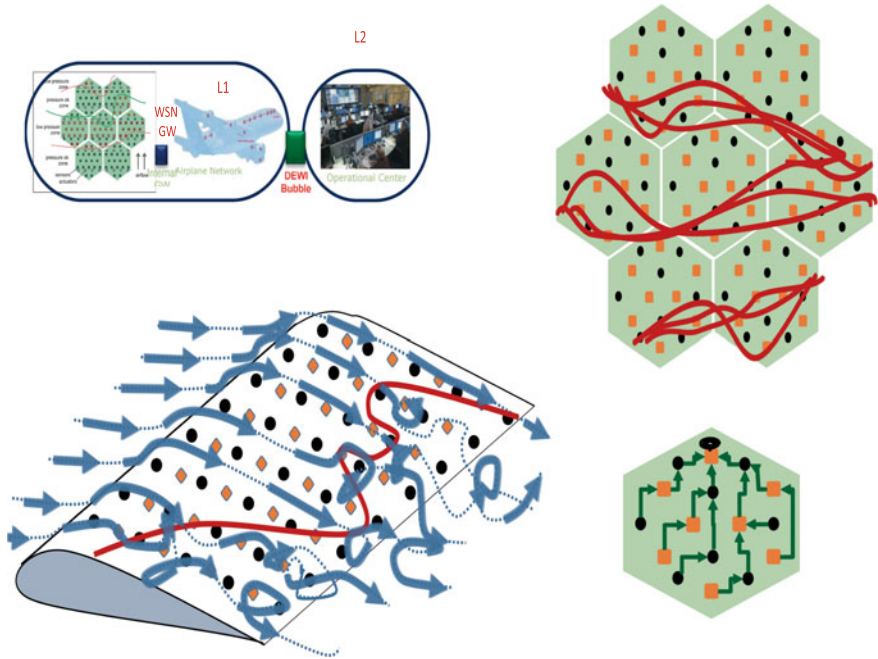


Fig. 9 Concept of active flow control using WAICs

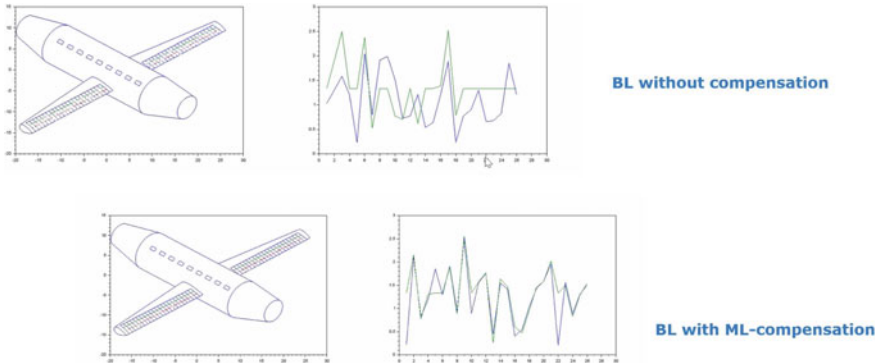


Fig. 10 System level simulation results for AFC system turbulence boundary layer tracking

operating in the 6 GHz band with 100 MHz of bandwidth and a power budget optimized to allow communication in free space loss over the surface of the wings of the mechanical model of the aircraft. The turbulence model used was proposed in the project SCOTT and consists of a space-time stochastic circular complex Gaussian model with a variable mean according to the aerodynamic parameters such as vehicle speed and angle of attack.

8 Conclusions

The results of our modelling of the Physical layer of wireless avionics intra-communication assisted by AI point towards several interesting conclusions. The use of MIMO can quickly reach an optimum point where wireless links behave as wire-line links. However, when interference is considered the demands can grow rapidly. The use of AI becomes fundamental in avionics scenarios with higher level of scattering distribution, and mainly to provide real-time capabilities for critical applications. We have shown that for an application of active flow control, the use of machine learning can compensate multiple errors in the tracking of a metric that measures the formation of turbulence over the wings of an aircraft. The errors are caused by a combination of fading, jamming interference and node misbehavior or node functional faults. A negative aspect of AI that was found in this work was the complexity of the algorithms that could have an impact on delay sensitive applications.

References

1. Kumar, M.: “Wireless Versus Wireline” Competing Broadband Access Technologies, 1 Sept. 2012. SSRN: <http://ssrn.com/abstract=2305028> or <http://dx.doi.org/10.2139/ssrn.2305028>
2. Liu, J., Demirkiran, I., Yang, T., Helfrick, A.: Communication schemes for aerospace wireless sensors. In: IEEE/AIAA 27th Digital Avionics Systems Conference, 26–30 Oct. 2008, pp. 5.D.4-1,5.D.4-9 (2008). <https://doi.org/10.1109/DASC.2008.4702861>
3. Long, L.N., Schweitzer, S.J.: Information and knowledge transfer through archival journals and online communities. AIAA Paper 2004-1264, Aerospace Sciences Meeting, Reno, NV (Jan. 2004)
4. Harrington, M.: Introduction to wireless systems in aerospace applications. Presentation IEEE Proceedings. <http://www.ieee-stc.org/proceedings/2008/pdfs/MH1933.pdf>
5. Fly-by-Wireless (FBWSS): Benefits, risks and technical challenges. CANEUS Fly-by-Wireless Workshop, Orono, ME, USA 08/24/2010. Dipl.-Ing. Oroitz Elgezabal, German Aerospace Center (DLR), Institute of Flight Systems (2010)
6. Field, S., Arnason, P., Furse, C.: Smart wire technology for aircraft applications. In: Proceedings of the 5th Joint NASA/FAA/DoD Conference on Aging Aircraft, Orlando, FL (Sept 2001)
7. Dornheim, M.A.: New rules and hardware for wiring soon to emerge. In: The Aviation Week Space Technology, vol. 2 (Apr. 2001)
8. Stone, T., Alena, R., Baldwin, J., Wilson, P.: A viable COTS based wireless architecture for spacecraft avionics. In: IEEE Aerospace Conference, 3–10 Mar. 2012, Big Sky MT, pp. 1–11 (2012). <https://doi.org/10.1109/AERO.2012.6187104>
9. Studor, G.: NASA Fly-by-Wireless Update. <http://hdl.handle.net/2060/20100031691>
10. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. *Comput. Netw.* Elsevier **38**, 393–422 (2002)
11. Rawat, P., Singh, K.D., Chaouchi, H., Bonnin, J.M.: Wireless sensor networks: a survey on recent developments and potential synergies. *J. Supercomput.* **68**, 1–48 (2014). <https://doi.org/10.1007/s11227-013-1021-9>
12. Technical characteristics and operational objectives for Wireless avionics intra-communications (WAIC) Report M. 2197 (ITU-R Report). <http://www.itu.int/pub/R-REP-M.2197>

13. Technical characteristics and protection criteria for Wireless Avionics Intra-Communication systems, Recommendation ITU-R M.2067, approved Nov. 2014. <http://www.itu.int/rec/R-REC-M/recommendation.asp?lang=en&parent=R-REC-M.2067>
14. Technical conditions for the use of the aeronautical mobile (R) service in the frequency band 4 200- 4 400 MHz to support wireless avionics intra-communication systems, Report ITU-R M.2283, approved July 2015. <http://www.itu.int/rec/R-REC-M/recommendation.asp?lang=en&parent=R-REC-M.2085>
15. Technical characteristics and spectrum requirements of Wireless Avionics Intra-Communications systems to support their safe operation, Report ITU-R M.2283, approved Dec. 2013. <http://www.itu.int/pub/R-REP-M/publications.aspx?lang=en&parent=R-REP-M.2283>
16. Technical conditions for the use of the aeronautical mobile (R) service in the frequency band 4 200-4 400 MHz to support wireless avionics intra-communication systems, Report ITU-R M.2283, approved July 2015. <http://www.itu.int/rec/R-REC-M/recommendation.asp?lang=en&parent=R-REC-M.2085>
17. Technical characteristics and spectrum requirements of Wireless Avionics Intra-Communications systems to support their safe operation, Report ITU-R M.2283, approved Dec. 2013. <http://www.itu.int/pub/R-REP-M/publications.aspx?lang=en&parent=R-REP-M.2283>
18. Consideration of the aeronautical mobile (route), aeronautical mobile, and aeronautical radio navigation services allocations to accommodate wireless avionics intra-communication, Report ITU-R M.2318, approved Nov. 2014. <http://www.itu.int/pub/R-REP-M/publications.aspx?lang=en&parent=R-REP-M.2318>
19. Compatibility analysis between wireless avionics intra-communication systems and systems in the existing services in the frequency band 4 200-4 400 MHz, Report ITU-R M.2319, approved Nov. 2014. <http://www.itu.int/pub/R-REP-M/publications.aspx?lang=en&parent=R-REP-M.2319>
20. Robles, R., Tovar, E., Cintra, J., Rocha, A.: Wireless avionics intra-communications: current trends and design issues. In: International Conference on Digital Information Management (ICDIM 2016). 19–21 Sept. 2016. Porto, Portugal (2016)
21. ITU Preliminary Document 5B/167-E, Characteristics of WAIC systems and bandwidth requirements to support their safe operation
22. Wilson, W., Atkinson, G.: Wireless sensing opportunities for aerospace applications. *Sens. Transducers J.* **94**(7), 83–90 (2008)
23. Goldsmith, D., Gaura, E., Brusey, J. et al.: Wireless sensor networks for aerospace applications-thermal monitoring for a gas turbine engine. In: Proceedings of Nanotech Conference and Expo, pp. 507–512. CRC Press-Taylor & Francis Group, Boca Raton, FL (2009)
24. Collins, J.: The challenges facing U.S. navy aircraft electrical wiring systems. In: Proceedings of the 9th Annual Aging Aircraft Conference (2006)
25. Haowei, B., Atiquzzaman, M., Lilja, D.: Wireless sensor network for aircraft health monitoring. In: Proceedings of the First International Conference on Broadband Networks, 2004. BroadNets 2004. pp. 748, 750 (25–29 Oct. 2004)
26. Yedavalli, R.K., Belapurkar, R.K.: Application of wireless sensor networks to aircraft control and health management systems. *J. Control Theory Appl.* **9**(1), 28–36 (2011)
27. Bai, H., Atiquzzaman, M., Lilja, D.: Wireless sensor network for aircraft health monitoring. In: Proceedings of the 1st International Conference on Broadband Networks. IEEE Computer Society, pp. 748–750. Los Alamitos, CA (2004)
28. Hamman, R.: Wireless solutions for aircraft condition based maintenance systems. In: IEEE Aerospace Conference Proceedings, 2002, vol. 6, pp. 6-2877, 6-2886 (2002)
29. Floerkemeier, C., Sarma, S.: An overview of RFID system interfaces and reader protocols. In: IEEE International Conference on RFID 2008, pp. 232, 240 (16–17 Apr. 2008)
30. Falk, R., Kohlmayer, F., Kopf, A., Mingyan, L.: High-assurance avionics multi-domain RFID processing system. In: IEEE International Conference on RFID, pp. 43, 50 (16–17 Apr. 2008). <https://doi.org/10.1109/RFID.2008.451936>

31. Sambou, B., Peyrard, F., Fraboul, C.: Scheduling avionics flows on an IEEE 802.11e HCCA and AFDX hybrid network. *IEEE Symposium on Computers and Communications (ISCC)*, pp. 205, 212 (28 June 2011–1 July 2011). <https://doi.org/10.1109/ISCC.2011.598384>
32. Sambou, B., Peyrard, F., Fraboul, C.: AFDX wireless scheduler and free bandwidth managing in 802.11e(HCCA)/AFDX network. In: *7th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2011, pp. 2109–2114, (4–8 July 2011). <https://doi.org/10.1109/IWCMC.2011.5982860>
33. Zhang, C., Xiao, J., Zhao, L.: Wireless asynchronous transfer mode based fly-by-wireless avionics network. In: *IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC)*, pp. 4C5-1, 4C5-9, (5–10 Oct. 2013). <https://doi.org/10.1109/DASC.2013.6712589>
34. Ketcham, R., Frolik, J., Covell, J.: Propagation measurement and statistical modeling for wireless sensor systems aboard helicopters. *IEEE Trans. Aerosp. Electron. Syst.* **44**(4), 1609–1615 (2008)
35. ARINC664 INC. *Aeronautical Radio. ARINC Specification 664. Avionics Full-Duplex Switched Ethernet*. Annapolis, Maryland: Aeronautical Radio, Inc
36. *Ethernet IEEE 802.3 Standard for Information technology-Telecommunications and information exchange between systems- Local and metropolitan area networks-specific requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*
37. *ISOCAN ISO 11898-1:2003 Road vehicles-Controller area network (CAN)-Part 1: Data link layer and physical signalling*
38. Niebla, C.P.: Topology and capacity planning for wireless heterogenous networks in aircraft cabins. In: *Proceedings of PIMRC05*, pp. 2088–2092 (11–14 Sept. 2005)
39. Niebla, C.P.: Coverage and capacity planning for aircraft in-cabin wireless heterogenous networks. In: *Proceedings of VTC 2003*, pp. 1658–1662 (6–9 Oct. 2003)
40. Leipold, F., Tassetto, D., Bovelli, S.: *Wireless in-cabin communication for aircraft infrastructure*. Springer US, J. Telecommun. Syst. 1–22
41. Holzbock, M., Hu, Y.F., Jahn, A., Werner, M.: Advances of aeronautical communications in the EU framework. *Int. J. Satell. Commun. Netw.* **22**, 113–137 (2004). <https://doi.org/10.1002/sat.777>
42. *WirelessCabin, FP4 EU project*. <http://wirelesscabin.triagnosys.com/>
43. Riera Diaz, N., Holzbock, M.: Aircraft cabin propagation for multimedia communication. In: *Proceedings of the 5th European Workshop on Mobile/Personal SATCOMS*, Baveno (Lake Maggiore), Italy (Sept 2002)
44. Chao, Z., Yu, J., Pank, K.: Multiple access points deployment optimization in cabin wireless communications. *IEEE Antennas Wirel. Propag. Lett.* **12**, 1220–1223 (2013)
45. Wargo, C.A., Dhas, C.: Security considerations for the e-enabled aircraft. In: *Proceedings 2003 IEEE Aerospace Conference*, vol. 4, pp. 1533–1550 (8–15 Mar. 2003)
46. Sampigethaya, K., Poovendran, R., Bushnell, L., Mingyan, L., Robinson, R., Lintelman, S.: Secure wireless collection and distribution of commercial air-plane health data. *IEEE Aerosp. Electron. Syst. Mag.* **24**(7), 14 (20 July 2009). <https://doi.org/10.1109/MAES.2009.5208555>
47. Sampigethaya, K., Poovendran, R., Bushnell, L.: Secure operation, maintenance and control of future e-enabled airplanes. *Proc. IEEE* **96**(12), 1992–2007 (2008)
48. Sampigethaya, K., Poovendran, R.: Privacy of air traffic management broadcasts. *IEEE Digital Avionics*
49. Cramer, J.: Update on WRC-12 issues impacting wireless avionics intracommunications. *ITU-R Working Party 5B and Future Regulatory Considerations* (Sept 2010)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Use of Artificial Intelligence as an Enabler for the Implementation of ETCS L3 and Other Innovative Rail Services



Francisco Parrilla Ayuso, Jose Manuel González Delgado, Jose Antonio Giménez Gómez, Jorge Rubio Cañete, Alejandro Díaz Díaz, Rogelio Hernandez, Jaime Señor, Gabriel Mujica, Andrés Otero, Jorge Portilla, Jesús Félez, Miguel A. Vaquero Serrano, Arrate Alonso Gómez, Bernd-Ludwig Wenning, and Gonzalo Ezquerro

Acronyms

ACS	Adaptable Communication System
AI	Artificial Intelligence
AMQP	Advanced Message Queuing Protocol
ATC	Automatic Train Control
ATO	Automatic Train Operation
ATP	Automatic Train Protection

F. Parrilla Ayuso (✉) · J. M. González Delgado · J. A. Giménez Gómez · J. Rubio Cañete · A. Díaz Díaz
Indra Sistemas S.A. (INDRA), Madrid, Spain
e-mail: fparrilla@indra.es

J. M. González Delgado
e-mail: jmgonzalezd@indra.es

J. A. Giménez Gómez
e-mail: jagimenez@indra.es

J. Rubio Cañete
e-mail: jrubic@indra.es

A. Díaz Díaz
e-mail: adiazd@indra.es

R. Hernandez · J. Señor · G. Mujica · A. Otero · J. Portilla · J. Félez · M. A. Vaquero Serrano
Universidad Politécnica de Madrid (UPM), Madrid, Spain
e-mail: r.hlorite@upm.es

J. Señor
e-mail: jaime.senors@upm.es

© The Author(s) 2024
M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_19

CCS	Command Control and Signalling
DGPS	Differential Geo-Positional System
ECC	Elliptic-Curve Cryptography
EKF	Extended Kalman Filter
ETCS	European Train Control System
FPGA	Field Programmable Gate Array
GLONASS	Global'naya Navigatsionnaya Sputnikovaya Sistema
GoA	Grade of Automation
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
I2V	Infrastructure to Vehicle
IMU	Inertial Measurement Unit
IoT	Internet of Things
ITS	Intelligence Transport System
KEM	Key Exchange Mechanism
KPI	Key Performance Indicator
MANET	Metropolitan Area Networks
ML	Machine Learning
MQTT	MQ Telemetry Transport
NIST	National Institute of Standards and Technology
OBU	On Board Unit
PQC	Post-Quantum Cryptography
RSSI	Received Signal Strength Indicator
S2R	Shift2Rail Initiative

G. Mujica

e-mail: gabriel.mujica@upm.es

A. Otero

e-mail: joseandres.otero@upm.es

J. Portilla

e-mail: Jorge.portilla@upm.es

J. Félez

e-mail: jesus.felez@upm.es

M. A. Vaquero Serrano

e-mail: miguel.vaquero.serrano@upm.es

A. Alonso Gómez

Escuela Politécnica Superior de Mondragón Unibertsitatea (EPS-MU), Mondragón, Spain

e-mail: aalonso@mondragon.edu

B.-L. Wenning

Munster Technological University, Bishopstown, Cork, Ireland

e-mail: berndludwig.wenning@mtu.ie

G. Ezquerro

JIG Internet Consulting SL (JIG), Logroño, Spain

e-mail: gonzalo@jig.es

SME	Small and Medium-Sized Enterprises
TCMS	Train Control and Management System
TWR	Time Difference of Arrival
TEN-T	Trans-European Transport Network
UWB	Ultra-Wideband
V2I	Vehicle to Infrastructure
V2X	Vehicle to Everything
WSN	Wireless Sensor Network

1 Introduction (INDRA)

During the last decades, rail transport has been characterized for having a long haul and slow developments. However, in the past years the different European Initiatives as Shift2Rail [1]¹ or ECSEL [2]² have been able to include the last improvements and trend technologies to reach the current society needs for railways.

The irruption of new technologies (IoT, Cloud/Edge Computing and new V2X technologies) is increasing the number of devices connected due to the smartphones and vehicles connectivity developments. This involves:

- V2X Communication improvements such as the development of 5G and 802.11.bd for critical communications.
- The inclusion of IoT and Cloud/Edge Computing technology foreseen as one of the main technologies that will enhance all the rail management, development, operation infrastructure and resources in a short term.

All these connected elements bring us a big amount of available data, that together with the use of Artificial Intelligence allows to consolidate the enablers for ETCS L3 (Train Integrity, Absolute Safety Train Positioning and V2X communications) and to **develop new services** in order to improve safety and to increase the efficiency in railway operation.

¹ Shift2Rail, as the first European rail initiative to seek focused research and innovation (R&I) and market-driven solutions by accelerating the integration of new and advanced technologies into innovative rail product solutions (2016–2023) that continues with ERJU (2023–2030) tries to address the new Rail Challenges by developing, integrating, demonstrating and validating innovative technologies and solutions following safety standards. To measure the value of these new solutions the S2R Multi-Annual Action Plan set the following key performance indicators [6]:

- 100% increase in rail capacity, leading to increased user demand;
- 50% increase in reliability, leading to improved quality of services;
- 50% reduction in life-cycle costs, leading to enhanced competitiveness;
- removal of remaining technical obstacles holding back the rail sector in terms of interoperability and efficiency.
- reduction of negative externalities linked to railway transport, in particular noise, vibrations, emissions and other environmental impacts.

² Addressed by the SCOTT Project use cases of railways [3].

SCOTT—Secure COnnected Trustable Things, Website: <https://www.scottproject.eu>.

In addition, the rail domain requires infrastructure and resources that are still expensive and require a long-time planning and execution. Therefore, the usage of the rail systems must be highly optimized, following strict security and safety regulations. **AI is the promising candidate**, identified as a trend technology in the following ten years by Gartner [4] to solve the limitations that were found in SCOTT, such as the lack of smoothing mechanisms in train coupling that provide comfort and safe conditions to the passengers and cargo, or the interconnection between rail traffic and road traffic in the smart city environment.

The SCOTT project following the ECSEL objectives, was focused on improving the rail systems capabilities in a safe and secure manner. However, the functionalities could be improved by moving the focus to the passenger's comfort and cargo security.

Using these key indicators and taking the Shift2Rail Innovative Programmes as a reference in the railway domain works, and based on SCOTT results, the project aims to continue the development of key technologies including AI to foster innovations in the railway domain and allocate the control resources in a more efficient way.

Moreover, in the project, the capacity of the railway systems to collaborate with other domains is one of the main strengths. Specifically, in a smart city environment, for the multi-modal traffic control it is essential to work in a cross-domain between rail and road, implementing AI mechanisms to allow a safe and secure management of the controlled area.

The scope of the project concerning the railway domain can be accomplished by improving safety and security, ensuring connectivity of all the railway systems, t by tackling the following objectives:

- Improve the Infrastructure management by increasing **the safety and security for both passengers and cargo**, providing clever, decentralized and flexible systems to enhance and substitute the Control Command and signalling (CCS) systems.
- Provide connectivity among maritime, road and rail domain through a full cargo manifest management, making use of a secure integration platform for the operators of different domains.
- Provide **safe and secure I2V/V2I communication** technology fully compatible with rail communication standards able to broadcast relevant information to different types of vehicles (train, cars) and the infrastructure in a trustable and smart way.
- Keep the **human user in the loop** during design, development, and evaluation to ensure safe, acceptable, and trustworthy overall performance of the system.
- Improve the flexibility on the rail and automotive domain, by increasing the efficiency in the decisions by making use of Edge-based Artificial Intelligence mechanisms to manage multimodal jam, providing interoperability in the smart city environment.
- **Increase the digitalization** in the railway domain to decrease costs and bureaucracy for EU citizens.
- Address social challenges through the **improvements over the electronic components and systems involved in the railway infrastructure**, increasing the globally competitive in the European Union by means of the Use Cases developed.

- **Inclusion of Small and Medium-Sized Enterprises (SMEs)** in the developments to reinforce solutions.

2 The Use Cases (INDRA)

2.1 T5.7 Intelligent Transportation for Smart Cities

The aim of this UC to use Artificial Intelligence (AI), making use of the system that priory among the rail and other different urban stakeholders, to enhance the control of the urban traffic. The integration of wireless technologies with AI is used to support V2X secure communications in the development of smart rail services based on Metropolitan Area Networks (MANET).

Current railway infrastructures coexist with other domains in urban environments where traffic events may occur. This UC aims to spread these developments to build, making use of AI, a smart management system for urban traffic control. This improvement on urban traffic management will reduce the number of injuries and human losses, by increasing safety and security in the railway lines due to the direct communication between the railway and other domains.

The urban traffic management system must make the decisions considering the information provided by all the different domains, (rail and road). Increasing the intelligence of the decision maker, the traffic jams in urban areas will be managed in a more effective way, making use of Artificial Intelligence mechanisms to develop the trustable decisor (Fig. 1).

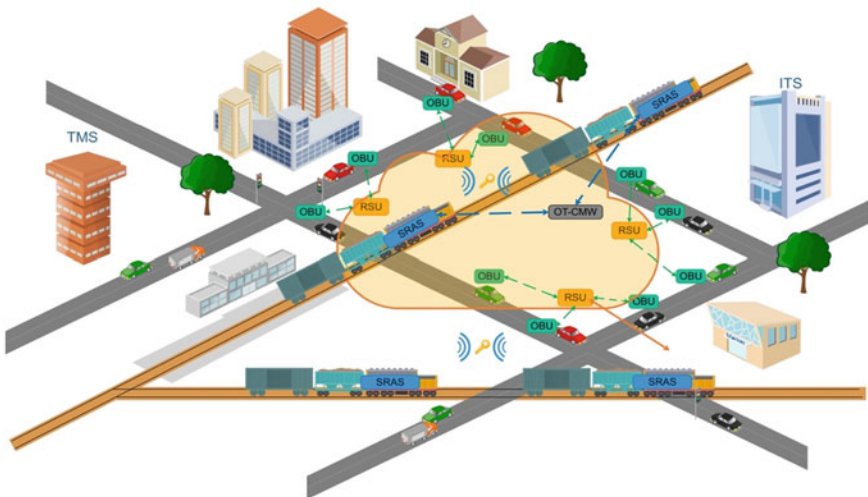


Fig. 1 Smart city transport—UC overview

The innovation of this solution is based on:

- Increasing the communication between all involved actors in the specific critical area for considering as much data as possible before assigning priorities to a specific actor in an intelligent way.
- Enhancing the management of cross-domain areas making use of Edge-based AI mechanisms.
- Managing multimodal jams.
- Improving the efficiency on the rail and automotive domain, by increasing the efficiency in the decisions.

The aim of the UC5.7 is focused on the development of a decision-making system, which makes use of an AI mechanism to manage the traffic jams, improving road and rail traffic in the cities and maintaining the trustability of the system.

The developments carried out in this UC are based on previous works performed in SCOTT and DEWI projects.

Along the European railway lines, there are many critical scenarios where several accidents occur every year. Many of these are located on the level crossing, where different actors such as pedestrians or vehicles are involved. During 2017 in Spain, just the incidents in the level crossings, both with and without safety barrier, rise to 25 cases according to the annual report performed by the “*Agencia Estatal de Seguridad Ferroviaria*” [5].

The rail operators are interested in increasing safety in the critical points making use of the new technologies due to most of these scenarios not being able to eliminate the risk. In fact, a new system called Trustable Warning System (TWS) will be brought to the market in the coming years for managing critical scenarios such as level crossings or working areas through the use of wireless communications [6].

All these explained innovation works allow an efficient implementation of smart, trustable, safe, and secure systems for rail automation that:

- Increment the safety in critical scenarios such as level crossings.
 - by providing intelligence to the trustable decisor
 - by increasing the communication between actors in critical scenarios
- Enhance management of the cross-domains areas
 - by establishing priorities
 - by managing multimodal jams
- Minimize the CAPEX/ cost
 - by implementing wireless solution
- Improve the reliability, safety and security of the system:
 - by applying safety and security rules and directives for a reliable communication between the urban stakeholders involved.
 - by providing the specification for the safety and security requirements necessary for the use of distributed AI mechanisms.

2.2 *T5.8 Intelligent Automation Services for Smart Transportation*

This Use Case aims to improve automation for operation for different rail services making use of Artificial Intelligence. The use of wireless technologies with AI developments is used to support V2X secure communications and to enhance V2V communications for coupling compositions.

Specifically, this Use Case is focused on the use of Artificial Intelligence for automating the smoothing of coupling and uncoupling processes, the grade of automation by improving ATO and ATP mechanisms and the security at V2X communication level.

The works performed in SCOTT Virtual Coupling [7]—that provide a solution that implements a safe and secure wireless virtual coupling—do not resolve a specific issue related to the comfort of passengers and the cargo safe trip during the coupling mode. The mechanisms to couple the trains require a certain level of automation with a specific timing. This timing completely covers the technical requirements to successfully couple and uncouple the involved trains. However, once the coupling process is performed, the distance between trains can suffer abrupt changes due to the variation of the speed and the acceleration/deceleration. These events may affect passengers' comfort and certain types of cargo operation can be affected.

Following X2RAIL-3 Virtual Train Coupling System concept [8] and to avoid these consequences over the passenger trains or the cargo, the procedures that keep the trains coupled must be smoothed. Increasing the intelligence in the automation procedures involved through Artificial Intelligence methods, these consequences can be solved or at least reduced. These can make the travel experience more comfortable for the train passengers. In the case of freight trains, the smooth of the process of speed variation can reduce the damage over the cargo. Furthermore, the inclusion of AI in the works developed in SCOTT project may allow the increase of line capacity and optimize the track occupancy (Fig. 2).

In this way, this Use Case aims to make use of Artificial Intelligence to improve the Automatic Train Operation (ATO) which is mainly used on **automated guideway transits** and **rapid transit systems** where is easier to ensure safe rail operations improving the rail capacity lines.

The innovation of this solution is based on:

- Improve flexibility by connecting all involved stakeholders by means of an automated and distributed system.
- Provide mechanisms to improve network capacity with a safe, secure and trustable system.
- Decentralize the control of decisions making use of AI.
- Provide optimal control forecasting of all the devices of signaling systems along the railway track, by connecting all to all.
- Smooth the speed change processes to allow a more practical and safe coupling process.

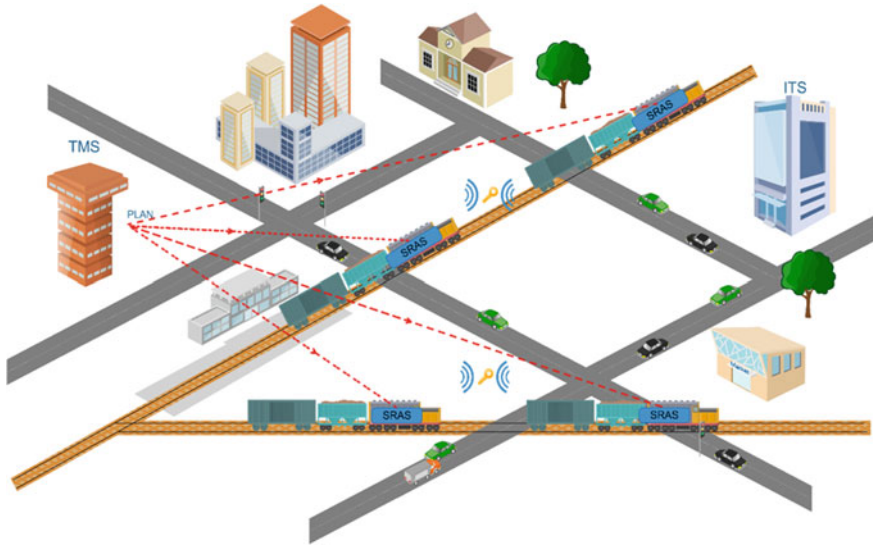


Fig. 2 Intelligent automation services for smart transportation—UC overview

- Implement AI technology to improve coupling, traveling and uncoupling phases.

This Use Case focuses on the automation of different train operation processes making use of Artificial Intelligence. The analysis of current technologies concerning the smoothing of virtual coupling and enhancing of ATO, ATC and ATP systems and the different grades of automation is needed to previously define the grade of automation that the system must reach.

The integration of existing technologies and new ones to enhance the automatic train operation to make the difference in the railway market. By improving functionalities such as speed control, timing control of the stops and decentralizing the decisions, it is possible to enhance the current state of the railway transportation, having an important impact in the railway market.

Moreover, the use of AI mechanisms to improve the deployment of smoother virtual coupling maneuvers during the trip will help to make the speed changes processes more comfortable for the passengers and safer for the cargo.

The development of this use case must accomplish the objective of solving or reducing the drawbacks and comfort for passengers and certain types of cargo operations in a safe and secure way. This will directly have an impact on the railway traffic, giving a new vision to the market concerning the possibility of reducing risks and improving the line capacity by making more efficient and comfortable coupling and uncoupling maneuvers.

Moreover, this development has to accomplish the objective of increasing the grade of automation of both the Rail operations and infrastructure, by implementing safe and secure capabilities. This will have an important impact in the market, allowing the implementation of a trustable system which enhances the features of

current ATO and ATP systems by using AI, introducing IoT concepts in the railway market to make it more competitive.

All these explained innovation works will allow an efficient implementation of smart, trustable, safe and secure systems for rail automation that will provide:

- The increment of the efficiency of the rail infrastructure and the On-Board systems:
 - by the inclusion of automating the coupling and uncoupling maneuvers along the tracks, via Artificial Intelligence.
 - by smoothing these processes and making them comfortable for the rail users, both for passengers and freight lines.
- The progressive conversion of conventional lines into ATO lines:
 - by providing intelligence to the railway traffic processes connecting all to all
 - by increasing the grade of automation of rail operation and infrastructure
- Improvement of the system flexibility:
 - by introducing the automation of the CCS
 - by including distributed solutions to efficiently manage the exchange of information
 - helping to acquire major capacity and improving the timetable adherence due to a more efficient traffic management (i.e., minimizing unexpected train stops or reducing the safe distance between trains).
- Major capacity and improved timetable adherence for a more efficient traffic management:
 - by minimizing unexpected train stops
 - by safely reducing the distance between trains
- Improvements on the reliability, safety and security of the system:
 - by applying safety and security rules and directives for reliable communication between the infrastructure and the trains.
 - by providing the specification for the safety and security requirements necessary for the use of distributed AI mechanisms.

3 The Platform (INDRA, JIG)

To be able to perform and implement the different use cases Indra has make use of a common framework based on an IoT Platform developed in SCOTT, to deploy the different modules providing native communications through the different subsystems enabling to collect, fuse, enrich and exploit all the data on the edge or in the cloud following the safety, security and privacy levels required for the different systems.

The SCOTT project is built on the excellent basis of the predecessor project DEWI³ and thereby, among others, reuse and extend the well-established DEWI Bubble concept and the related, ISO 29182 compliant multi-domain high-level architecture [9]. Within the DEWI project key solutions for wireless seamless connectivity and interoperability in smart cities and infrastructures were developed. DEWI was started in March 2014 as part of the ARTEMIS Joint Undertaking and ended in April 2017.

The DEWI bubble concept, the defined DEWI high-level architecture as well as the DEWI technology items, has been used as a starting point for systems development within SCOTT and can be seen as the continuation of DEWIs' technology solutions.

Complementary to DEWI the SCOTT project (2017–2020) puts additional focus on:

- extending and connecting Bubbles and integrating distributed Bubbles into the Cloud
- extending the high-level architecture concerning security, trustworthy and cloud integration
- the development of safe and secure solutions for wireless distributed systems: implementing a meta-bubble layer where multiple bubbles need to cooperate in deterministic (real-time) and secure way to establish systems in distributed locations
- elaboration of new approaches for secure distributed cloud integration—extending DEWI high-level architecture
- developing secure and trustable applications coming from new domains such as Health and Home (besides commercial/public buildings)

InSecTT goes a significant step further:

- **Bring Internet of Things and Artificial Intelligence together (“Artificial Intelligence of Things”, AIoT)**
- **Move AI to the edge**, i.e., provide **intelligent processing** of data applications and communication characteristics **locally at the edge** to enable **real-time and safety-critical** industrial applications
- Develop **industrial-grade secure** and **reliable solutions** that can cope with cyberattacks and difficult network conditions
- Enable **AI-enhanced wireless transmission**
- Provide trust measures for user acceptance, **making AI/ML more explainable** and not just a black box that cannot be understood
- Provide **re-usable solutions across industrial domains** (Fig. 3)

The SCOTT IoT Platform—Fig. 4—results grant to InSecTT the necessary components to develop and deploy the different AI modules and to interconnect all the different subsystems (On-Board, On-Track deployments) on the different domains (Rail, Road) including the simulation and validations tools.

³ DEWI—Dependable Embedded Wireless Infrastructure, Website: <http://www.dewiproject.eu/>.

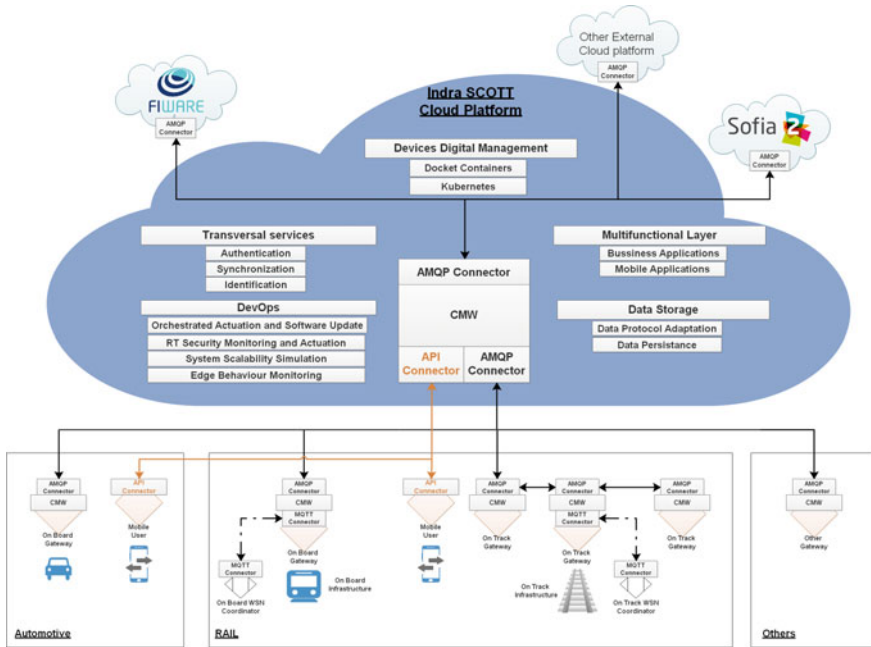


Fig. 3 The Indra IoT platform

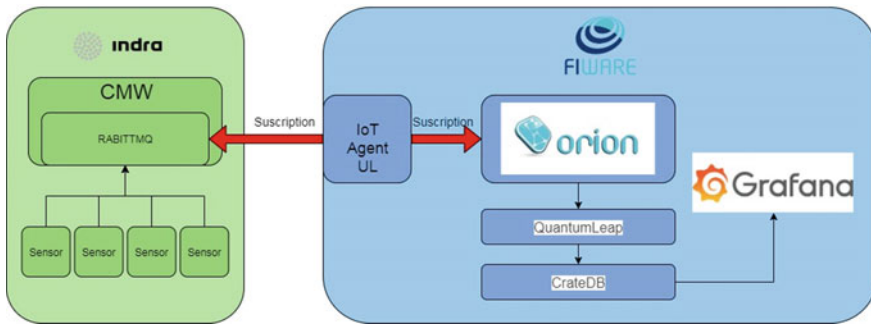


Fig. 4 Indra IoT platform–FIWARE integration

This platform has been essential to provide more than 30 millions of messages to the different AI modules (provided by different On-Board and On-Track subsystems) to be exploited for training and fine-tuning of the different AI modules.

To develop the applications and related services, an integration between the platform provided by Indra and a service structure based on Fiware has been carried out.

FIWARE [10] is an open-source initiative that aims to promote the creation of standards necessary to develop Smart applications in different domains: Smart Cities,

Smart Ports, Smart Logistics, Smart Factories, among others. Any Smart application is characterized by collecting relevant information for the application from different sources about what is happening at any given time. This is known as “context information”. Current and historical context information is processed, visualized and analyzed on a large scale.

To integrate Indra and Fiware environments, the Fiware IoT Agent UL 2.0 has been used. This IoT Agent allows to convert AMQP communication protocol into MQTT, essential to use the CE Context Broker, and the communication between both environments:

To complete the environment, different Dashboards and a Mobile application have been developed.

The end-user application developed in the project has the following objectives:

- Control unassembled rolling stock: positioning, wagon details, tractor heads, etc....
- Control of moving composite trains by mapping with points generated from GeoServer.
- Control and warnings on the sensor system installed on the train and connected through the WSN.

Figma framework has been used for the design of the application, on which the designs have been generated in a modular way. In this way, the components of the application can be reused to build new functionalities from these same components.

The design of the application contains the following functionalities:

- Control of unassembled rolling stock: Here, wagons and tractor units are controlled before the composition of the train, as it is show on Fig. 5.
- Management and control of the trainsets and associated sensors: From this view the composed trainsets are monitored, as it is show on Fig. 6.

By selecting the desired train, the details of the start, destination and intermediate control points of the train are monitored, as well as the status and metrics of the different sensors installed, as it is show on Fig. 7.

In addition, different control panels have been designed for real-time monitoring of the different data captured both from the sensor networks deployed and from the different subsystems; as well as for their subsequent analysis to verify the different developments implemented, as it is shown on Fig. 8 where it is presented an example of train composition, train integrity and train length dashboard.

4 Relevant AI Enablers Developed (INDRA)

Artificial Intelligence and Machine Learning are in a strong growing phase, making significant advances in difficult pattern recognition tasks [11]. But these current successes have largely come from systems that run on central servers with abundant computational and memory resources. When deployed on IoT systems, application

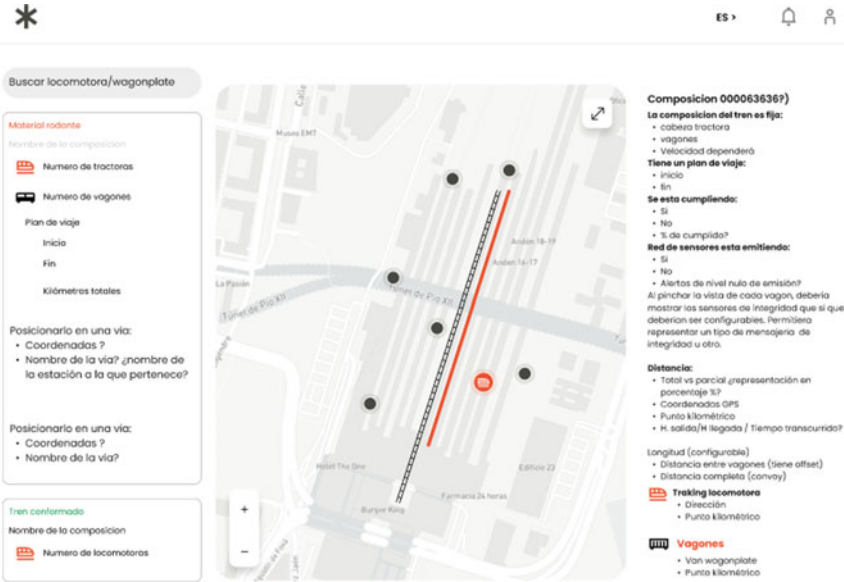


Fig. 5 Rolling stock management

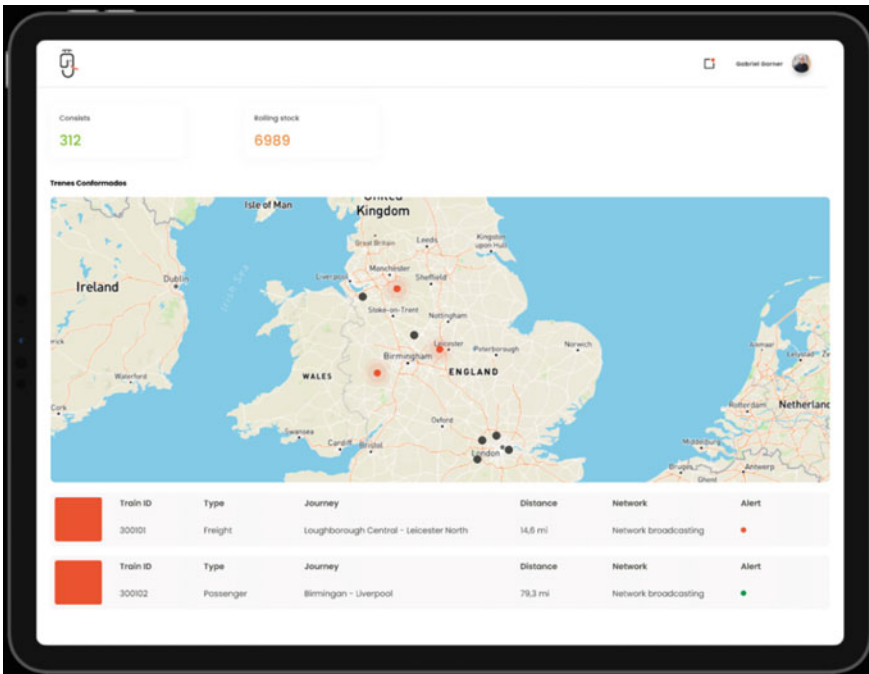


Fig. 6 Train management

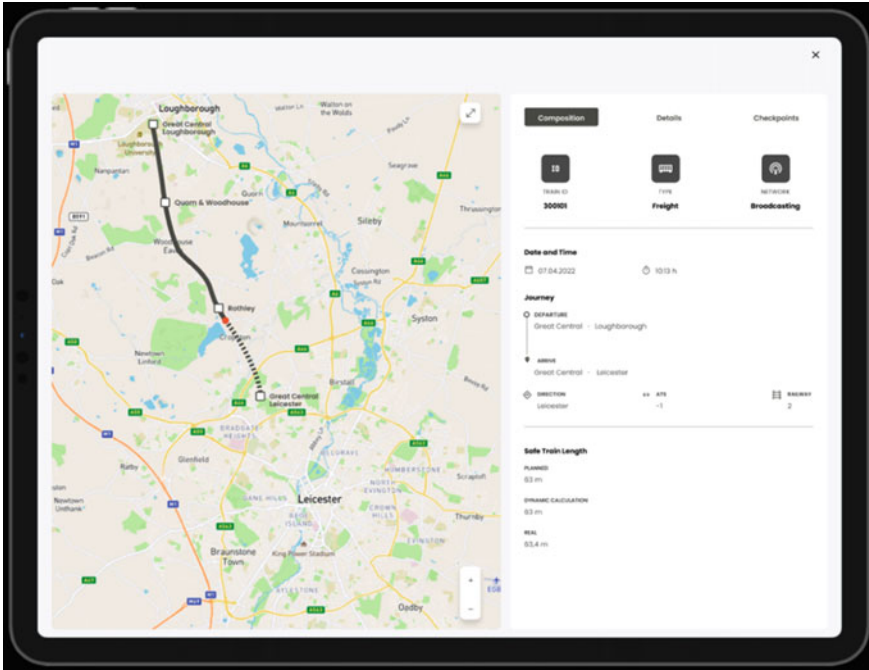


Fig. 7 Train detail

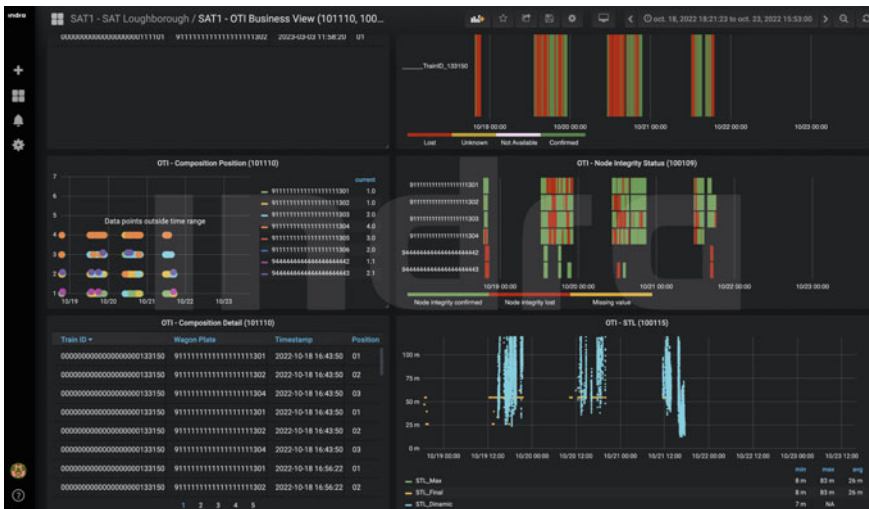


Fig. 8 Train composition, train integrity and train length dashboard

successes rely on high bandwidth connectivity, and very loose latency requirements for decisions. Industry, however, is now realizing the need for diverse applications that can run at very low latencies, and this will require new algorithms and models that can be deployed at the edge of the network, or on client devices. This has led to the rise of **fog computing** as a technology (e.g. [12]), and industry is now developing new architectures, capable of running on devices with limited CPUs and memory (e.g. [13, 14]). In addition, as **intelligence and autonomy** are being pushed out to the edge or to the device, there is a need for coordination of, and collaboration between, these intelligent systems, possibly in real time. **Distributed AI, or Multi-agent reasoning, is a well-established research discipline** (e.g. [15, 16]), but again relying on loose latency requirements and with few constraints on communication.

AI Managing and Monitoring IoT Systems and Simulations

Monitoring and managing the vastly increasing amount of IoT devices in operational and development phases is an ongoing challenge, which is yet to have harnessed the potential of AI/ML. Operational system status covers security, performance, device, and data integrity issues. In the development phase, comprehensive simulations for the interactions between different IoT devices and environment are necessary. AI solutions allow automatic realistic test case generations as well as automated evaluation of simulated performance under different conditions. Many solutions exist in providing management and monitoring of IoT systems and simulations, which are reliant on pre-defined rules input by humans. Deploying different machine learning methods and/or deep learning has great potential for increased efficiency and less reliance on human interaction.

Distributed AI on Edge Architecture

There is a clear need to develop reduced-order AI/ML models and algorithms which can be executed on resource-constrained devices. Further, there is a need for a general framework which can determine efficient placement of computational responsibilities over the end-to-end device-edge-cloud architecture. This may involve models trained in the cloud and then reduced and deployed on the edge, or it may require collaborative AI systems which execute what they can at the edge and communicate boundary cases along the chain for higher-powered processing.

Trust Framework for AI/IoT Development

InSecTT will provide a trustable framework for evaluating and developing trusted AI solutions. InSecTT will extend existing principles for trustworthy AI (such as transparency, controllability, and predictability) and generic Trust Framework created in the SCOTT project for providing models in addressing trust issues and trust requirements in AI operations used in IoT/Cyber physical systems.

4.1 AI Mechanisms for T2X Communications Systems (INDRA, EPS-MU, MTU)

The scenario is focused on the Adaptable Communication System (ACS), following X2RAIL-1 [17], X2RAIL-3 [18] works executed in Shift2Rail, to be used into a vehicle and the treatment of the communication system interfaces to reach a service specified accuracy. The figure shows, in a graphical perspective, the procedure to follow in the scenario. All the solutions, both in receiver and transmitter sides, exchange all the signalling information available for each of them to be able to select the channel in a coordinated manner. This data is provided to the AI modules to make the channel selection in a coordinated manner with the other side (Fig. 9).

Concerning the T5.7, sub-BB 3.4.1. “*Intelligent Routing Platform*”, a decision maker has been developed to make dynamic decisions about the best channel for wireless communications. The selection is based on communication system metrics and rail KPIs using AI and ML methods. For this purpose, it is required to receive the output of the sub-BB 3.3.1 that provides precise information in real time from the different communication systems. The output of these modules is the communication system selected for each interface (V2I/I2V communications, Public Communication System, etc.).

During the project, INDRA has worked on studying the Multi-Access Management Services (MAMS) protocol as an Adaptable Communication System (ACS) selector when several networks are involved. In this type of scenario, a user can be simultaneously connected to multiple networks using different access technologies and network architectures that offer different QoS capabilities variable in time. The management of these scenarios nature is the motivation to use MAMS.

The smart selection of the network channel improves the QoS of the user. Focusing on the ACS scenario where it is deployed, it is not static since it is a transportation environment, so the characteristics of the available networks are constantly changing at each moment.

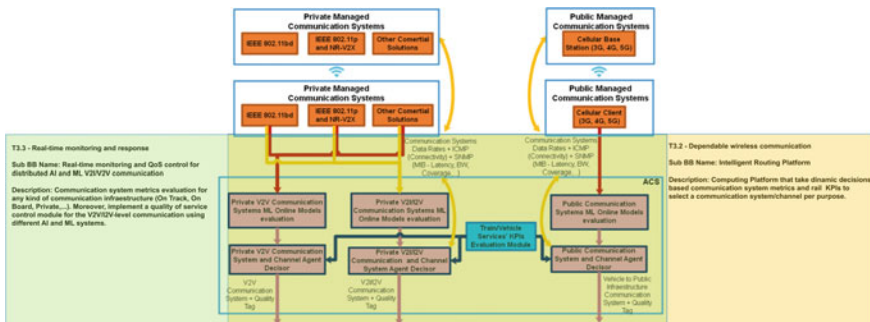


Fig. 9 T2X communication system architecture

The MAMS framework gives a solution for managing multi connectivity scenarios. The mechanisms used are not dependent on any protocol, since the idea of MAMS is to complement the existing ones giving a way to negotiate and configure them to enhance the use of the network. The fact that MAMS framework does not depend on the access technologies, allows the implementation to be change-proof regarding future new protocols and mechanisms. To know which network should be selected, the networks will be constantly monitored to ensure the best channel selection.

Has been concluded that the requirements of the MAMS framework to cover the ACS functionalities are as follows:

- Access-technology-agnostic interworking: the type of technology used by the network does not have to be relevant to the system.
- Independent access path selection for uplink and downlink: the communication should be able to be through different networks for uplink and downlink.
- Adaptive access network path selection: all the available networks must be monitored to select the best one considering delay and capacity.
- Multipath support and aggregation of access link capacities: it must be able to enable simultaneous paths. The aggregation must support existing protocols.
- Lossless path switching when changing from one connection to other, the framework should ensure mechanisms to receive messages in order.

The intelligent routing platform solution based on TBB3.4 and adopted in UC5.7 lead by INDRA in collaboration with the MTU and EPS- MU. INDRA has overseen designing and defining the high-level architecture based on the UC context and the development. Besides the identification of the requirements and definition of the KPIs, the developments has been also supervised as well as the different tests.

This development is a part of the intelligent routing platform module, which serves for both the scenarios of UC5.7 and UC5.8. However, in this case it has been focused on the Vehicle-to-Anything (V2X) communications. It is about the development of a hybrid vehicular communication platform, aka Multi-Radio Access Technology (multi-RAT) for vehicular communications. EPS-MU understands Dependability as a measure of availability and reliability, and in the field of wireless communications is governed by:

- Coverage probability of the network,
- Latency of data transmission, and
- Transmission error probability.

EPS-MU aims to address each of them by developing a vehicular communication platform which:

- Implementation of BS/RSU capabilities to enhance coverage (multi-Radio Access Technology in the 5.9 GHz band: 802.11p and C-V2X),
- Implementation of Edge node capabilities to decrease latency of data transmission, and

- Implementation of next generation Dedicated Short Range Communication protocol (specification under development in ETSI).

Because of the requirements of INDRA on the use of MAMS protocol, another bullet point has been added to the list, *implementation of the MAMS protocol on a virtual environment*, in close cooperation with MTU.

These actions have been reported within WP3 BB3.2 deliverables and a set of 21 requirements have been defined by EPS-MU. The results obtained from these methods have been important for the Adaptable Communication System (ACS).

This submodule will be used for testing AI algorithms also developed within InSecTT, with the aim of estimating the currently available data rates, and will make a near future prediction about how these data rates will develop. Additionally, to this AI functionality, EPS-MU initiated collaboration with ISEP, in relation to AI algorithm identification for channel prediction purposes. Yet this latter solution is in an early developing stage and thus has not been integrated into the ACS.

INDRA together with MTU and EPS-MU developed the smart router in the SCOTT project, whose objective was to select the best of the 3 protocols to carry the train-ground/ground-train/train-train communication.

In the context of this use case, sub-BB 3.3.1 “*Real-time monitoring and QoS control for distributed AI and ML V2X*” is about the development of AI and ML methods to monitor and control V2X communication links. The output of these methods is important for the Adaptable Communication System (ACS) to decide which communication links to use to fulfil the use case service KPIs. For each of the communication links that are available to the vehicle, AI algorithms estimate the current available data rates, and will make a near future prediction about how these data rates will develop. The AI algorithms used here will be online learning methods, for example based on Online Support Vector Regression (SVR), that take available link quality parameters as input. The link quality parameters to be used depend on the technology used on the link (e.g., 4G, 5G or other technologies).

4.1.1 Results

Within the scope of the ACS module development work, AI methods for uplink data rate estimation and interface selection have been developed with a focus on cellular networks, in particular 5G. Compared to earlier technologies such as 3G and 4G, data rate estimation based on link quality parameters is more challenging in current 5G deployments as these are usually 5G NSA (non-standalone), which means they rely on a 4G network for management and signaling and only use the 5G carrier for the actual data traffic. This challenge, as well as the proposed solutions and achieved results for data rate estimation and interface selection, are discussed in detail in the chapter “AI-enhanced Connection Management for Cellular Networks”.

With the development of the MAMS protocol, the decision module of selecting an interface was moved from the onboard systems to the cloud. The onboard systems, running a MAMS CCM, report the outcome of data rate and latency estimation to

a cloud server running a MAMS NCM, and the cloud server responds by matching them to required KPIs and decides which interface to use, based on the reported parameters and the KPIs. The decision is then sent back to the onboard system via MAMS, and the onboard system switches between interfaces accordingly. This setup was tested by deploying the onboard system in a car, deploying a stationary IEEE 802.11p device as a roadside unit, and then driving the car on a route which featured areas with and without connectivity to the roadside unit. Estimated data rates and latencies, as well as interface decisions, were stored on the cloud server during the tests.

The data that was stored during the test has then been analysed to verify whether the outcome of the interface decision fulfils the KPIs as required. Figure 10 shows the evolution of the latency over time. In the first graph, the latency of the selected interface is plotted, where it is always lower than the maximum level defined for the KPI. The graph below shows the latency for each interface. Comparing with the selection, it can be confirmed that the NCM is able to change CCM's configuration when the interface—in use—no longer fulfils the requirement (in this case, changing from 802.11p to 5G NR). In a similar way, Fig. 11: Data rate for the selected interface (top) and data rates of all available interfaces (bottom) shows the data rate, where the selected interface is always above or the same as the minimum level set in the KPI. Both figures also show that the selected interface KPI is not always optimal, as the routing criteria is giving different priorities for each interface, but these priorities are only applied when multiple interfaces meet the requirements. The priority list used in these tests is [802.11p, 5G NR, LTE, 3G], giving higher priority from left to right.



Fig. 10 Latency for the selected interface (top) and latency for all available interfaces (bottom)

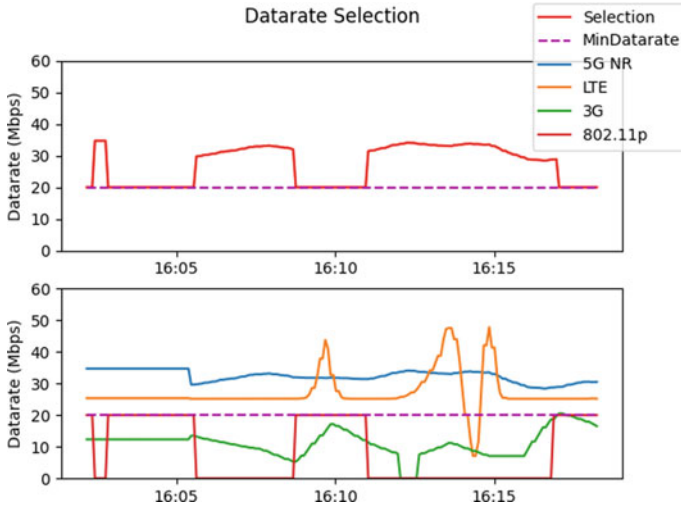


Fig. 11 Data rate for the selected interface (top) and data rates of all available interfaces (bottom)

4.2 AI Mechanisms for Train Positioning System (INDRA, UPM)

Two different TBBs have been designed and implemented: GNSS Measurements correction and adaptation evaluator and Train Positioning supervised decisor as depicted on the following Fig. 12.

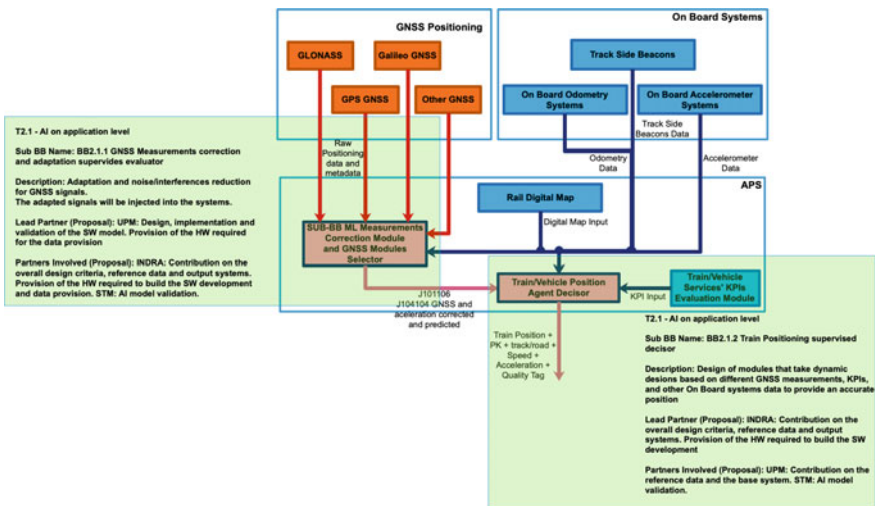


Fig. 12 Train positioning system architecture

The current Autonomous Positioning System (APS), which was developed in SCOTT project, is only based on Global Position System (GPS). To fulfil the positioning requirements and provide a more accurate measurement, the need to develop a generic Global Navigation Satellite System (GNSS)⁴ measurement evaluator has been identified.

This first TBB is focused on improving the GNSS signal quality of the positioning system by reducing interferences and noise generated. This is made making use of AI techniques to fuse the input GPS signal with measurements from odometers and inertial measurement units to obtain a corrected positioning signal.

Aid the selection of the GNSS source as well as to perform an initial noise and interference reduction stage.

The module will receive signals in real time from different GNSS systems such as Global Position System (GPS), Global'naya Navigatsionnaya Sputnikovaya Sistema (GLONASS), and/or Galileo to analyze, evaluate and correct them. It also receives measurements from the odometry system and the On-Board accelerometers to be fused with the selected GNSS source.

The main objective of the second TBB (the decisor) is to dynamically decide the best GNSS source based on multiple KPIs. The best selected source will also be corrected and predicted, using on-board systems, as in the previous TBB, to properly locate the train. The GNSS sources that are considered in real time by the decisor are Global Position System (GPS), Global'naya Navigatsionnaya Sputnikovaya Sistema (GLONASS), and/or Galileo.

The output will also be a digital map that allows the driver to know the real-time position in a graphical way including relevant data (legends) such as the kilometer point, the direction of the railroad track, the speed and/or the acceleration.

The positioning decisor solution based on TBB2.1 and adopted in UC5.7 has been developed by INDRA in collaboration with UPM. INDRA has overseen the requirement identification, high-level architecture design and specification as well as of providing the required datasets related with this module. UPM has developed and implemented the final module. Finally, INDRA has validated on real scenarios the different developments and implementations integrating the different submodules making use of X2RAIL-4 Demonstrator resources.

4.2.1 Results

A solution for edge prototyping based on the Cookie IoT hardware platform [19] developed at the Centro de Electrónica Industrial of the Universidad Politécnica de Madrid, which maximizes the concept of flexibility, reliability and modularity, is proposed in this work. To enable fast, robust, and energy-efficient integrity loss detection, two different networks are established depending on the type of module mounted in the Cookie: a low-power GNSS module and an UWB module. The GNSS cookie version also reports the IMU and RSSI values.

The GNSS module integrated within the IoT edge device a very low energy consumption, limiting its accuracy compared to other high end solutions. To enhance the accuracy of this approach, several solutions have been proposed.

GNSS Selector

The GNSS module can receive signals from GPS, DGPS, GLONASS and Galileo. By using range-based techniques to determine the position, the module can combine multiple signals to identify the satellites that offer optimal coverage in the current location.

To account for potential dynamic operation scenarios where not all system signals may be available, the operating mode of the GNSS edge device is selected, based on the quality of the signal, as determined by several parameters such as horizontal, vertical, and position dilution of precision. By monitoring these parameters in real-time, each node can determine and adjust its mode during operation. Furthermore, the coordinator also receives information about each node's active mode and quality of signal and can use configuration messages included in the routing protocol to set the thresholds of the algorithm or force a specific GNSS mode.

Extended Kalman Filter with Sensor Fusion

To overcome the disadvantages of fixed and irregular structures for track-side positioning, new approaches have been proposed, which fuse GNSS and IMU [20], resulting in what is known as sensor fusion. The reason they are used together is that the error provided by one system is reduced when integrated with the other, and vice versa. Traditional sensor fusion techniques come from the application of Bayesian filters such as the Extended Kalman Filter (EKF).

In this context, the Kalman filter is used to estimate the state of the measured element by processing a set of observations that contain both noise and uncertainty. The Kalman filter algorithm employs feedback to apply control and estimate the process. At any given time, the process state is predicted, and feedback is obtained through measurements, which incorporate noise and are weighted based on their certainty.

The system operates on the premise of correcting the measurements of the inertial system upon receipt of the GNSS signal, thereby preventing the accumulation of errors over time. In the absence of a GNSS signal, the EKF provides its solution integrating the IMU measurements. As a result, the fusion strategy provides a high update rate, a navigation solution in the event of GNSS absence and increased integrity, as faulty GNSS measurements will be detected and rejected.

Although the Kalman filter is a well-established technique for state estimation, especially when dealing with measurements incorporating noise, a major drawback is the requirement for prior knowledge of process and measurement noise statistics, since the variance of the Kalman filter is calculated as a function of the estimated variances. Therefore, the choice of the value of the fitting parameters makes a huge difference in the final state estimate.

For adjusting the filter and reducing the error of the output trajectory the STONEX S900 GNSS, a high precision GNSS receiver, has been used a reference ground truth,

i.e. the “real trajectory”. The STONEX S900 is a GNSS receiver that is used for high-precision positioning applications. It works by receiving signals from multiple satellite constellations, including GPS, GLONASS, Galileo, and BeiDou, and using a differential positioning to improve the accuracy of the measurements. It compares the measurements from a reference station that provides the corrections for the accurate positioning.

In the first iteration, the filter is adjusted manually. To evaluate the effectiveness of the manual adjustment, the error in the trajectory is estimated with respect to the reference trajectory. This error metric is provided both as a mean value and over the entire path. This value is the module of the deviation at the time of the measurement, and the performance is based on its minimization. The aim of the manual adjustment is to reduce this error by varying the components of the R and Q matrices. First, the covariance matrix R, which measures variance in the GNSS position measurement, is modified. This modification consists of decreasing the value of the components of this matrix, with the aim of giving more weight to the correcting measurements, and thus making the output trajectory more similar to the one marked by the GNSS signals. Once the R matrix is estimated, the noise matrix, Q, is adjusted. It measures the variance of the acceleration and orientation variables. The effect of Q is related to the error of the process, i.e. how controllable it is and how well known it is. In other words, it informs about the noise of the model. If this deviation is taken to be zero, the estimates are assumed to be ideal.

The final stage of the tuning is focused on solving an optimization problem through the application of a genetic algorithm [21]. This process involves the use of a set of initial parameters that are the ones obtained from manual tuning. The proposed genetic algorithm, complying with Darwin’s theory of natural selection, obtains the vector of parameters that optimally adjusts the filter to the set of situations tested. The genetic algorithm is designed to generate offspring in each generation using an elitist strategy and mutation operators.

In the context of this problem, each individual would be a vector of component parameter values of the matrices R and Q. For the filter adjustment, the filter is run for each set of parameters and fitness is estimated according to the error between the estimated states and the actual values of the states. This error is measured with the same error function discussed previously. The type of genetic operator applied to solve the problem is the mutation operator. With an elitist strategy, we avoid losing the best individuals from each generation by copying them to the next generation. In our case, the top 10% of the population.

In the assessment of the integrated system’s positioning error, the evolutionary algorithm is utilized to optimize its performance in various scenarios. This is essential since the system can encounter multiple factors that may affect the accuracy of its results. Among these situations, the following cases have been tested:

- Periods of GNSS signal absence e.g. due to urban environments.
- Different vehicle operations on predominantly curved and predominantly straight sections.

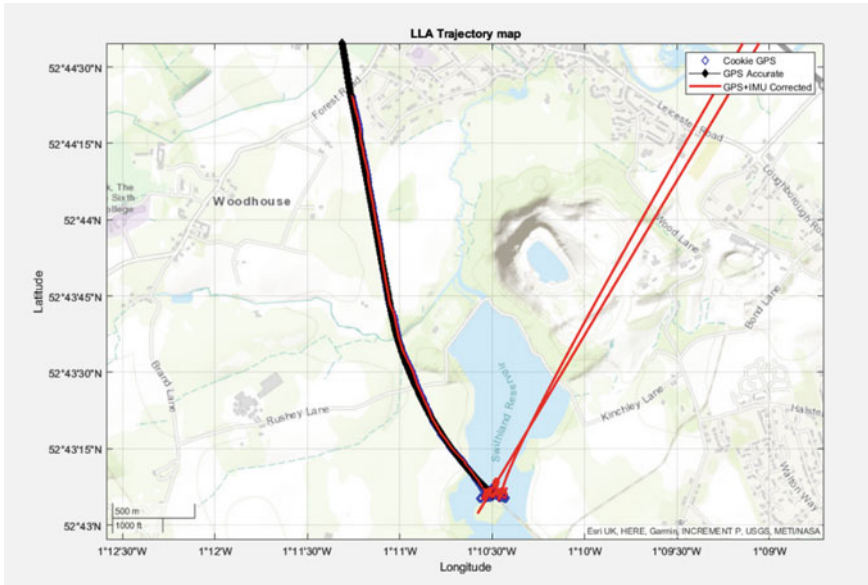


Fig. 13 Visualization of the IoT edge device and reference data collected in a railway

- Incorporation of ambient noise into the GNSS transmitted signal simulating the degradation of system accuracy caused by atmospheric changes, which affect the propagation of satellite signals.

The study evaluates several machine learning classification algorithms to identify the optimal Kalman parameter values for these different scenarios. In particular, a binary tree model is implemented to detect interruptions and noise in GNSS signals, achieving a success rate of 73%. Furthermore, the Naive Bayes method is utilized to detect vehicle maneuvers, with a success rate of 86% based on data obtained from the gyroscope and accelerometer. For all conditions, the obtained results show a reduction of the mean error of the trajectory with respect to the error without applying the Kalman filter (Fig. 13—Visualization of the IoT edge device and reference data collected in a railway).

With this filter and adjustment, a reduction in the mean error of $2.4117e-04$ degrees has been obtained.

4.3 AI Mechanisms for Train Integrity System (INDRA, UPM)

The train integrity process has been developed in the SCOTT project based on a combination of all the outputs provided by the positioning sensors (UWB, GNSS,

accelerometers) and the GNSS (Sect. 4.2) to generate a corrected estimation of the train integrity, following X2RAIL-2 [22–24] Shift2Rail project. However, this system does not consider the physical characteristics of the composition, which affects the calculation of the integrity. To improve the current system, this module processes the WSAW measurements (RSSI, Train Length, and Accelerometer) through an AI system to obtain a corrected and predicted measurement for the decisor.

The train integrity evaluator solution based on TBB2.1 and adopted in UC5.7 has been developed by INDRA and UPM.

The module requires the entry of different WSAWs such as the RSSI, train length and accelerometer. Additionally, other WSAW as UWB could be integrated as is shown in the Fig. 14. The evaluator gathers information from the edge and as a result, we obtain the weighted measurement that will serve as input for the decisor.

The integrity process is essential for the system developed in the T5.7. Currently, the integrity is calculated through the collection of the edge nodes information that indicates the position of the wagons. This module enhances the present process by applying AI mechanisms that provide a precise measurement of the train length and train integrity based on the weights generated by the evaluator and the KPIs defined for each service.

The train integrity decisor solution based on TBB2.1 and adopted in UC5.7 lead by INDRA in collaboration with the UPM. INDRA has overseen the requirements

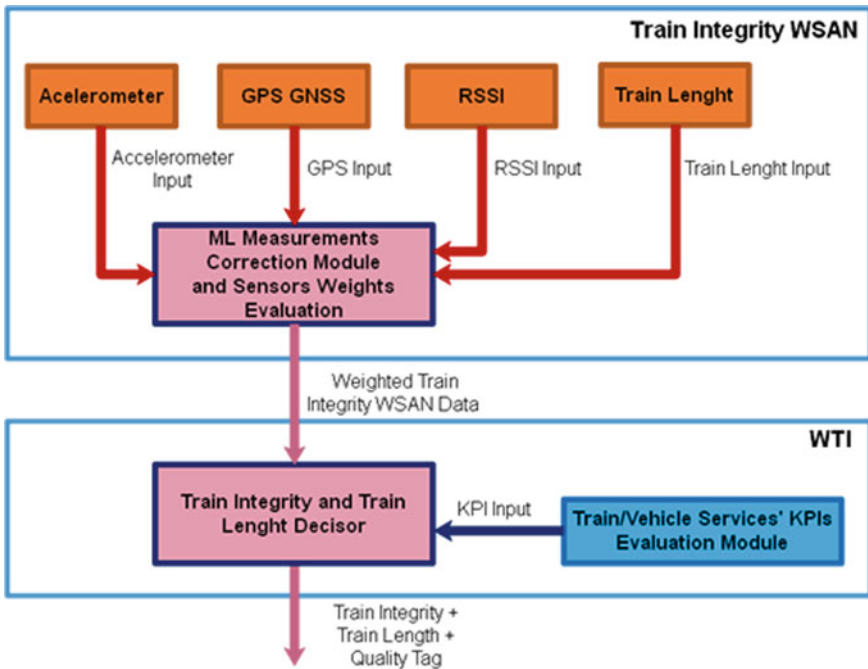


Fig. 14 Train integrity system architecture

definition and reference data provision, besides the execution tasks in a validation and verification layer. On the other hand, the UPM has contributed to the reference data and participated in the design and development tasks. Finally, INDRA has validated on real scenarios the different developments and implementation integrating the different submodules making use of X2RAIL-4 Demonstrator resources.

4.3.1 Results

To forward sensor reports, an optimized routing protocol over IEEE 802.15.4 for resource-constrained sensors is used, which enables the Wireless Sensor Network (WSN) to operate effectively even in unstable environments such as railway conditions. This protocol uses a multi-hop approach and adapts the network topology based on the RSSI value. This approach aims to reduce network overload by minimising the number of hops and ensures that the network can quickly reorganise in the event of node failure, thereby improving reliability.

All the data collected by the deployed WSNs is transmitted to the coordinators of the network, which serves as the anchor of the network and is located at the head of the train, which receives the packages and transmits the processed information to the central system of the train. The central system performs sensor fusion and provides the state of the composition. The proposed solution offers an effective and efficient approach to edge prototyping in the railway industry, but has been also identified as a potential solution for applications in other fields as well.

Communication between all the coordinators and the central system is established using IEEE 802.11 to enhance connectivity and robustness. Furthermore, to increase overall network robustness, two GNSS networks are deployed on both sides of the train, while two UWB WSNs are deployed between wagons. The nodes transmit at half the frequency of the coordinator to comply with the Nyquist-Shannon sampling theorem. This parameter can be configured in real time with configuration messages implemented in the custom protocol, along with other messages that enable power saving during non-operation windows and relative location message exchanges.

Prior to the integrity stage, it is necessary to determine the train composition, as each individual wagon may have distinct dynamic characteristics, such as weight, braking force or identification number. If the train composition remains constant, these parameters can be predetermined and fixed. Nevertheless, in cases of dynamic train compositions requiring wagon rearrangement, the use of WSNs can offer a flexible and economical solution to dynamically locate the composition's position. By implementing this approach, train compositions can be dynamically modified, leading to greater operational efficiency and adaptability.

This process takes place during the inauguration stage, which involves the identification of sensors that will transmit data during the integrity stage. At this stage, the position of the composition is determined based on the relative location of the nodes at that moment. Through the identification of transmitting sensors and their position, this process establishes the initial state of the composition, which is a prerequisite for the central system to accurately interpret the transmitted data.

To enable this relative positioning in environments where GNSS signals are not accessible and in static applications, a method that utilizes node sensor parameters that are intrinsically dependent only on the sensor node is proposed. To provide localization in outdoor environments without the need of dedicated hardware, this method utilizes RSSI and the topology of the network. For this use case involving trains, the linear topology of the train cars simplifies the localization problem by reducing its dimensions. Although RSSI is not typically a reliable parameter for accurate localization without dedicated hardware, in this particular case, the difference in length between train cars and the associated error in the position determined through the RSSI parameter provides enough accuracy for this relative localization problem.

The routing protocol utilized in this system is limited to only containing the parent MAC address and the number of hops for each node since it is designed for unstable environments. To obtain more information, custom messages are utilized, which include details such as the parent–child relationships between nodes, the number of hops along the route, and the RSSI value. By leveraging this additional information, the system’s coordinator can determine the relative position of each node within the composition, enabling effective localization.

The systems were deployed in multiple tests carried out in the United Kingdom railway (Fig. 15—Deployment of Cookie node during tests conducted in the United Kingdom), verifying their functionality. Through the testing phase, any potential issues were identified and resolved. Consequently, the systems were successfully deployed and tested in an empirical setting, ensuring their reliability.

The UWB module, compliant with the Cookie architecture, implements the advantages of this technology. The wide frequency band used is ideal for unstable environments, since signals are spread and can resist interferences, increasing its reliability. Furthermore, low power consumption is a characteristic commonly found in this technology.

To determine integrity, the distance between UWB node and coordinator is determined with Two-Way Ranging (TWR), a range-based technique that uses the time

Fig. 15 Deployment of Cookie node during tests conducted in the United Kingdom



of flight of the signals. Since this is done in a wide band, a high precision is obtained with this technology. The value is received by the coordinator of this WSN and sent to the central system, where it is merged with the other sensor parameters.

The WSN utilizing UWB modules is deployed in the inter-wagon area, where a direct line of sight can be established. The system provides measurement reports with centimeter-level precision. To assess its performance, two mirror coordinator-node pairs are placed at the same joint.

4.4 AI Mechanisms for Object Detection System for Railways (INDRA, UPM)

The system developed in the T5.7 is mainly focused on enhancing the cross-domain areas as level crossings. Nowadays, a system known as “*Trustable Warning System*” (TWS) has been developed in SCOTT project to secure the critical areas in the tracks based on the object detection. However, this detection is only supported by the 3D-LIDAR sensor, which detects a set of objects previously defined and its speed.

This module integrates the data produced by RGB cameras and the 3D-LIDAR sensors to provide a set of relevant features from the objects visible in the scene. Among these features can be highlighted the mass and estimated weight of the object, its area and speed of movement, as well as the spatial composition of all the objects in the scene. The system is based on machine learning techniques.

The object selector solution based on TBB2.1 and adopted in UC5.7 has been developed by Indra and UPM (Fig. 16).

In surveillance systems, especially in high-risk areas like railway level crossings, object detection and object classification are crucial technologies. These systems need to accurately detect, track, and classify any object that enters into the monitored area in real-time to prevent collisions or accidents in the future. To address this need, the proposed solution is an edge IoT hardware platform that can detect, track, and classify objects in a railway level crossing scenario. The system calculates and transmits its response from the IoT platform to the train, triggering a warning action to avoid potential collisions.

The main objective has been to develop a reliable surveillance system for railway level crossings using a low-resolution LiDAR object detection and classification system that uses a single sensor and that is suitable for integration into a custom IoT edge node. The main innovation lies into the integrated implementation of LiDAR data management. To achieve this, a custom hardware platform including a Google Coral neural Accelerator and AM Cortex A5 processor has been used as the processing core element due to its trade-off between cost and performance.

To optimize computational resources and focus only on relevant areas, the object detection system first eliminates points outside the region of interest, which is the level crossing area. The frame is then projected onto a 2D plane in order to further optimize resource usage. As this is a static application, a golden background is stored

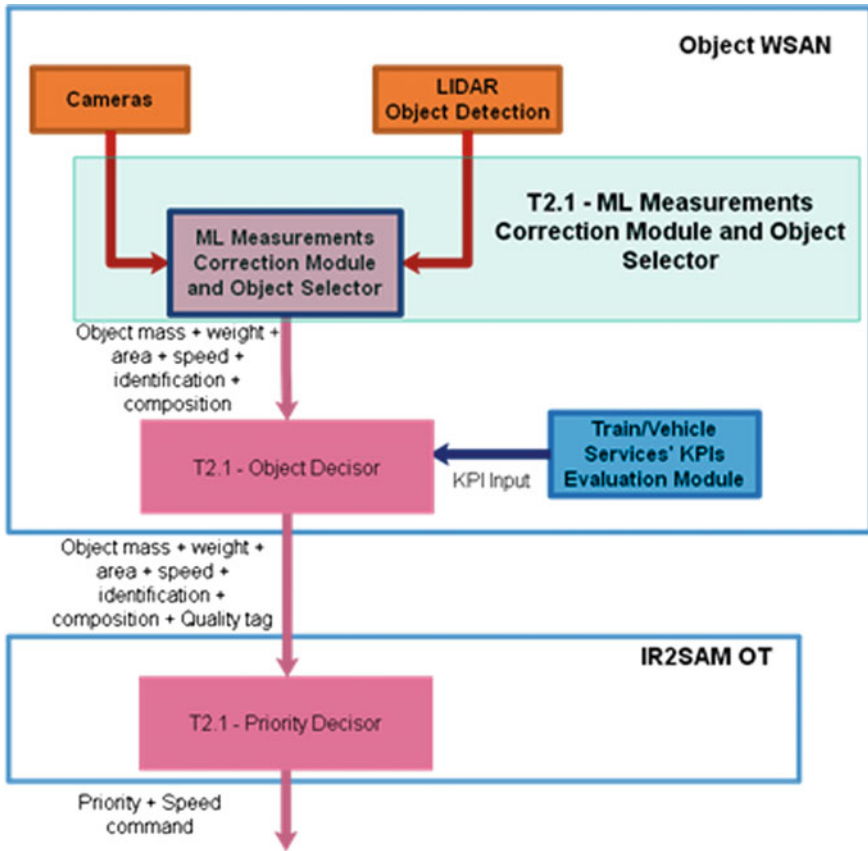


Fig. 16 Object detector system architecture

for frames with no moving objects, and this background is subsequently subtracted from the following frames. Object detection is then carried out on these areas to obtain bounding boxes of the objects that cluster the points that are used in the classification stage. Once any object is detected in the level crossing, it is tracked to estimate its location and speed. These parameters are required to decide whether the train must be stopped for safety reasons. By using this method, the system meets the requirements for both explainability and real-time performance.

During a second stage of the process, the detected objects are classified using the VoxNet DNN. This particular DNN provided the lowest processing time along with high accuracy, being the best solution to use in edge systems with point clouds. The points of each detected object are transformed into a voxel grid format before being fed into the DNN. The DNN has been trained with synthetic data captured in a simulation of the level crossing and fine-tuned with the real data. In this way, the DNN learns to recognize object features with a substantial volume of data that has been automatically generated and labeled in the custom simulated environment.

Subsequently, through the fine tuning process, it learns the specific details of the real data that were captured during deployment. The intensive processing involved in this stage is executed by the neural accelerator Coral EdgeTPU chip, which is incorporated into the custom IoT edge node.

4.4.1 Results

Table 2 illustrates the evaluation outcomes. The sensitivity value presented in the results is associated with the FN parameter, which is crucial for safety reasons in the proposed use case. A reduced rate of FN must be matched by a high value of F1, since it correlates the TP, FP, and FN performance metrics. Finally, the value of the specificity has also to be considered because it relates TN with FP rates.

4.5 AI Mechanisms for Adaptative Coupling Distance Control (INDRA, UPM)

To improve the virtual coupling system, two different modules has been developed: the real time monitoring and QoS control for platoon strategy and the real time monitoring and QoS control for vehicle.

The first one consists of the vehicle model fine-tuning that refines on real time the specific dynamic train characteristics model.

Figure 17 shows the connection between the monitoring module and components for the vehicle model characterization.

The second one consists of making use of the vehicle model fine-tuning described on the previous paragraph, to adapt on real time the movement directives to the specific dynamic train characteristics.

Figure 18 shows the connection between the monitoring module and components for the vehicle model characterization.

Both train dynamic characterization module and virtual coupling smoothing control have been defined and designed by INDRA in collaboration with UPM to

Table 2 Object detector results 1

	Object detection		Object detection and classification
	Fine exploration	Complete evaluation	
Evaluated frames	1800	23,417	3000
Sensitivity	98.65%	99.46%	96.43%
Specificity	98.68%	98.85%	96.65%
Precision	98.68%	98.70%	94.64%
F1	98.65%	98.93%	95.53%

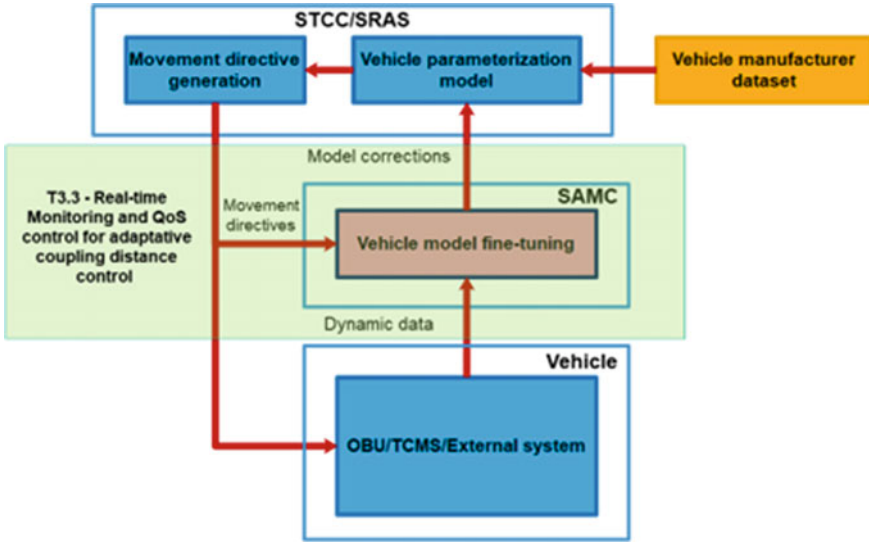


Fig. 17 Adaptative coupling distance control architecture 1

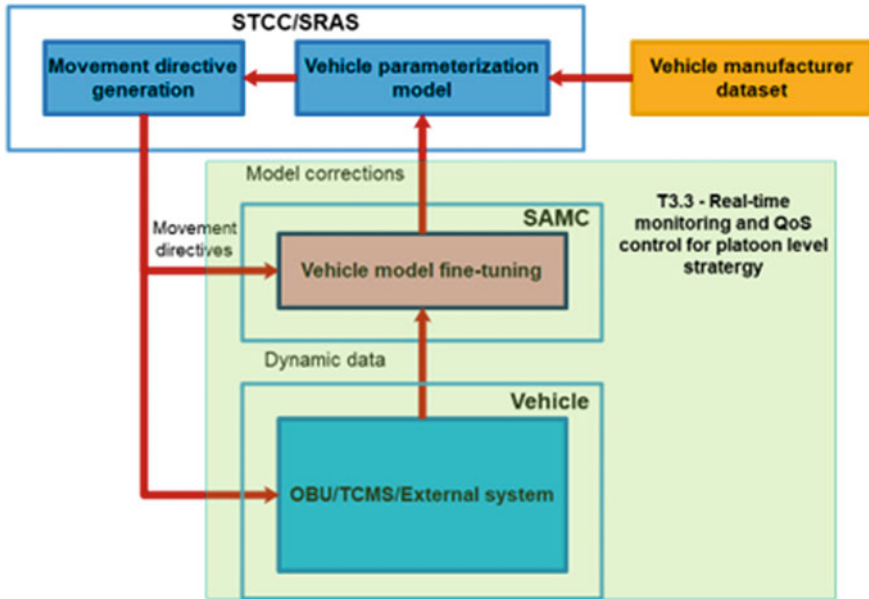


Fig. 18 Adaptative coupling distance control Architecture 2

integrate them on previous existing works developed and implemented by INDRA with the new modules to improve both tactical and operational virtual coupling layers.

4.5.1 Results

In the field of virtual coupling, UPM has worked in different areas. Firstly, the train dynamic characteristics establishing different categories have been parametrized. Up to date, three different categories have been considered: metro, intercity and high-speed. For each of them, the technical data that involves the dynamics of the train has been collected, with the objective of defining a “standard” train for each category. This train will be used in the different simulations to study and design the virtual coupling control between trains.

Secondly, UPM has parametrized three different lines, one for each category. A metro, an intercity and a high-speed line defining speed limitation, slope and curve radius have been modelled. These values correspond to real railway lines.

Thirdly, for each category, a nominal virtual coupling control to analyze several aspects has been implemented:

- Control strategies. The leader movement is defined by a standard tracking control of a predefined velocity profile. For the followers, several strategies have been studied: minimize the distance with the front train, minimize the relative velocity between trains, maximize the follower velocity, and combinations between them (for example, minimize the distance between trains and minimize the relative velocity between them). All these strategies have been studied with the objective of maximizing line capacity: maximum number of trains at the same time and minimum trip time.
- Security constraints. A security distance between the trains that must be respected all the times has been considered. This distance depends on the velocity of each train and on the braking deceleration of each train. Different situations from the most conservative to the most aggressive have been considered, analyzing the effect of the convoy behavior.
- Performance issues. The different parameters involved in the controller have been studied to reduce the computation time. UPM has studied the effect of increasing the prediction horizon and the discretization time to obtain the better controller performance with accurate results.
- Smoothness analysis. In the same way, the effect of different prediction horizons and different states predictions in order to have the smoothest behavior of the coupling have been studied.

After that, Indra has continued with the development of a robust control strategy for the coupling considering the uncertainties that can uncertainty appear in the train location, delays in communication and in the application of the traction/brake forces, line conditions: adherence and environment conditions as wind and tunnel factors, train characteristics: mass and aerodynamic drag. References [25] and [26] present the developed control system.

This model has been tested in a MatLab environment with three different categories: metro, intercity and high-speed. For each one of them, we have collected technical data that involves the dynamics of the train, with the objective of defining a “standard” train for each category. These trains have been used in the different simulations to study and design the robust controller for virtual coupling control between trains.

A comparative study has been also done between the nominal controller previously defined and the new robust controller, showing the differences between them.

Finally, the developed SW has been implemented to be used in the INDRA simulator where the virtual coupling module has been implemented through the SAMC module within the overall architecture of the simulator where the test cases have been tested and validated.

The control model was tested with a situation in which the convoy operationally behaves as a single train since, from the users’ point of view, all the virtually coupled train set components arrive at the station at the same time and behave as if they were a single complete train. Although when the convoy is running, the distance between the components increases due to the safety conditions which have been set, we consider that the convoy preserves its integrity, operating as a single train, when the communication between trains is maintained and the above-mentioned condition of all the components arriving at the station at the same time is respected. The figures included in the following simulations illustrate this phenomenon.

Some examples of the simulations done in a metro line are included, in Fig. 19—Slope and radius considered in the simulation scenario. and in Fig. 20—Maximum driving speed in compliance with speed limits.

Figure 21 represent the behavior of the convoy when the follower experiences a 10% loss of adhesion during braking, comparing different controllers.

In this metro line, several simulations have been developed to test the controllers. As an example, the behavior of an uncertain variable that corresponds to a disturbance involving a loss of adhesion at the entrance of the third station. This situation is extremely unfavorable since the disturbance was applied only to the follower, and the train in front was considered to continue braking without any loss of adhesion.

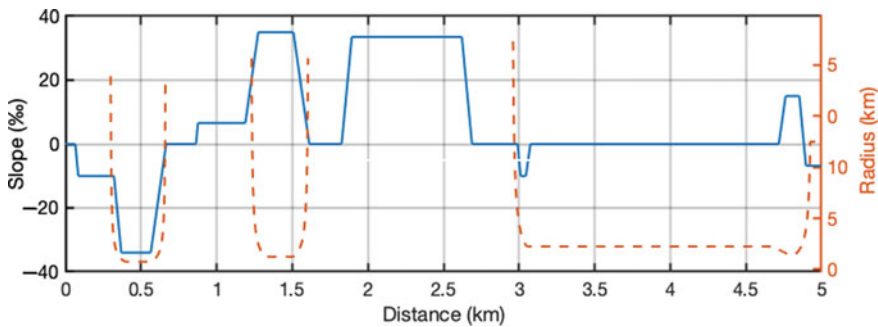


Fig. 19 Slope and radius considered in the simulation scenario

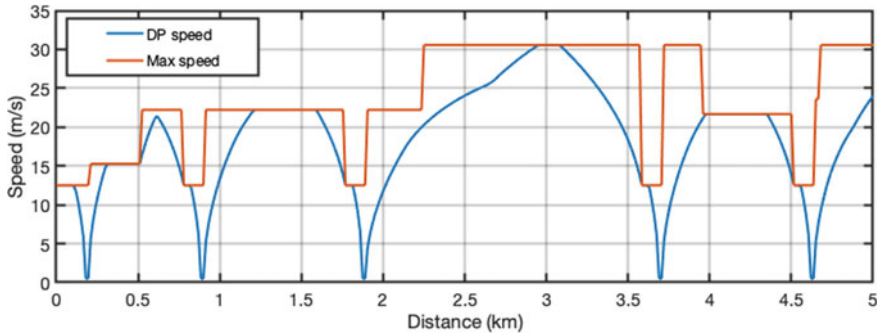


Fig. 20 Maximum driving speed in compliance with speed limits

Figure 21 represents the time–velocity curve and shows the speed of the leader, which is the same for both the nominal and robust controllers. Moreover, in acceleration (i.e., the increasing parts of the curves), the leader has a slightly higher velocity than the followers, thus increasing the distance between the trains. However, in braking (i.e., the decreasing part of the curve), when the velocity of the leader decreases, the follower becomes closer as it has a higher speed, always maintaining the safe distance imposed by the constraints. It is also possible to observe the stops at the stations.

In this figure, it can also be seen how the integrity of the convoy is maintained, as the two trains stop at the station at almost the same time. This effect can be seen in the time–velocity plot, where both trains in the convoy stop at the stations at almost the same time. The plot below shows the distance between the leader and the follower. Here, the most critical moment occurs at the entrance of the stations, where the leader stops, and the follower must also stop maintaining a safe distance.

The curves labelled “10% adhesion loss” include the distance between the leader and the follower for the NMPC and RMPC when there is no disturbance, i.e., no adhesion loss. These curves show a very similar behavior in both cases.

INDRA, based on previous works developed on SCOTT, has adapted the Virtual Coupling system to the system architecture defined on X2RAIL-3, including as a module the developments achieved by UPM on the Operational and strategic layer, validating the changes on the physical platforms making use of the simulator environment deployed at Indra premises covering the different use cases defined on X2RAIL-3.

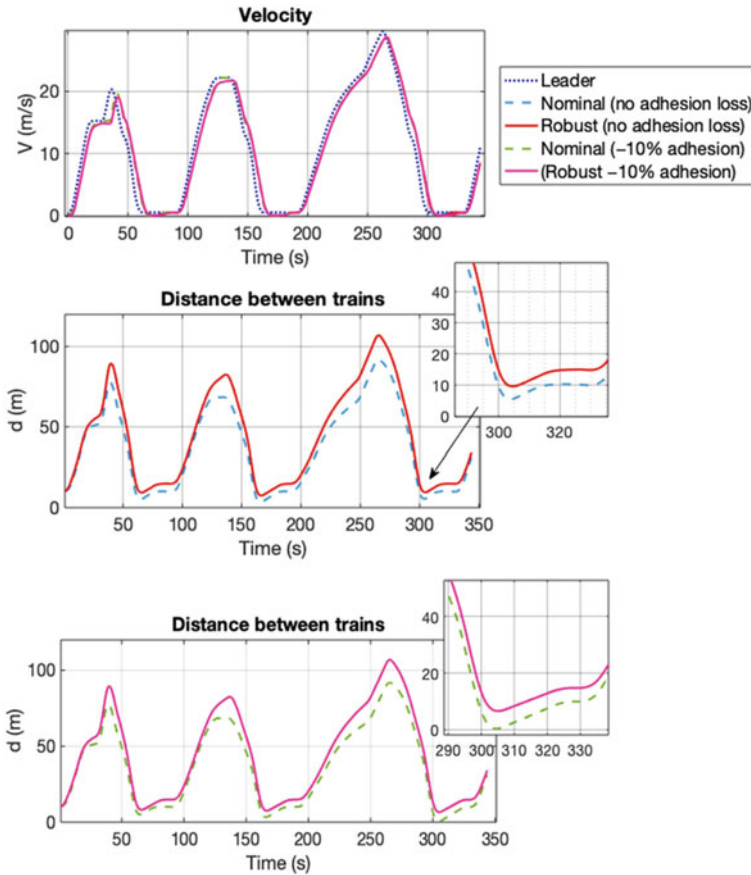


Fig. 21 Results of simulation 1: 10% adhesion loss. Variables versus time

4.6 Security Mechanisms in IoT Deployments in the Railway Domain (UPM)

The energy consumption constraints of the IoT edge devices are very restrictive when these ones are powered from batteries, which usually leads to using low-performance hardware with the aim of improving autonomy. These restrictions may be incompatible with the implementation of common cryptographic protocols that consume a non-negligible number of resources, making it easier for an attacker to gather data or get the control over the network.

To achieve a robust IoT application in terms of security, three main aspects must be considered:

- Confidentiality by means of encryption of sensitive data being transmitted along the network.

- Integrity, such as being able to check whether an attacker could have tried to modify any data.
- Authenticity of data, allowing to verify if the received data comes from who was supposed to have it.

With these points in mind, a common approach for securing communications in a network can be simplified in three steps:

1. Authenticate a user/node in the network prior to sending or requesting any sensitive data.
2. Establish a secure communication channel by exchanging some kind of shared secret, e.g., keys.
3. Use the previously shared secret to exchange sensitive data and verify its integrity.

Any of those steps can be made with symmetric or asymmetric schemes, depending on the requirements of the given application.

In addition, IoT deployments present an extra attack surface related to how easy it might be to access the nodes physically. Being this the case, a wide spectrum of side-channel attacks could be performed on the nodes, making it possible for an attacker to monitor leakage of information from power consumption or electromagnetic radiation.

This problem of securing IoT networks is a well-known one, and many solutions have been proposed [27]. A common approach is to speed up cryptographic tasks in hardware, thus reducing their impact on time and energy consumption. This idea is widely spread in the industry, where it is easy to find new microcontrollers targeting IoT designs that incorporate peripherals implementing commonly used ciphers or cryptographic operations. There are also commercial products that aim to solve the problem with side-channel attacks, such as the ATECC608B [28], which serves as a hardware accelerator but also as a trusted platform module storing data securely.

However, several new threats are being addressed due to the growth of technologies for quantum computing realization. There exist quantum algorithms that can solve efficiently the mathematical problems in which some classical cryptographic schemes are based on. Peter Shor's algorithm [29] can break not only schemes based on the large integer's factorization problem, such as RSA, but also the ones whose security relies on the discrete logarithm problem. This means that elliptic-curve cryptography (ECC) also becomes insecure, being this the preferred one on IoT networks because of the lower key sizes compared to RSA.

Regarding symmetric cryptography, Grover's quantum search algorithm [30] provides a quadratic speed up on brute force attacks on the key space. Nevertheless, this attack still presents an exponential complexity in execution time, thus, doubling the size of the keys would keep the same level of security.

Using symmetric schemes is, however, not recommended for all scenarios. The need for all the nodes in the network to share the same secret key becomes a problem when the communication channels cannot be trusted at all, and the distribution of this key material could be a point of attack. Therefore, new proposals have been made to keep using asymmetric schemes in the future presence of quantum computers.

In particular, post-quantum cryptography (PQC) aims to provide new asymmetric schemes whose security is based on mathematical problems that cannot be solved by known quantum algorithms but without the need of access to quantum resources (in contrast to quantum or semi-quantum alternatives). Thus, these schemes might be a possible solution to IoT security.

The interest in the study of PQC is such that the National Institute of Standards and Technology (NIST) proposed a call for standardization of some of these schemes. The final candidate for standardization in the key exchange mechanism (KEM) category was CRYSTALS-Kyber [31], which is a scheme based on the hardness of certain problems over lattices. The call for standardizing digital signature algorithms (DSA) is still running, being the three finalists CRYSTALS-Dilithium [32], FALCON [33] and SPHINCS + [34], being the first and the second also lattice-based cryptosystems, while the third one provides hash-based signatures.

The problem with PQC algorithms is that they tend to consume more resources than the ones used nowadays, since they present larger key and ciphertext sizes and/or execution times. Thus, the constraints imposed on IoT networks become even more restrictive if we want to keep acceptable security levels in the future. Works such as [35] and [36] show a comparison between NTRU (another lattice based PQC scheme proposed as finalist in the NIST's KEM standardization process), RSA and ECC. More focused on popular communication protocols, [37] presents an implementation of NTRU encryption along with the MQTT protocol.

The problem with the research related to PQC is that most works are focused on high-performance devices, such as processors commonly used in desktop-level applications or powerful FPGA architectures. The NIST defines the ARM Cortex-M4 processor as the reference for low-performance devices, but this is not the only architecture used on many IoT edge implementations, where it is easy to find simpler devices with ARM Cortex-M0 or AVR architectures. Thus, there is a need to study how these PQC schemes could be implemented within the heterogeneity of products available for IoT designs.

5 Conclusions (INDRA)

During this project, the decision-making system used in critical points along the railway lines has been enhanced using existing technologies and new ones. The integration of existing technologies and new ones for several services (i.e., decision-making system in shared areas, enhancing and smoothing virtual coupling management or improving functionalities such as speed control, timing control of the stops, positioning or Train integrity monitor), has to be the basis of the work to be performed during the project. For that purpose, the main focus has been on:

- Implement AI mechanisms for **improving Absolute and Relative Train Positioning** to increase rail lines capacity reinforcing safety conditions.

- Implement AI applications for **improving Train composition process and Train Integrity Monitoring** to automate the process of train inauguration and to improve Train Integrity calculus to be more reactive to a train integrity lost and consequently be able to increase the rail lines capacity.
- Implement **AI mechanisms to improve object detection systems** to increase the safety on critical areas (level crossing, crossing) but to obtain additional information for the different perception systems thinking about the future needs of the implementation of an ATO GoA 4 system
- Implement AI applications for **improving virtual coupling train** movements and distances adjustment in order to increase passenger comfort and cargo trust

This will significantly impact the European Union to achieve the full potential of the IoT and the Artificial Intelligence in the Railway domain. The wireless communications developed in SCOTT project set the basis for helping to reduce time and costs, making the development and management of rail infrastructures and train compositions smarter and more efficient.

Although SCOTT project scope tackled several issues of the railway domain in terms of Internet of Things and wireless technologies investigation, the functionalities developed in that project could be improved moving the focus to the passenger's comfort and cargo security, supplying to the SCOTT systems Artificial Intelligence and a better allocation of the control resources.

The development of innovative solutions for movement control, automatic operation, management of multimodal areas and distributed rail signalling systems will give the European Rail Supply Chain a competitive advantage in the worldwide market and will generate new business opportunities for the European Railway Industry.

The works performed in InSecTT project are fully aligned with the innovation and the transformation that have been carried out in the Shift2Rail Innovative Programmes. Several technologies investigated in S2R have been improved with the InSecTT innovation topics as IoT and AI setting the bases for ERJU new projects. It is important to remark that the use of AI technologies is just a first step in the integration of these technologies in the railway domain, deeply linked and aligned with S2R baselines and programmes. Several examples of the issues studied in S2R that InSecTT can improve to make railway domain more competitive are:

- The wireless technologies used to enhance the communication between the different objects and signalling systems
- Train determination techniques for the solution of a train coupling, making the manoeuvres more comfortable and safe for both passengers and cargo.
- Use of the freight rail innovative solutions to provide a real time management of the cargo in the railway domain, adding the connectivity with other domains thanks to the connection with port facilities.

In this way, the use of Artificial Intelligence on transportation for both Control and Management will improve the rail and road capacity, decreasing the delays

and improving the plan, making the transportation experience more comfortable for passengers.

By connecting all-to-all, increasing the level of automation, decentralizing the decisions and enhancing the efficiency of the system, it is possible to enhance the current state of the railway transportation, having an important impact in the domain market.

As it stated in the Rail Use Cases description, the main objectives described above are covered.

Gathering all the topics explained above, the expected impacts of InSecTT project concerning the Railway domain are:

- Increase the safety of the current coupling processes, making the solution highly competitive and effective in secure and safe related environments.
- Adapt the speed change processes to improve passengers comfort and trust and cargo security by using Artificial Intelligence mechanisms to control the input parameters and smoothing the speed and acceleration/deceleration variations.
- Provide an improvement of Virtual Coupling manoeuvres via Artificial Intelligence by improving the smooth in the speed change processes in order to enhance passengers comfort as well as transport companies trust in this type of virtual composition.
- Improve the current state of the system to control coupling/uncoupling processes in a safe and secure way.
- Convert conventional lines into ATO lines in a safe and secure way, by means of all-to-all connection in a decentralized environment.
- Demonstrate new signalling systems based on wireless communications that make possible to connect On Board and On Track stakeholders in a distributed and decentralized environment
- Enhance the management of cross-domains areas, specifically rail and road areas and port facilities, focusing on the multimodal jams management.
- Achieve a more efficient multi-domain traffic management by allowing to share the vehicles and trains information and distributing it using decentralized systems. In addition, the system can integrate vessels to the network for cargo management.
- Solve the safety deficiencies in current systems with the purpose of reducing the number of injures and human losses as well as the damages on wagons, cargo and railways infrastructures.
- Improve the use of multipurpose WSN applying artificial intelligence for the safety and non-critical systems installed on vehicles, trains and multimodal infrastructures, such as railway connected ports.
- Include distributed solutions to efficiently manage the exchange of information and data distribution through a decentralized system, which controls specific areas and the wayside objects and signalling system, allocated within that area.
- Increase punctuality of operating lines reducing time and enhancing the timetable management in an automatic way. It improves passengers and end-users experience, making the services more trustable.

- Reduce costs and time concerning coupling services and control of coupled compositions, allowing an efficient management of the track capacity.
- Enhance energy efficiency of current railway systems, reducing the environmental impact.
- Connect devices and manage them in an intelligent way, increasing the level of automation and enhancing the efficiency of the system.
- Add flexibility to the routes and tracks management, by allowing several routes operating at the same time, controlled by an autonomous system, which makes use of Artificial Intelligent methods.

References

1. Shift2Rail Joint Undertaking: Mission and Objectives. <https://rail-research.europa.eu/about-shift2rail/mission-and-objectives/>. Accessed March 2023
2. ECSEL Join Undertaking Objectives: <https://www.kdt-ju.europa.eu/ecsel-ju-what-we-do>. Accessed March 2023
3. SCOTT “Full Project Proposal (Technical Annex)”, 2017-05-01
4. Panetta, K.: Gartner Top 10 Strategic Technology Trends for 2019, 2018-10-15
5. Observatorio de Transporte y la Logística en España, “Informe Annual 2017”, March 2018, pp. 208–226
6. SCOTT D20.1 “Critical Area Trustable Warning System Requirements Specification”, v1.0, 2018-01-24
7. SCOTT D19.4 “Smart Train Composition Coupling demonstrator”, v1.0, 2021-03-24
8. X2RAIL-3 D6.1 Virtual Coupling System Concept and Application Conditions, v1.0, 2020-01-15
9. DEWI. D311.007 “Reference Architecture Summary”, v1.0 2016-07-19
10. FIWARE. About it. <https://www.fiware.org/about-us/>. Accessed March 2023
11. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**, 436–444 (2015)
12. Perera, C., Qin, Y., Estrella, J.C., Reiff_Marganec, S., Vasilakos, A.: Fog computing for sustainable smart cities: a survey. *ACM Comput. Surv.* **50**(3), Article 32 (2017)
13. Magno, M., Pritz, M., Mayer, P., Benini, L.: DeepEmote: towards multi-layer neural networks in a low power wearable multi-sensor bracelet. In: 2017 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI), pp 32–37 (2017)
14. Edge TPU: <https://cloud.google.com/edge-tpu/>. Accessed March 2023
15. Ramchurn, S.D., Huynh, T.H., Jennings, N.R.: Trust in multi-agent systems. *Knowl. Eng. Rev.* **19**(1), 1–25 (2004)
16. Yeoh, W., Yokoo, M.: Distributed problem solving. *AI Mag.* **33**(3), 53–65 (2012)
17. X2RAIL-1. D3.3 “Specification of the Communication System and Guideline for Choice of Technology”, v1.1, 2019–01–29
18. X2RAIL-3 D3.3 “Adaptable Communication System Field Test Strategy”, v1.0, 2020-09-29
19. Merino, P., Mujica, G., Senor, J., Portilla, J.: A modular IoT hardware platform for distributed and secured extreme edge computing. *Electronics* **9**(3) (2020)
20. Li, D., Jia, X., Zhao, J.: A novel hybrid fusion algorithm for low-cost GPS/INS integrated navigation system during GPS outages. *IEEE Access* **8**, 53984–53996 (2020)
21. Bell, J., Stol, K.A.: Tuning a GPS/IMU Kalman filter for a robot driver. In: Australasian Conference on Robotics and Automation (2006)
22. X2RAIL-2 D4.1 Train Integrity Concept and Functional Requirement Specification v5.0, 2020-06-23

23. X2RAIL-2 D4.2 Functional architecture & Interfaces specifications & Candidate technologies selection v3.0, 2020-09-04
24. X2RAIL-2 D4.3 Test scenarios, test cases and test procedures definition v2.0, 2020-11-02
25. Felez, J., Vaquero-Serrano, M.A., de Dios Sanz, J.: A robust model predictive control for virtual coupling in train sets. *Actuators* **11**, 372 (2022). <https://doi.org/10.3390/act11120372>
26. Vaquero-Serrano, M.A., Felez, J.: A decentralized robust control approach for virtually coupled train sets. *Comput.-Aided Civ. Infrastruct. Eng.* **00**, 1–20 (2023). <https://doi.org/10.1111/mice.12985>
27. Mohamed, A.M.A., Hamad, Y.A.M.: IoT security: review and future directions for protection models. In: *Proceedings of International Conference on Computer and Information Technology (ICCIT)*, pp. 1–4 (2020)
28. “CryptoAuthentication™ device summary,” Data Sheet ATECC608A, Microchip, Chandler, AZ, USA (2020)
29. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26**(5), 1484–1509 (1997)
30. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: *Proceedings of 28th Annual ACM Symposium on Theory of Computing*, New York, NY, USA, pp. 212–219 (1996)
31. Bos, J. et al.: CRYSTALS—Kyber: a CCA-secure module-lattice-based KEM. In: *Proc. IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 353–367 (2018)
32. Ducas, L. et al.: CRYSTALS-dilithium—algorithm specifications and supporting documentation (2017)
33. Fouque, P.-A. et al.: FALCON: fast-fourier lattice-based compact signatures over NTRU (2019)
34. Hülsing, A. et al.: SPHINCS+ submission to the NIST post-quantum project (2019)
35. Nandanavanam, A., Upasana, I., Nandanavanam, N.: NTRU and RSA cryptosystems for data security in IoT environment. In: *Proceedings International Conference Smart Technologies Computer Electrical, Electronics (ICSTCEE)*, pp. 371–376 (2020)
36. Upasana, I., Nandanavanam, N., Nandanavanam, A., Naaz, N.: Performance characteristics of NTRU and ECC cryptosystem in context of IoT environment. In: *Proceedings of IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, pp. 23–28 (2020)
37. Agus, Y.M., Murti, M.A., Kurniawan, F., Cahyani, N., Satrya, G.: An efficient implementation of Ntru encryption in post-quantum Internet of Things. In: *Proceedings of 27th International Conference on Telecommunications (ICT)*, pp. 1–5 (2020)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Innovative Solutions for Maritime Infrastructures Monitoring and Protection



Francesco Pacini, Daniele Marroccella, Antonio Lagudi, Manuel Fortunato Drago, Francesco Buffone, Floriano De Rango, and Fabio Bruno

Abstract The global economy heavily relies on maritime commerce, commonly known as the “Blue Economy,” which involves the transportation of raw materials and products via maritime traffic and the utilization of marine resources, including oil, gas, and underwater mines. The importance of various infrastructures linked to the sea, such as fish farms, pipelines, underwater power and data cables, cannot be overstated as they have significant impacts on both local and global economies. Therefore, safeguarding these infrastructures against both symmetric and asymmetric threats is crucial to prevent disruptions to vital services and avoid negative impacts on the economy and quality of life. This book chapter presents an underwater access control system designed to monitor and prevent unauthorized access to port areas from the seaside to safeguard maritime infrastructures. The system consists of acoustic and magnetic barriers connected to a wireless communication network that can collect and transmit data to the infrastructure control centre. The system provides robust, dependable, and secure solutions for near real-time data transfer, enabling continuous monitoring and prompt response to potential threats. This chapter provides a detailed description of the system architecture and its primary components, focusing on the technological aspects and innovative solutions used to meet the proposed requirements.

F. Pacini (✉) · D. Marroccella
Leonardo S.P.A, Electronic Division—L.o.B. Underwater Armaments and Systems, Livorno/
Pozzuoli, Italy
e-mail: francesco.pacini@leonardo.com

A. Lagudi · M. F. Drago · F. Bruno
Department of Mechanical, Energy and Management Engineering (DIMEG), University of
Calabria, Arcavacata di Rende, Italy

F. Buffone · F. De Rango
Department of Informatics, Modelling, Electronics and System Engineering (DIMES), University
of Calabria, Arcavacata di Rende, Italy

1 Introduction

The Blue Economy is crucial for the growth of human society as 70% of the Earth's surface is covered by water. About two-thirds of the world's population live near the coasts and benefit from seafood and raw materials from the ocean, as well as goods and services transported by sea.

There are various infrastructures that support the Blue Economy, both globally and locally. For instance, sea highways are routes used by cargo ships to transport goods and services. Thanks to a network of over 750 harbours, around 90% of all goods and materials are shipped by sea among continents. In 2020, there were 15 harbours with an annual traffic of over 10 million TEU (Twenty-foot Equivalent Unit), which is the ISO standard measurement unit for container volume. Similarly, all 49 major hubs worldwide had traffic higher than 4 million TEU each. Touristic traffic is also crucial in some parts of the world, with at least ten harbours having touristic traffic higher than 1.7 million passengers per year on average in the last 5 years [1].

Another essential infrastructure supporting the Blue Economy is the network of oil and gas pipelines that distribute energy sources. Despite the shift towards green energy, these pipelines will continue to refuel many countries for several decades. Moreover, the world network of underwater cables for information and data transfer is a strategic asset distributing information and data in near real-time to manage and control various aspects of daily life. In 2005, more than 200 cables were deployed worldwide, with an estimated total length of around 1.3 million km [2]. This infrastructure is the backbone of the digital economy, which will continue to expand to meet the increasing demand for data sharing. Finally, lithium and other rare-earth materials form the basis of new electronic and electric technologies, and the search for new mines and deposits is moving to the sea floor to increase their availability. The race for these elements will be strategic for the green transition and the autonomy of nations and the EU, with significant investments and efforts being made in this direction.

Monitoring and protecting infrastructures is important from both civilian and military perspectives. Different types of threats exist based on their potential effects on infrastructures themselves. Nations and national entities are interested in strategic to tactical infrastructures to push other nations and/or force their position in some contexts. For example, stopping oil/gas flow can create dramatic impacts on a regional economy with extension to the entire world. Another example is monitoring data flows accessing underwater cables to spy and get sensitive information to predict entities' actions and/or manipulate governments [3].

Threats can be listed considering the increasing level of danger: at the lowest level, Intelligence Surveillance Reconnaissance (ISR) missions aim to acquire information about infrastructures to plan future attacks/reactions. These activities in a maritime environment are today simplified by the availability of autonomous/remotely controlled underwater and surfaced vehicles. These platforms can be relatively cheaper, with quite long endurance, fitted with different payloads to acquire

data (movies and pictures by optical cameras, acoustic panorama by sonar, etc.). Due to their small dimensions and low noise print, it is difficult to detect and identify them.

An intermediate dangerous activity is the theft of raw materials, which often involves damaging infrastructure, such as underwater pipelines, and can lead to environmental damage. In some countries, this activity is typical and can result in oil spills from terminals, causing further harm to the environment.

The most dangerous threats involve military forces damaging strategic infrastructure such as pipelines, cables, terminals, power plants (including wind farms), and other critical structures. To carry out these actions, divers require support platforms such as Swimmer Delivery Vehicles or mini submarines. In the future, autonomous underwater vehicles may also be used for this purpose. These platforms can operate autonomously and damage infrastructure in several ways, including deploying explosives, using robotic arms, or exploding themselves near the target. In this context, the main targets that need to be identified and detected include divers, autonomous underwater vehicles (AUVs), remotely operated vehicles (ROVs), mini submarines, midgets, unmanned surface vehicles (USVs), and small crafts [4].

To effectively monitor and control maritime infrastructure, new technologies are required that are specifically designed for the underwater domain. This is because the physics of the water present unique challenges, such as high pressure that requires materials with high strength, poor visibility that limits the use of cameras, and underwater communication that is limited to acoustic channels with minimal bandwidth and throughput. Additionally, the deployment and maintenance of systems in the underwater domain are complex, and a large number of sensors are required to cover the vast area of the water column, making the solution costly compared to surface or aerial solutions.

As part of the InSecTT project, innovative applications and solutions have been explored for monitoring and protecting maritime infrastructures for both military and civilian purposes. The aim has been to improve the state of the art and make underwater protection systems more affordable and efficient.

2 Underwater Technologies for Monitoring Maritime Infrastructures

Monitoring and protecting maritime infrastructures is a challenging task that requires advanced technologies and systems. One of the most commonly used technologies is the deployment of surveillance cameras [5]. However, cameras have certain limitations in underwater environments due to poor visibility and the need for special housings to protect them from water pressure. Other technologies that have been explored include sonar systems, acoustic sensors, and radar [6]. More recently, AUVs and ROVs have become increasingly popular for monitoring and inspecting underwater structures [7]. These vehicles are equipped with various sensors and cameras

that can capture high-resolution images and collect data on underwater structures and environments. In addition, AUVs and ROVs can be operated remotely, reducing the risks associated with underwater operations.

The size of underwater areas and infrastructures of interest is typically large, making it impractical to rely on a single sensor for monitoring. Instead, a network of nodes forming underwater barriers is needed to adequately cover the area, depending on its size and shape [8]. These systems are used to provide continuous coverage and real-time data transmission. These networks can consist of acoustic, optical and magnetic sensors, as well as communication systems that can transmit data to shore-based stations.

Figure 1 illustrates the concept of using underwater barriers, which consist of a network of sensors (either acoustic, magnetic or a combination of both) to monitor the water column and the surrounding volume. Two homogeneous barriers are depicted in the top picture, with red dots representing magnetic sensors and black dots representing acoustic transducers. These barriers are connected to a junction box composed of groups of nodes. The bottom picture shows the concept of a mixed barrier, where magnetic and acoustic nodes are connected to the same line.

The approach of using a network of nodes is highly adaptable, expandable, and customizable, as it allows for the incorporation of various functionalities and capabilities by simply adding more nodes with different payloads. These nodes can operate in parallel or in conjunction with one another to provide a comprehensive dataset. The inclusion of intermediate processing nodes, akin to edge computing, allows for the fusion of data and the extraction of the most crucial information from a vast dataset. This is especially valuable in the case of barriers composed of a significant number of acoustic sensors. The integration of various data sources makes it possible to highlight information and behaviours that may not be apparent in a single dataset [9].

Different types of infrastructure have unique characteristics that require specific monitoring and protection solutions. However, it's important to have a flexible and

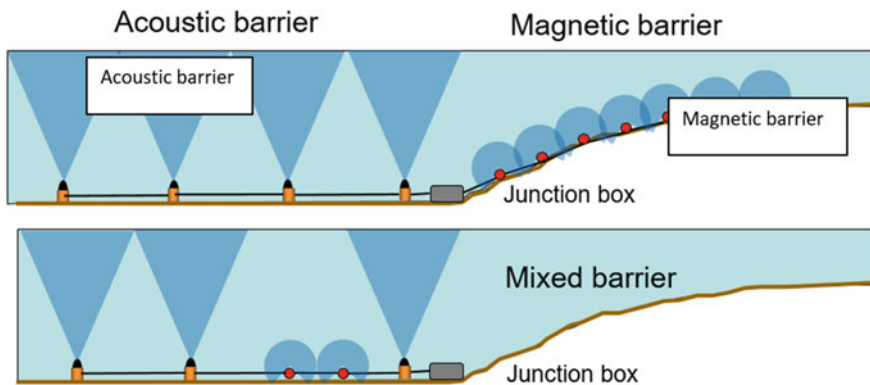


Fig. 1 Concept of underwater sensor barriers

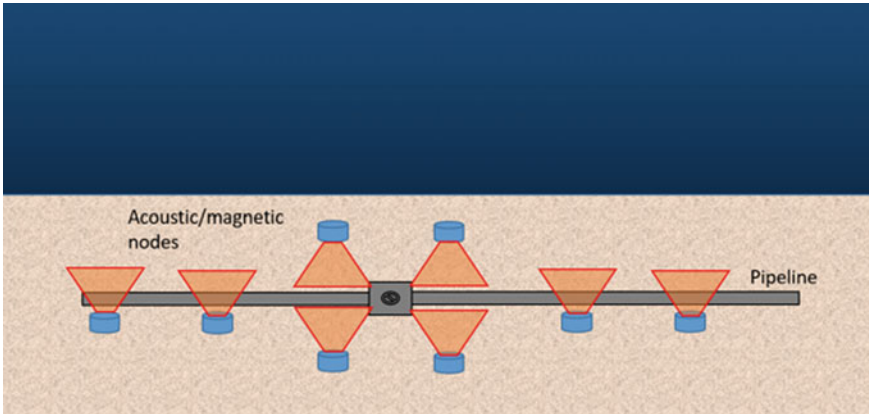


Fig. 2 Sensors strategically deployed along an underwater pipeline, with a higher concentration near gate valves and other critical points to ensure optimal monitoring and safety

scalable system regardless of the specific target. For example, long pipelines need thousands of sensors to cover their entire length, which can be costly and complicated to deploy and maintain (Fig. 2). To meet this requirement, low-cost nodes and wireless solutions are necessary. However, sensors require a power source, and a wired infrastructure can provide an unlimited power supply compared to a battery pack.

Wind farms cover a wide area, but they may already have an existing wired network for distributing energy. In this case, a wired network of sensors could benefit from these cables (Fig. 3). Alternatively, single or groups of nodes could be powered by a wind generator to reduce the number of connections and length of cables.

For perimeter monitoring, such as harbour entrances, borders of wind farms, oil and gas platforms, etc., a barrier of nodes can be deployed along lines to create a physical barrier (Fig. 4). For long infrastructures like pipelines and cables, single or set of nodes can be deployed along the length in random configurations to create uncertainty and make it difficult for threats to approach.

Different types of sensors can be used underwater. Acoustic sensors, or sonar, are the most effective. They can be passive, which means they only pick up noise and extract information from the background. Or they can be active, sending out pulses and checking for echoes to identify targets and anomalies. Passive sensors are cheaper and use less power, but their performance is limited by the noise level of the target compared to the environment and other disturbances. Active sensors are more effective and can create “acoustic” images with varying levels of precision depending on the complexity of the sensor. However, they use more energy and produce more data. Passive sensors can be used as standalone wireless sensors, while active sensors must be connected to a network to provide power and share large amounts of data. The range of acoustic sensors depends on the characteristics of seawater (temperature and salinity) and external disturbances such as wind, rain, and human-made noise.

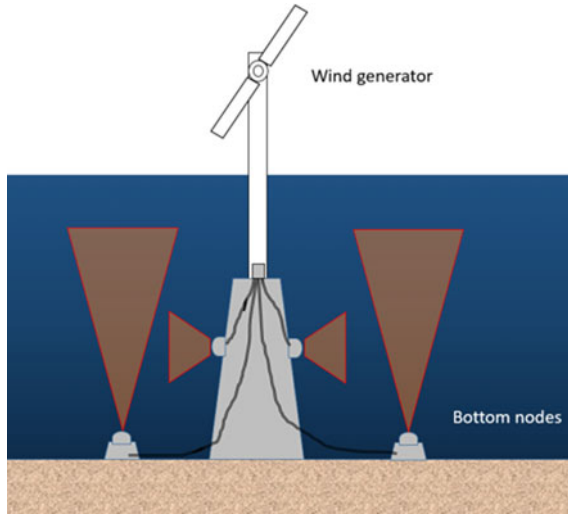


Fig. 3 Example of sensor distribution near a wind generator. Sensors are strategically placed to ensure optimal monitoring and performance

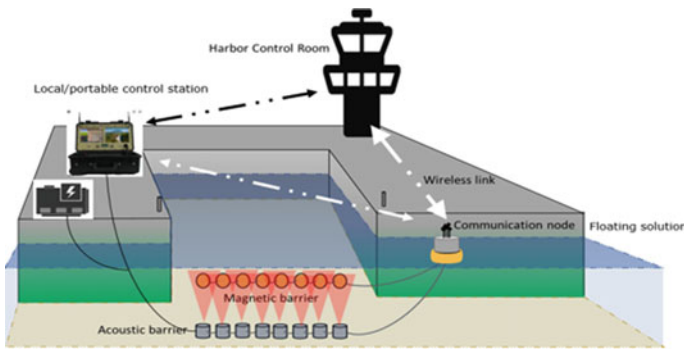


Fig. 4 Example of a perimeter and/or harbour entrance monitoring system based on acoustic and magnetic barriers to ensure security and safety

Underwater cameras are used for optical sensing. However, their effectiveness can be hindered by factors such as water turbidity, which is particularly prevalent in shallow waters near the seabed or river mouths. Limited light penetration in the water also affects their performance, with surface areas having a lot of reverberation and lamps only propagating light for a few meters in deep waters. Consequently, they are only suitable for short-range monitoring and are usually mounted on mobile nodes such as ROVs and AUVs to provide additional information in close proximity to the target.

Magnetic sensors detect local changes in the earth's magnetic field caused by ferromagnetic or metallic objects, or objects that produce electromagnetic fields. They are passive, use very little power, and produce low amounts of data. However, their range is limited due to the quick reduction of the magnetic field with distance. For instance, the signal produced by small objects like ROVs or AUVs can only be detected up to a few meters before being overwhelmed by background noise. Various types of magnetic transducers are available, but the most commonly used ones are fluxgate sensors. These sensors are affordable, rugged, and compact, with recent advancements in miniaturization leading to complete sensor solutions in the form of IC chips. To gather as much information as possible, the sensor is made up of three orthogonal transducers that measure the three components of the magnetic field in 3D space and combine the information to extract the module and data on the direction. The primary requirements for magnetic sensors include sensitivity to detect minimum variations in the earth's magnetic field due to distant movements, the ability to avoid saturation by background magnetic fields, and a low cost that allows for deploying a large number of sensors to compensate for their short range.

Environmental sensors are designed to gather a variety of data on the surrounding environment, such as chemical, physical, and biological information. They can be utilized for specific purposes, such as monitoring local pollution levels or providing support information to optimize acoustic performance of adaptable active sonar. These sensors typically have a limited range and are installed in close proximity to the target area, measuring data only at their specific location.

An optimized monitoring system should use a variety of sensor types, taking into account their different performances, limitations, and constraints, in order to achieve optimal coverage and effectiveness. Nodes can be designed to be interchangeable and use standard interfaces, allowing different payloads to be integrated together and simplifying deployment and maintenance. This approach ensures that different types of probes can be used according to potential targets, maximizing the system's capabilities.

3 Securing Ports from Seaside: Developing an Underwater Access Control System to Monitor and Prevent Unauthorized Entry

3.1 A Cost-Effective and Adaptable Underwater Barrier Based on Acoustic and Magnetic Sensors

The InSecTT project had a specific goal to support the maritime industry through the creation of advanced, cost-effective sensors for underwater use. To achieve this objective, high-performance magnetic and acoustic sensors has been developed, as well as a shared interface to enable the creation of single or multi-sensor barriers. Furthermore, innovative mechanical solution for easy deployment and maintenance

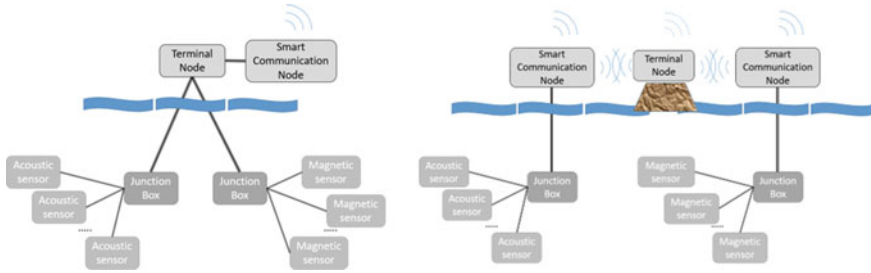


Fig. 5 Examples of connections between junction boxes and terminal nodes

of the system has been investigated, where sensors could be replaced with interchangeable payloads. In addition, the project sought to simplify data transfer from various types of nodes and payloads through extensive research and study.

The configuration of both acoustic and magnetic barriers is based on several components, including sensors nodes, connection cables, junction boxes, a local terminal node connected to a communication system, and a remote control station ashore. Figure 5 illustrates two possible configurations, with the terminal node and smart communication node integrated on a floating platform on the left, wired to the junction boxes, and the communication system transferring data to the remote control station through a wireless communication system. On the right, each junction box is wired to a communication node that provides the wireless link to the terminal node/remote control station.

Low-cost sensors can be achieved by using commercial off-the-shelf (COTS) components, which offer advantages such as mass production, maintenance and support, standardization, and access to open-source software modules. However, adapting existing transducers to new applications or functionalities requires a preliminary verification and validation phase. During the first phase of the InSecTT project, various solutions were compared, and their performance levels and cost-effectiveness analysed to select the most suitable solutions.

Magnetic sensor. A digital model of flux gate magnetic sensor was compared to an analog one (Figs. 6 and 7). While the digital model is more compact and cheaper, its performance did not meet the project's requirements. In contrast, the analog model was found to be compliant with the project's needs, but additional electronic components were required for analog-to-digital (A-to-D) conversion. As a result, the analog model was chosen for implementation in the prototypes of magnetic nodes, which are currently being tested in the maritime environment.

Acoustic sensor. Three different models of acoustic transducers, commonly used in fish finder applications, were compared during pool trials to verify their performances. However, a lengthy study phase was required to determine the appropriate conditions for their proper deployment on the sea bottom, as the installation process differs significantly from that of a fish finder (as shown in Fig. 8). Furthermore, fish finders are typically installed inside the ship hull, utilizing an acoustic window,

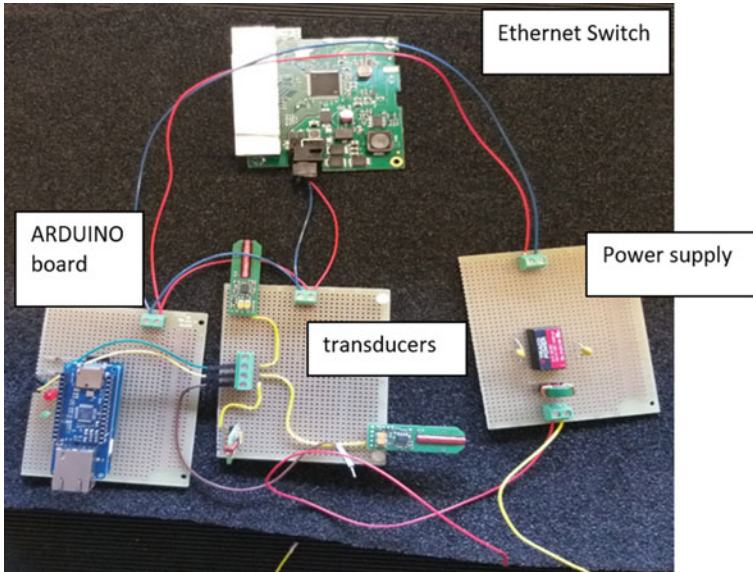


Fig. 6 Main components used in the prototyping of the analog magnetic sensor

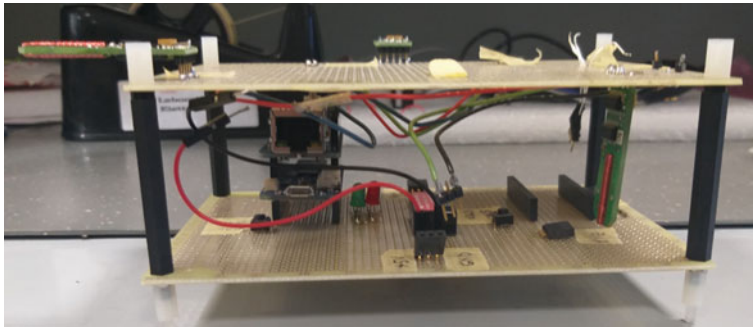


Fig. 7 Integrated prototype of the analog magnetic sensor

whereas transducers for barrier installations are placed directly in water. In the current design, two transducers are mounted on top of a waterproof case to facilitate deployment. Water volume monitoring is a similar activity, but the position of the sensors may vary depending on the specific application. Figure 9 provides some potential solutions for sensor placement, but other configurations may also be employed.

Following the selection of the transducers, the electronic components were designed and implemented, including power supply units, data acquisition boards based on Arduino and Raspberry Pi 4, and an Ethernet network for data and command sharing. The wired data distribution network was specifically designed

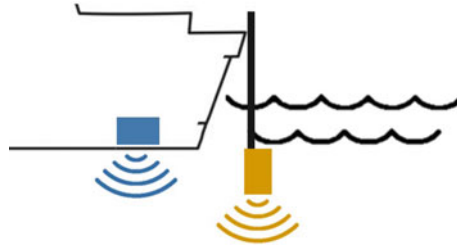


Fig. 8 Typical installation of fish finder transducers: attached to the rear part of boats or inside just on the bottom. In both cases the transducers point to the sea bottom to search for prey

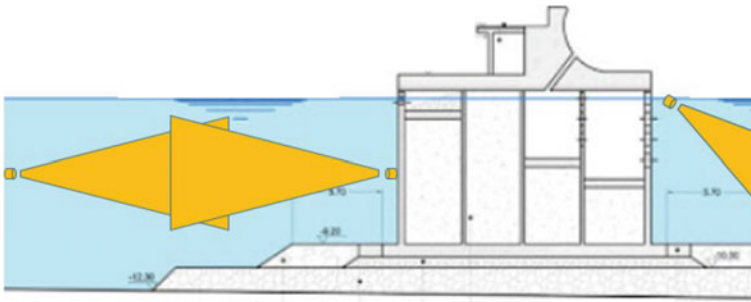


Fig. 9 Possible layout for the deployment of acoustic sensors

with a redundant link to ensure data transfer in the event of node malfunction or broken cables.

Given the deployment in very shallow waters, infrastructures could be damaged due to extreme environmental conditions such as waves and currents. Therefore, a decision has been taken to adopt a wired network based on the following constraints and requirements:

- sensor nodes require an external power supply for extended periods of use. The most straightforward approach is to utilize thin cables that can accommodate limited energy consumption and also incorporate communication functionalities. However, the most effective solution is to employ fibre optic cables and electro-optic converters integrated into the Ethernet switches. Despite the advantages of fibre optic cables, the barriers have been designed to use them exclusively for data transfer, while the prototypes are based on Ethernet cables. This decision was made to ensure easy deployment, maintenance, and the ability to adapt or modify the system as necessary;
- acoustic transducers generate a large volume of measurements, which can be filtered and fused by the processing unit installed in the nodes. However, the total number of measurements requires a significant bandwidth to ensure near real-time dispatch, as shown in Fig. 10. This requirement is not compatible with the

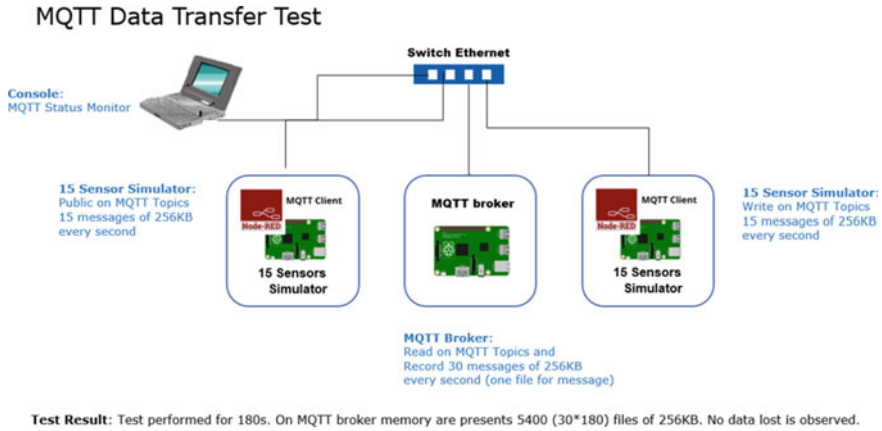


Fig. 10 Communication test conducted to verify the performance of the acoustic barrier wired network and its compatibility with the MQTT protocol

use of acoustic modems or other underwater wireless communication systems, especially given the potential delays resulting from limited range;

- magnetic transducers acquire a relatively low amount of data and operate in close proximity to one another, making their range compatible with various underwater wireless communication systems. However, deployment in extremely shallow waters is not compatible with optical and acoustic communication models, which are susceptible to environmental noise, reflections, and diffraction of light and acoustic waves.

Despite the design constraints and logistical complexity associated with a wired solution, it is still preferable due to the high bandwidth and reliable data transfer it provides. This approach requires consideration of factors such as cable length, underwater connectors, and cable weight during deployment. Maintenance can also be more complex due to the disconnection of nodes, the need for more spare parts, and longer time required to complete activities.

To manage data flows, an MQTT protocol within the wired network has been implemented. It facilitates the transmission of measurements from the nodes to the junction box and from the junction box to the remote control station ashore, as well as the bidirectional periodic checks and control messages. Additionally, it enables commands to be sent from the remote control station ashore to the barriers/nodes.

The MQTT protocol has a proven track record in managing networks of sensors, nodes, and components in various applications. Its adoption is widespread, and there are existing codes and solutions that can be adapted to meet requirements. Furthermore, the protocol is subject to continued maintenance and improvements, ensuring that state-of-the-art solutions are always available.

Since infrastructure monitoring is a time-critical activity, the system requires robust, reliable, and secure data distribution solutions. As previously stated, the wired underwater network meets these requirements. However, since the junction box must

collect all the data and transmit it to a remote control station that may be far from the barriers, wireless communication links must connect the barriers to the system. The components of the wireless network must account for the complexity of the maritime environment and the various constraints that can affect data transmission, such as:

- real-time data transfer is a fundamental requirement to ensure continuous monitoring of the infrastructure, early identification of potential threats, and prompt reaction to prevent undesired events. However, achieving real-time data transfer in the maritime environment can be a challenging task that requires extensive study and design activities;
- the use of surface buoys introduces additional challenges, as the antennas mounted on them are subject to movement and the signal could be affected by waves. This can lead to temporary breaks in communication and the need to consider strategies to mitigate their impact;
- transmission paths must consider the influence of sea surface reflections and the range, which are dependent on the transmitted power and the antenna's height above sea level. Meeting international standards for the first parameter is crucial, while the second parameter affects the size and stability of the buoy.

Existing COTS components designed for maritime environments (e.g. Glomex WBBot, Lowrance, Bantex, Scout, Ubiquity Networks, and Radwin) could provide a viable solution. However, to ensure robustness, redundant antennas could be integrated, and multiple communication protocols could be used in parallel to provide alternatives in case of issues such as antenna malfunctions, bandwidth saturation, and interference. Such a solution has been studied in this work and a detailed description of the implementation is provided in the following sections.

In the InSecTT project, there is a focus on studying, implementing, and demonstrating Artificial Intelligence/Machine Learning (AI/ML) applications in various domains and use cases, including the proposed barriers and monitoring systems. These solutions can support data analysis, data fusion, and human interpretation.

To achieve a full integrated tactical picture, the remote control console should merge data from various sensors, and AI/ML algorithms can help in different ways. For example, they can speed up the processing activity by integrating and fusing data, filter data, especially for big amounts of acoustic data, and extract low-level magnetic signals from background noise. Additionally, AI/ML algorithms can help with threat identification and classification, which is very promising given their ability to quickly compare known signals (such as the typical magnetic footprint of a target) with real-time measurements.

3.2 *A Software Defined Networking for Wireless Communication in Harbour Infrastructures*

Effective communication is critical in a harbour environment due to its complex and dynamic nature. Connecting various equipment and devices, such as ships, cranes, and sensor systems, can lead to challenges like network congestion, maintenance difficulties, and cybersecurity threats. These challenges can negatively impact port operations, reducing efficiency, reliability, and safety. As a result, it's crucial to prioritize requirements like reliability, scalability, and data transmission security [10].

In order to tackle the challenges posed by issues such as network congestion, cybersecurity threats, and troubleshooting difficulties, a Software Defined Network (SDN) architecture based on multi-interface wireless communication nodes has been proposed. This architecture aims to establish secure and reliable connections between the underwater acoustic and magnetic barriers and the Information and Communication Technologies (ICT) infrastructure of the harbour.

SDN provides a scalable and dynamically reconfigurable network architecture that separates control and forwarding functions for easy network management [11]. This architecture is divided into three main functional layers:

- *The data plane*, also known as the infrastructure layer, consists mainly of forwarding elements (FEs) interconnected via wired or wireless media. The FEs follow the instructions provided by the controller to perform packet forwarding.
- *The control plane* includes a set of software controllers that provide the control logic used to program the functions of the FEs. This layer performs general functions such as system configuration and management and the exchange of routing table information.
- *The application plane* consists of programs that provide network functions specifically for controlling data plane devices. These functions include policy implementation, network management, and security services.

The control layer of an SDN architecture provides three communication interfaces that enable it to supervise and manage network behaviour [12]:

- *The southbound interface* allows communication between controllers and communication devices. The OpenFlow protocol is the de facto standard used as the southbound Application Programming Interface (API) in SDN networks.
- *The northbound interface* provides access for applications on the application plane to the controller.
- *The east/westbound interfaces* allow communication between multiple controllers to expand control over a larger domain and increase reliability and fault tolerance.

SDN technology is a beneficial solution for the communication challenges faced in harbour environments [13]. It provides increased flexibility, maintainability, and programmability for port networks. By separating the control and forwarding planes, the network can be easily and efficiently managed, which is particularly useful in the dynamic and complex environment of a harbour. The network can adapt to changing

conditions and quickly connect new devices and equipment without the need for manual configuration, resulting in improved network efficiency.

In a harbour setting, SDN architecture improves maintainability. Centralized network control and real-time monitoring and troubleshooting enable network administrators to swiftly identify and resolve any issues, significantly reducing downtime and improving network reliability.

SDN also enhances programmability, allowing the network to be programmatically controlled through APIs. This feature enables task automation and integration with other systems, such as network management. Traditional networks typically use device or vendor-specific Command Line Interface (CLI) or Graphical User Interface (GUI) for network management, making it difficult to automate tasks and integrate network management with other systems. In contrast, SDN architecture exposes the control plane through APIs, enabling the automation of tasks such as provisioning new services, configuring network devices, and detecting and resolving network problems [14].

4 Smart Communication Node

The Smart Communication Node (SCN) is an architecture developed within the InSecTT project that leverages SDN technology (as shown in Fig. 11). The SCN consist of two primary entities: the SDN Controller and the Forwarding Devices (FDs). The SDN Controller serves as the brains of the system, managing and controlling the behaviour of the FDs, which act as data plane devices that execute decisions made by the SDN Controller.

FDs are multi-interface wireless communication nodes equipped with several Wireless Modules, a logical entity that manages a specific wireless interface. Each FD integrates one or more Wireless Modules, each for a wireless interface that needs integration. A Wireless Module comprises custom logic that can be adjusted to meet the particular needs of the network. The FD design is modular, making it extensible with other types of Wireless Modules in the future without completely redesigning the architecture. This modularity allows for scalability and flexibility of the SCN, allowing different wireless interfaces to be employed in various parts of the network, depending on their unique requirements. For example, an area may require a higher bandwidth and throughput, while another may necessitate longer range and lower power consumption. The modular design accommodates diverse wireless interfaces, making the SCN adaptable, scalable, and more flexible overall.

As depicted in Fig. 12, the FD prototype developed for the InSecTT project is equipped with two Wireless Modules: LoRa and Wi-Fi. LoRa is a low-power, long-range wireless technology that is suitable for IoT applications and can communicate over distances of several kilometers [15]. Meanwhile, Wi-Fi is a high-speed wire-less technology commonly used in homes and businesses. The inclusion of both LoRa and Wi-Fi in the FD design allows the system to take advantage of the strengths of both technologies, enhancing its adaptability and versatility for different use cases [16].

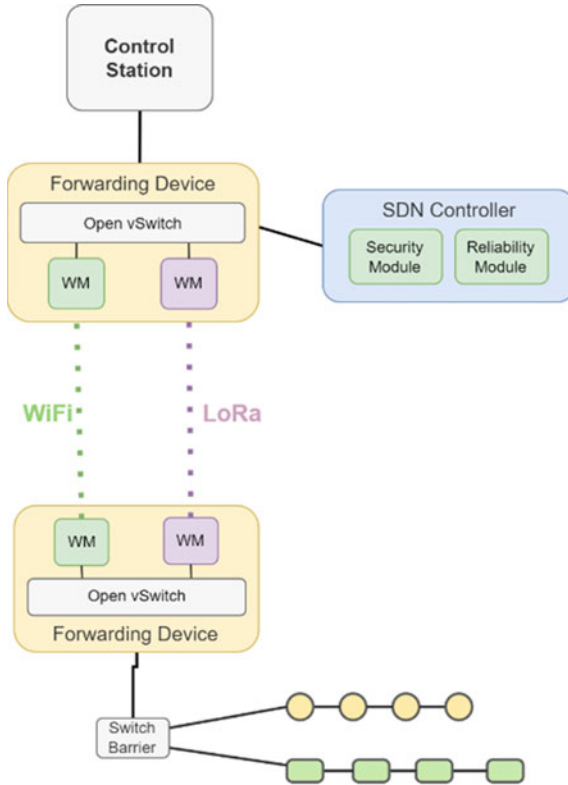


Fig. 11 Architecture of the smart communication node (SCN)



Fig. 12 Prototype of the forwarding device (FD) equipped with LoRa and Wi-Fi interfaces

The SDN architecture of the SCN employs the OpenFlow protocol as the South-bound API for communication between the SDN Controller and the FDs. To make the FD OpenFlow-enabled, the physical device is equipped with OpenVSwitch (OVS), a virtual switch that supports the OpenFlow protocol [17]. This configuration enables the SDN Controller to establish network flows between the FDs using OpenFlow rules and obtain statistics on the ports, flows, or flow tables of these devices.

The ONOS controller [18] was chosen as the SDN Controller because of its scalability, flexibility, and robustness, making it an ideal choice for managing the network of FDs in dynamic environments like harbours. The SDN Controller executes several applications, including two custom applications, the Reliability Module, and the Security Module, to supervise and control the network of FDs.

The Reliability Module is responsible for ensuring reliable wireless communication between FDs by analysing the physical channel to determine the most dependable wireless interface for communication.

The Security Module uses a Deep Learning model to conduct an IP-level analysis of each FD to detect possible attacks. If anomalies are detected, the type of attack is classified, and the most appropriate mitigation action is taken using OpenFlow's features. To ensure complete protection, this approach is combined with other security mechanisms, such as using Transport Layer Security (TLS) to communicate between the SDN Controller and FDs. TLS encrypts data and verifies the identity of participants involved in the communication, preventing data tampering and interception over network connections [19].

5 Reliability Module: An SDN-Based Approach for Reliable Wireless Communication

Harbours often experience interference from machinery, which can cause significant disruptions to wireless signals. Additionally, the harsh environment of a harbour, with saltwater and high humidity, can also have a negative impact on wireless communication. As a result, reliable and robust wireless communication between the FDs in the network is critical to ensuring the smooth operation avoiding potential delays and losses. The FDs and the Reliability Module in the SDN Controller are specifically designed to meet this requirement and provide a set of tools that will be described in this section.

The FDs and SDN Controller use a custom protocol (the SCN Protocol) in conjunction with OpenFlow for communication. OpenFlow provides a standard set of functions for managing and controlling packet flow in the network, while the SCN protocol covers all functionality not included in OpenFlow. The SCN protocol is used to send custom messages containing wireless statistics for each Wireless Module, enabling the SDN Controller to obtain a more detailed understanding of the network's performance and make decisions based on the wireless interface's performance. This

is especially important when there are multiple wireless interfaces available, as the SCN protocol allows the SDN Controller to select the most reliable interface for use.

The Wireless Agent (shown on the left in Fig. 13) is responsible for managing the main business logic of an FD. Its primary function is to collect information about the performance of the wireless interfaces on the FD and transmit this data to the SDN controller. The SDN controller can then use this information to determine how to manage the flow of data packets throughout the network, ensuring reliable wireless communication. The Wireless Agent can send statistics in two ways: either in response to an explicit request from the SDN controller, or through an event-based logic that allows the wireless agent to send statistics when certain conditions are met, such as when a certain statistic exceeds a specific threshold value.

The SDN Controller is the other critical component of the SCN architecture (Fig. 13, right) consisting of multiple SDN applications that collaborate to manage and control the network of FDs. The SCN Core module acts as an intermediary between the SDN applications and the management logic of the FDs in the network. It maintains a record of the connected devices and their Wireless Modules, controls communication via the SCN Protocol, and offers APIs for other applications on the SDN Controller. These APIs enable other modules to identify the connected FDs, send them protocol messages, detect device connections or disconnections, and more.

The Wireless Stats Collector is a sub-module of the Reliability Module within the SDN Controller that collects various statistics from the Wireless Modules of each FD in the network. These statistics comprise signal strength, error rate, signal-to-noise, and other relevant metrics for wireless communication. The collected statistics provide an extensive view of the wireless environment’s condition, which is critical

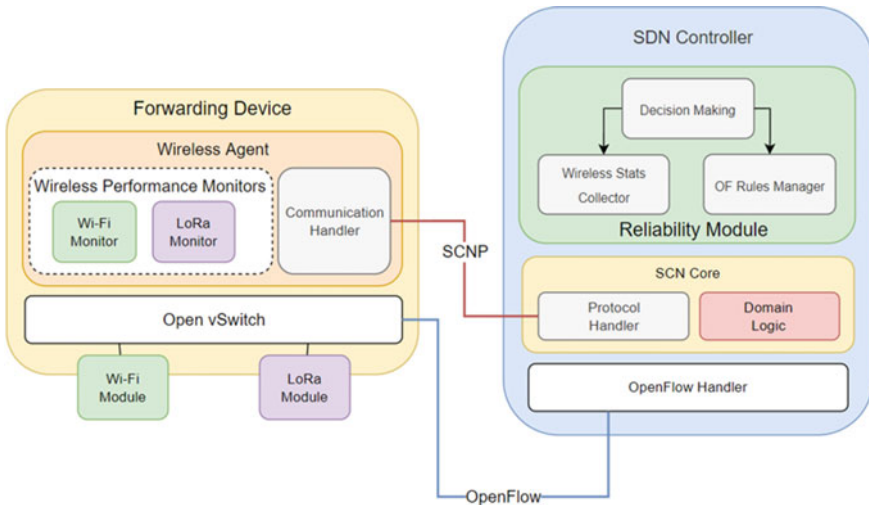


Fig. 13 Software architecture for the reliability solution, showing the architecture of the forwarding device (FD) on the left and the architecture of the SDN controller on the right

for the network to adapt to changing conditions in real time. The Decision Making sub-module uses the collected statistics to identify the most reliable Wireless Module for communication between a pair of FDs. The Decision Making sub-module is flexible, allowing for the addition of new decision-making strategies to adapt to various environments. A performance index, based on factors such as signal strength and noise, is used to calculate the quality of wireless communication between two FDs for implementing a decision-making strategy. Once the most reliable Wireless Module is identified, the OpenFlow Rule Manager sends flow rules to the FDs to direct network flow towards the chosen interface. This sub-module's traffic direction ability allows granular control over traffic flow, restricting communication between specific hosts to enhance security and network protection against potential threats.

6 Security Module: A Network Intrusion Detection System for Wireless SDN Networks

The SDN approach offers flexibility, programmability, and maintainability, but it also brings new security threats. Each layer of the SDN architecture can potentially create vulnerabilities that affect the network's overall security. Malicious applications, controller vulnerabilities, flow rule legitimacy and consistency, non-standardization of northbound interfaces, and security risks associated with southbound interface communication are the typical security concerns that arise [20].

To address these security threats, a Network Intrusion Detection System (NIDS) was designed and developed to monitor the entire network and identify intrusions and attacks by analysing traffic. The NIDS is designed to detect attacks launched by hackers attempting to gain unauthorized access to the network, steal information, or disrupt service [21]. The NIDS operates across all three layers of the SDN architecture (Fig. 14). Data plane devices collect information on the traffic they handle and periodically send it to the controller using the IPFIX (IP Flow Information Export) protocol specification, which is an IETF protocol supported by OVS. IPFIX defines how flow information is formatted and transferred to one or more collectors in a certain observation domain.

Several scientific works [22–25] have demonstrated how IPFIX features can detect and identify attacks by representing key events. IPFIX not only defines basic statistics that Exporters must send to Collectors (IANA-registered Information Elements), but also enables the introduction of new, enterprise-specific Information Elements to meet specific needs.

The Controller receives IPFIX traffic statistics from switches in the network and uses the Collector sub-module to process them. The Collector sub-module then notifies a list of subscribers, including the Detector sub-module, of the collected statistics. The Detector sub-module aggregates the flow information obtained from various switches and stores it temporarily in a cache. Periodically, it invokes the IDS Service at the application level to analyse the network flows. The IDS Service

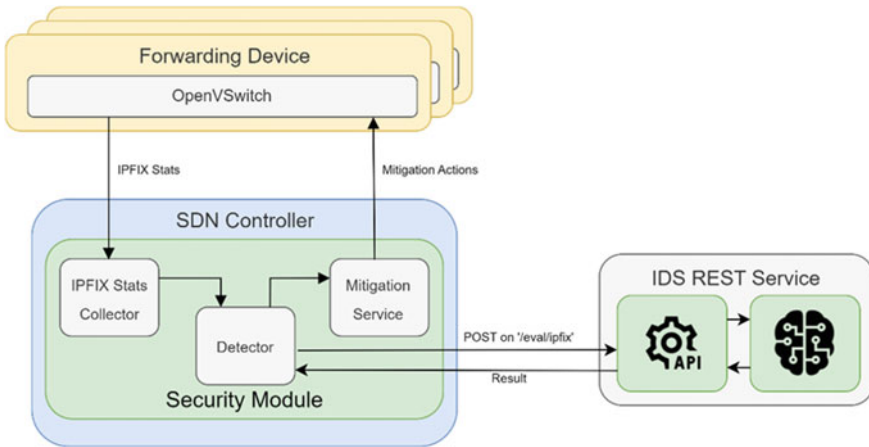


Fig. 14 Software architecture of the network intrusion detection system

employs a deep learning model to detect and identify possible attacks by analysing the data. The detection and classification tasks were addressed through the use of Deep Learning techniques because:

- Deep learning models can learn meaningful representations from complex, high-dimensional data.
- Incremental and transfer Learning paradigms can improve current performance as the data grows and adapt models to new types of attacks.

The model used in the analysis is a stacked model consisting of:

- *Anomaly Detector*: One-Class Classification (OCC) Deep Neural Network (DNN) trained by minimizing the HSC Loss [26] to detect anomalous flows. The use of an OCC approach is justified by the fact that more normal samples are usually available and the prevention of large-scale attacks is difficult due to the ever-changing nature of attacks. However, like [26], the concept of Outlier Exposure [27] has been exploited to improve understanding of what is normal. HSC Loss is a variant of cross entropy loss which forces the model to learn a latent space in which normal examples are mapped close to the origin while anomalous examples are far from the origin. This neural network considers all flows that have an outlieriness greater than a fixed threshold ϑ , determined during testing based on the ROC curve, to be anomalous.
- *Intrusion Classifier*: Multi-Class Classification DNN trained by minimizing the Categorical Cross Entropy to identify the type of attack that was performed.

The dataset used to train the Anomaly Detector and Intrusion Classifier models is sourced from [28], an extended version of [29]. To ensure that the models are not biased and are able to interpret the data correctly during training, a data cleaning phase is performed before appropriate pre-processing of the features. The data cleaning

phase eliminates attack classes and features that are not relevant to the detection of attacks in Software Defined Underwater Sensor Networks. In the pre-processing phase, numerical data is scaled using standard scaling, while categorical data is binary encoded. The IDS Service receives flows to be analysed on a REST channel in a JSON message, reorders and transforms the flow statistics into the expected format, and initiates the inference process. By building a REST API for the model, multiple applications can use it. Once the inference is complete, the predictions for each flow are returned to the Detector using the same channel. The Detector then provides each prediction to the Mitigation sub-module along with information required to select the flow for the most suitable mitigation action in the event of an anomaly. The Mitigation sub-module also maintains a record of the attack history to enable mitigation rules to be applied to new switches added to the network. It may decide to apply one of the following actions:

- dropping malicious flows;
- forwarding the malicious flows to a honeypot of the system in use;
- limiting malicious flows by applying QoS operations via OpenFlow Meter, which introduces a check on ingress packet rate and byte rate for each port of every device before applying the expected treatment.

Currently, the system employs different mitigation strategies depending on the type of attack detected. Attacks that cause system stress and limit its availability, such as Denial of Service (DoS) and Distributed DoS (DDoS), are mitigated by dropping the corresponding traffic. For attacks where it is beneficial to gather information about the attacker and their methods, the forward strategy to Honeypot is used. For rare attacks that are difficult to recognize using the current model, OpenFlow Meter tables are set up. It should be noted that IPFIX statistics for flows forwarded to the Honeypot are not collected or analysed, as the purpose of the Honeypots is to gather information on hacker behaviour.

7 Conclusions

The emergence of the Blue Economy has shifted the attention of coastal nations towards the monitoring and protection of their maritime infrastructure. As a result, the protection of essential infrastructure such as oil rigs, pipelines, and offshore renewable power plants has become increasingly critical due to the evolving international political landscape. The InSecTT project aims to address these challenges by implementing smart, secure, and affordable solutions that deploy sensors and barriers to monitor maritime infrastructure. These solutions will be integrated with wireless communication systems, including AI/ML based solutions, to provide a comprehensive framework for protecting these essential assets.

The vast areas to be monitored and the distance from the coast pose significant challenges for the communication system. Therefore, specific wireless solutions

based on redundancy and multi-protocols working in parallel must be studied and implemented, taking into account the challenges of the maritime environment.

The collaboration between Leonardo S.p.A. and University of Calabria has resulted in a valid solution for this specific application. This solution involves the deployment of acoustic and magnetic barriers at sea, integrated with a smart and redundant wireless communication system, to allow for integration with the harbour and/or infrastructure control system. However, further work is required to transform prototypes into products, and continuous monitoring of technological advances in components and wireless communication systems is necessary to keep the system up-to-date. Finally, progress in AI/ML algorithms and solutions for big-data processing and management will provide further improvements to safety and security, as well as efficient management of the barriers and the wireless communication infrastructure, in an integrated environment.

References

1. The EU Blue Economy Report 2022. Accessed 20 Apr 2023
2. Carter, L.: Submarine cables and the oceans: connecting the world (No. 31). UNEP/Earthprint (2009)
3. Amin, M.: Toward secure and resilient interdependent infrastructures. *J. Infrastruct. Syst.* **8**(3), 67–75 (2002)
4. Wendler-Bosco, V., Nicholson, C.: Port disruption impact on the maritime supply chain: a literature review. *Sustain. Resil. Infrastruct.* **5**(6), 378–394 (2020)
5. Kong, M., et al.: Real-time optical-wireless video surveillance system for high visual-fidelity underwater monitoring. *IEEE Photon. J.* **14**(2), 1–9 (2022)
6. Shimoyama, T., et al.: Maritime infrastructure security using underwater sonar systems. *Hitachi Rev.* **62**, 214–218 (2013)
7. Terracciano, D.S., Bazzarello, L., Caiti, A., Costanzi, R., Manzari, V.: Marine robots for underwater surveillance. *Curr. Robot. Rep.* **1**, 159–167 (2020)
8. Kastek, M., Dulski, R., Zyczkowski, M., Szustakowski, M., Trzaskawka, P., Ciurapinski, W., Grelowska, G., Gloza, I., Milewski, S., Listewnik, K.: Multisensor system for the protection of critical infrastructure of a seaport. In: *Proceedings of SPIE 8388, Unattended Ground, Sea, and Air Sensor Technologies and Applications XIV*, 83880M (24 May 2012)
9. Amutha, J., Sharma, S., Nagar, J.: WSN strategies based on sensors, deployment, sensing models, coverage and energy efficiency: review, approaches and open issues. *Wirel. Pers. Commun.* **111**, 1089–1115 (2020)
10. Yau, K.L.A., Peng, S., Qadir, J., Low, Y.C., Ling, M.H.: Towards smart port infrastructures: Enhancing port activities using information and communications technology. *IEEE Access* **8**, 83387–83404 (2020)
11. Singh, S., Jha, R.K.: A survey on software defined networking: architecture for next generation network. *J. Netw. Syst. Manage.* **25**, 321–374 (2017)
12. Latif, Z., Sharif, K., Li, F., Karim, M.M., Wang, Y.: *A Comprehensive Survey of Interface Protocols for Software Defined Networks* (2019)
13. Pradhan, B., Srivastava, G., Roy, D.S., Reddy, K.H.K., Lin, J.C.-W.: Traffic classification in underwater networks using SDN and data-driven hybrid metaheuristics. *ACM Trans. Sen. Netw.* **18** (2022)
14. Liu, J., Xu, Q.: Machine learning in software defined network. In: *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)* (2019)

15. Stumpo, D., Rango, F.D., Buffone, F., Tropea, M.: Performance of extended LoRaEnergySim simulator in supporting multi-gateway scenarios and interference management. In: 26th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications, DS-RT 2022, Alès, France, September 26–28 (2022)
16. De Rango, F., Lipari, A., Stumpo, D., Iera, A.: Dynamic switching in LoRaWAN under multiple gateways and heavy traffic load. In: 2021 IEEE Global Communications Conference (GLOBECOM) (2021)
17. Pfaff, B., Pettit, J., Koponen, T., Jackson, E.J., Zhou, A., Rajahalme, J., Gross, J., Wang, A., Stringer, J., Shelar, P., Amidon, K., Casado, M.: The design and implementation of open Vswitch. In: Proceedings of the 12th USENIX Conference on Networked Systems Design and Implementation, USA (2015)
18. Berde, P., Gerola, M., Hart, J., Higuchi, Y., Kobayashi, M., Koide, T., Lantz, B., O'Connor, B., Radoslavov, P., Snow, W., Parulkar, G.: ONOS: towards an open, distributed SDN OS. In: Proceedings of the Third Workshop on Hot Topics in Software Defined Networking, New York, NY, USA (2014)
19. De Rango, F., Potrino, G., Tropea, M., Fazio, P.: Energy-aware dynamic internet of things security system based on elliptic curve cryptography and message queue telemetry transport protocol for mitigating replay attacks. *Pervas. Mob. Comput.* **61** (2020)
20. Liu, Y., Zhao, B., Zhao, P., Fan, P., Liu, H.: A survey: typical security issues of software-defined networking. *China Commun.* **16**, 13–31 (2019)
21. Sultana, N., Chilamkurti, N., Peng, W., Alhadad, R.: Survey on SDN based network intrusion detection system using machine learning approaches. *Peer-to-Peer Netw. Appl.* **12**, 1–9 (2019)
22. Sarhan, M., Layeghy, S., Portmann, M.: Evaluating standard feature sets towards increased generalisability and explainability of ML-based network intrusion detection (2021)
23. Delplace, A., Hermoso, S., Anandita, K.: Cyber Attack Detection thanks to Machine Learning Algorithms (2020)
24. Erlacher, F., Dressler, F.: FIXIDS: a high-speed signature-based flow intrusion detection system. In: NOMS 2018—2018 IEEE/IFIP Network Operations and Management Symposium (2018)
25. Flanagan, K., Fallon, E., Connolly, P., Awad, A.: NetFlow Anomaly Detection Through Parallel Cluster Density Analysis in Continuous Time-Series (2017)
26. Ruff, L., Vandermeulen, R.A., Franks, B.J., Müller, K.-R., Kloft, M.: Rethinking Assumptions in Deep Anomaly Detection (2021)
27. Hendrycks, D., Mazeika, M., Dietterich, T.: Deep Anomaly Detection with Outlier Exposure (2019)
28. Sarhan, M., Layeghy, S., Portmann, M.: Towards a Standard Feature Set for Network Intrusion Detection System Datasets (2021)
29. Sarhan, M., Layeghy, S., Moustafa, N., Portmann, M.: NetFlow Datasets for Machine Learning-Based Network Intrusion Detection Systems. Lecture Notes of the Institute for Computer Sciences, pp. 117–135. Springer International Publishing, Social Informatics and Telecommunications Engineering (2021)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Security of Wireless IoT in Smart Manufacturing: Vulnerabilities and Countermeasures



Fatima Tu Zahra, Yavuz Selim Bostanci, and Mujdat Soy Turk

Abstract This chapter discusses modern smart manufacturing systems, the challenges in building such systems, and their vulnerabilities due to the lack of security features. The manufacturing industry has been revolutionized by the rapid growth of Wireless Sensor Networks (WSN) and the Internet of Things (IoT). Today, smart manufacturing systems are essential for the progress of Industry 4.0. The emerging smart Industry 4.0 benefits from the software and hardware components of the IoT ecosystem and creates a bridge between digital and physical environments which increases productivity, reduces costs, and provides better customer experience and satisfaction. IoT systems facilitate edge-computing, fog, and cloud systems and enable data-driven decisions with data analytics and artificial intelligence. However, integrating these promising technologies into the industry has led to new challenges by increasing the opportunities for adversaries to attack and sabotage industrial systems. Possible outcomes of such attacks can be extended from economic damage, loss of critical information, loss of production, serious injuries, and even loss of life. In this chapter, the security of manufacturing systems, their vulnerabilities, and potential types of cyber-attacks are elaborated on to provide insights into the liability of the existing approaches. Additionally, countermeasures to attacks and their limitations regarding existing and future security challenges are detailed to raise awareness regarding available technologies.

F. T. Zahra · Y. S. Bostanci · M. Soy Turk

Vehicular Networking and Intelligent Transportation Systems Research Lab, Marmara University, Istanbul, Turkey

e-mail: fatima.zahra@venit.org

Y. S. Bostanci

e-mail: yavuz.bostanci@venit.org

M. Soy Turk (✉)

Department of Computer Engineering,

Marmara University, Istanbul, Turkey

e-mail: mujdat.soyturk@marmara.edu.tr

© The Author(s) 2024

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_21

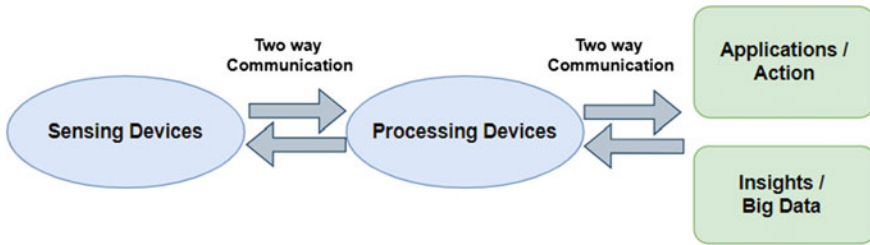


Fig. 1 Building blocks of an IoT-based smart manufacturing system

1 Smart Manufacturing Systems

Smart manufacturing (SM) is an information and technology-driven approach that employs efficient internet-connected machinery with integrated digital and physical processes in industries to continuously monitor, evaluate and assist the production process in real-time. It is an advanced application of the Industrial Internet of Things (IIoT) [1]. Specifically, it involves deploying interconnected sensors, machinery, and other instruments, which develop industrial applications to build a services network regarding energy management and production capacity in the manufacturing plant. The goal of SM is to find opportunities to automate manufacturing operations and analyze data to make data-driven decisions to improve overall manufacturing performance [2].

Smart manufacturing eases monitoring the performance at each step in the production chain. The data collected from the devices can be stored locally or managed in the cloud through APIs and services. The real-time and historical data from the machines and sensors can be streamed with the application of IIoT. Data analysts work on the data obtained from the environment to expose hidden features that can help identify potential improvements in different stages of the manufacturing process or apply predictive maintenance to prevent loss of profit [3]. Furthermore, the digital twins of industrial processes are used to identify problems and demonstrate the consequences of different scenarios and efficiently start/configure the processes in a safely manner. A simplified diagram of the building blocks of IoT-enabled SM [4] is shown in Fig. 1.

IoT systems are enabling manufacturing systems to innovate and automate their production and development process using programmable logic controllers (PLCs) and automatons that communicate with each other in real-time and enable data-driven decisions. For example, a smart factory can order new supplies of raw material automatically when the material is running out, find out failing equipment before an incident occurs with predictive maintenance, or shutdown/reboot a process/robot itself if there is a fault in the system. The main components of a smart manufacturing system [4] are shown in Fig. 2.

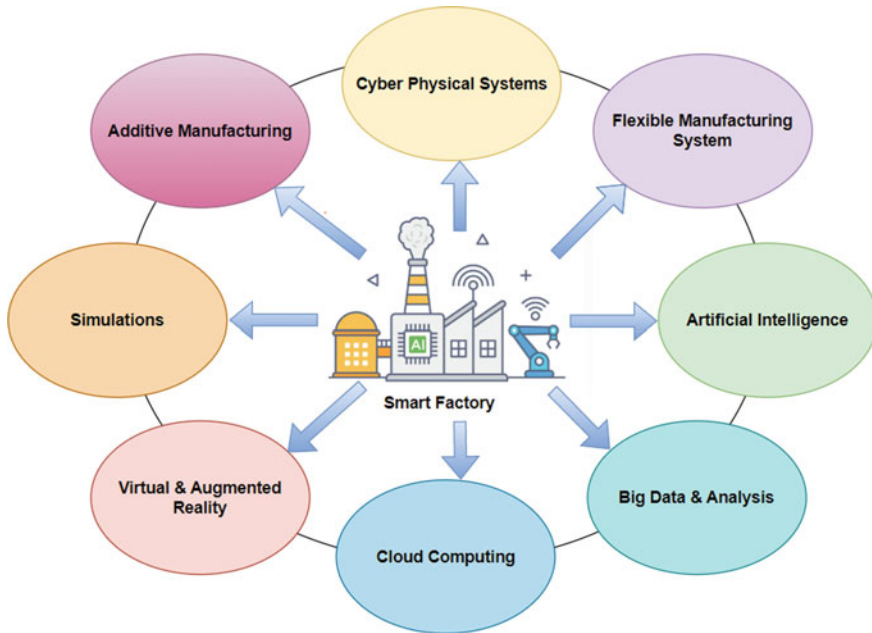


Fig. 2 Main components of smart manufacturing System

2 Current Smart Manufacturing Systems and Practices

Smart Manufacturing Systems (SMS) collect and analyze real-time data to improve decision-making accuracy and boost the productivity and performance of a plant. The research and development by academia and industry have improved network-controlled systems, robotics, and IoT systems tremendously over time [5]. This has contributed significantly to the current smart factories for efficient, sustainable, and cost-effective operation. Figure 3 shows the Computer Integrated Manufacturing (CIM) model which displays the hierarchical architecture of the communication connections of computer systems and controls of an SMS from the sensor-actuator to the enterprise level [6].

Figure 3 illustrates the five layered hierarchical model of the current manufacturing systems in industries. The top level is the Enterprise Level where the operational management decisions are made to define the production workflow of the end product (for example, launching a new model of a car). At the second level, The plant management controls the production flow locally inside the production plant(Every plant controls the production flow of the new model on its own keeping the launch date in view). The Supervisory Level in the production facility manages several manufacturing cells of end-product each of which performs a different function (Such as in the case of a car, its communication system, engine, body, and other parts). The Cell Control Level controls and manages individual process functions (production

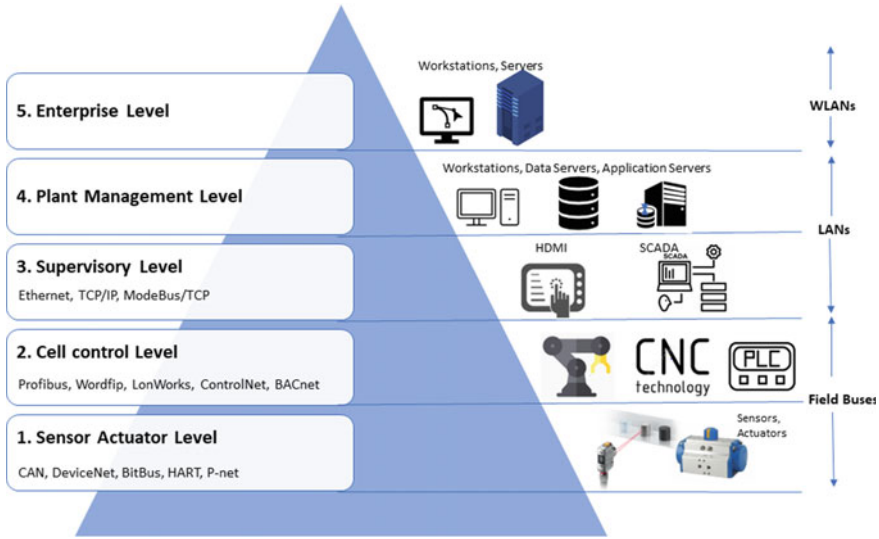


Fig. 3 Hierarchical computer integrated manufacturing model

of each individual part of the car). The Sensor & Actuator level consists of physical hardware and controllers to carry out the industrial processes. In higher layers of this model generally, TCP/IP protocol stack is used while in the lower layers, special protocols are utilized to ensure lower latencies and industry-specialized requirements. These communication protocols lack sufficient security mechanisms to strengthen authentication, integrity, confidentiality, data freshness, and methods to detect faults and anomalies [7, 8].

The master-slave structure can be extensively found in the lower layers of CIM. Here, the master usually is used to start the communication. Industry 4.0 has a more decentralized system where process components are smart and autonomous. These CIM components are smart as they are informed of their environment and can share data with each other. Here machines are autonomous objects in the production line that can perform various tasks according to the pre-coded instructions and are making real-time decisions at each step during the manufacturing process.

The disadvantage of such an open and self-aware and self-governed system is that it is more susceptible to both active and passive security attacks. The consequences of the attacks can be catastrophic compared to conventional systems with such features as the reconfiguration flexibility of the devices, open architecture, and usage of data analytics that can lead to complex dynamic behaviors. These can result in faults to the production line, the product, the physical environment, and as well as to the people.

3 Challenges of Smart Manufacturing

A huge part of economic growth is dependent on manufacturing industries. In the current era of the modern fourth industrial revolution, investments in smart manufacturing have been rapidly growing as they have assisted factories to achieve 17–20% productivity gains whilst simultaneously reaching an increase of 15–20% in quality gains [9, 10]. As a result, all big manufacturing companies in the world are digitizing their factories and manufacturing processes to maintain their credibility and growth in an extremely competitive global market. The automation achieved in manufacturing industries has resulted in better quality products, flexibility, and enhanced productivity. One major problem of this whole scenario is that security is treated as a minor concern rather than a vital component of the development and deployment process. This significantly increases vulnerability and cyber security attacks on existing manufacturing systems. The vulnerabilities and loopholes in the current systems are not investigated thoroughly resulting in unprepared and defenseless organizations in the face of security threats [11].

IoT in SM has led to a major improvement in product quality and efficiency of the manufacturing process, but still, there are enormous uncertainty issues that can arise during its implementation [12]. Some of the main challenges include the uncertainty of machine designers and builders and even the end users to contrive this technology, as each enterprise has its own design requirements and process. This also creates the need for a customized design which is usually expensive and requires experts in the related domain. The security-related issues are also a huge challenge, especially where information from the industry must be collected and shared as big data with the co-partners to make it an IoT-cultured organization or for the stakeholders and monitor every part of the process advancements. Another pressing issue is affiliation and collaboration with other establishments and bringing all the supporting organizations into one coherent package [13]. The development of a connected manufacturing policy to the standards for transparent and sustainable IoT communication systems is another obstacle in SM.

4 Smart Manufacturing Vulnerabilities and Attacks

Smart manufacturing systems are also mentioned as the Industrial Internet of Things (IIoT). IoT systems benefit from both wired and wireless connectivity to collect, monitor, and process the data coming from the working environment and communicate with each other to operate tasks. Integrating IoT devices into the manufacturing site enables us to control the production process in real-time without the need of being on the site, and it provides the ability to control the cyber-physical systems remotely.

The idea of smart manufacturing suggests that the whole manufacturing process can be automated with the use of connected things and artificial intelligence to

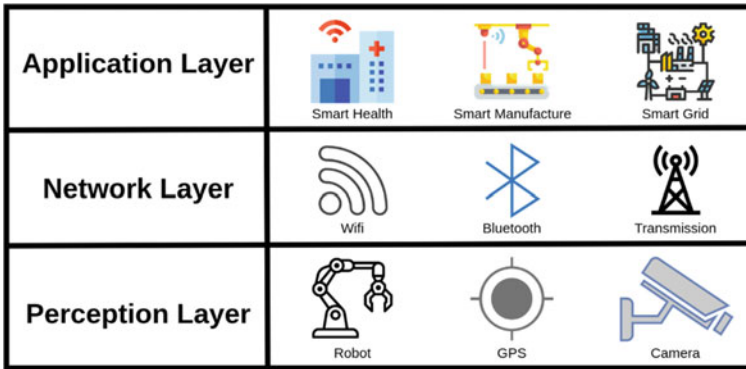


Fig. 4 General structure of an IIoT system

monitor and take action for critical decisions. It increases the efficiency of production and generates data to be evaluated whether by algorithms or data experts. However, the vulnerability of these systems may provide an intruder to perform malicious operations when they are not addressed and eliminated gracefully (e.g. malware injected into the code of the robot arm system to change its course of action). Figure 4 shows the layered structure of IIoT systems [14, 15].

According to the TCS Global Trend Study report in July 2015, security and reliability are the most important factors which make it difficult to integrate IoT systems into workspaces [16]. At this point, considering the possible attacks that can be made because of having vulnerabilities in any of these layers, industries need to consider the criticality of the applications and take precautions to prevent harmful outcomes. In this section, the most popular and well-known cyber-attacks are described.

4.1 Perception/Physical Layer Attacks

The perception layer can be defined as the lowest layer in the conventional IoT structure and it contains the physical devices [17]. IoT devices are connected to each other and to the network in this layer. The data gathered from the environment is processed with the deployed smart devices and the final processed data is transmitted to others [18]. In IoT systems, the data wireless networks are the preferred transmission method in many applications. The nature of wireless networks allows a person to easily monitor, intervene, and take actions to disrupt the system although there are security mechanisms that try to prevent that [19]. Some of the most common perception layer attacks are described below:

4.1.1 Denial of Service (DoS) Attacks

The Denial of Service (DoS) attacks are one of the major security attacks among others in terms of application in different layers. The target of DoS attacks in the perception layer are the IoT components such as the sensors, actuators, IoT devices, etc. The attacker uses the vulnerabilities of the IoT nodes and blocks communication. One of the consequences of the DoS attack is that the attack puts the node under too many computation tasks and may cause the node to be forced to stay awake due to high network traffic than preferred and result in “barrage” and “sleep deprivation” [20, 21]. The barrage attack differs from the sleep deprivation attack in terms of the rate of the sent requests, with the former being characterized by a high package rate that quickly depletes the resources of the victim. Given this nature, the barrage attack can be readily detected [22].

4.1.2 False Data Injection Attacks

The attacker may inject false information into the network by capturing data transmitted by an IoT device and replacing it with a false message. Then the false information is transmitted to other entities in the network [6, 20].

4.1.3 Replay Attacks

Some attacks aim to gain the trust of IoT system components first. The attacker captures the data package that was sent previously by an IoT node by “eavesdropping” and retransmits it to a target node to misdirect its work cycle. The attacker can interrupt the ongoing process in the target node by transmitting valid data, hence causing the target to believe the data is coming from another legitimate node [14]. Since the information gathered from the network is resent to a target node without needing modification or decryption, the attacker does not need to have complex skills which is another danger of the replay attack.

4.1.4 Eavesdropping and Interference

Eavesdropping is a type of passive attack where the intruder senses the network to gain critical information and it can reveal the communication patterns of private entities, by mapping out the general behavior of the network. Wireless communications by nature are vulnerable to this type of attack since the transmission medium is exposed to third-party listeners [6, 20].

4.1.5 Side-Channel Attacks

Side-channel attackers target the device's hardware or software by either intercepting the continuous transmission of electromagnetic waves or wireless communication among nodes. In doing so the attacker can access private information such as power consumption, electromagnetic and thermal emissions from hardware components, the topology of the wireless network, and such. These attacks can also be used to analyze the timing of certain operations. In some cases, this is used as an assistance factor to a cryptanalysis attack where the side-channel attack analyzes the time it takes for the encryption process to perform. Using the gathered inference, the attacker can predict the encryption key and increase the effectiveness of a cryptanalysis attack [23].

4.2 Network Layer Attacks

The network layer in the IIoT architecture creates connectivity between things and enables communication among components. It serves as a bridge between the lower (Perception) and the upper (Application) layers. Individual components such as the nodes of the IoT are aware of other changes in the environment over the network layer since all components in the system provide their data in this layer. Therefore, it serves as the center of the network architecture to provide key information to the intended clients. Any cyber-attack on this layer might cause every process to be halted in the system. In this part, the most important and effective attacks against the network layer are explained [24].

4.2.1 Spoofing Attacks

The attacker impersonates an existing device in the IoT system to disguise it as a legitimate network user. The attacker can pose as a legitimate user and gains access to critical information from the network and/or provide false data through the system using the identity of a trusted system entity.

4.2.2 Man-in-the-Middle Attack

The attacker places itself in between the communication of two entities and relays communication between them. This allows the attacker to access and control private traffic and possibly alter the exchanged information, thereby compromising the privacy and integrity of the data transferred.

4.2.3 DoS Attack at Network Layer

Denial-of-Service (DoS) attacks are cyber-attacks that attempt to make a website or network resource unavailable to its intended users. DoS attacks are carried out by flooding the targeted system with traffic or sending specially crafted packets that can cause the system to crash, become unresponsive, or slow down [25]. Here are some common types of DoS attacks:

1. **SYN Flood Attack:** This type of DoS attack exploits a vulnerability in the TCP protocol, causing a server to allocate resources for incoming connection requests without ever completing the three-way handshake process. This leads to a denial of service by overwhelming the server's ability to respond to legitimate requests.
2. **Ping of Death:** This type of DoS attack involves sending an IP packet that exceeds the maximum size allowed by the protocol. This can cause the system to crash or become unstable.
3. **UDP Flood Attack:** This type of DoS attack involves sending a large number of UDP packets to a targeted system, overwhelming its ability to handle the traffic.

DoS attack causes a loss of network connection and in the end makes the services in the system unavailable to other legitimate nodes/users by pushing the limits of the target network with a single computer. The core idea here is to use up the available network resources in a short amount of time by sending an inappropriate number of requests, thus preventing the intended users from getting a response from the network. Attackers can increase the number of requests in a given time or increase the processing time per packet in order to condense network traffic. The type of attack where multiple computers are used to attack a single target is called "Distributed Denial of Service (DDoS)".

4.2.4 Wormhole Attack

The wormhole attack in IoT uses two new nodes which are superior to the other nodes in the target system so that the information transmission path can be routed. The key point in this attack is to place these wormhole nodes in a perfect position so that their distance from the other nodes is minimized. During the attack, the first attacker node receives data at one location of the network and transmits this to the other attacker node, which in the end carries this information to the final destination as shown in Fig. 5. While the malicious nodes transmit the data in the network, they inform the other nodes about their route so that other nodes think this is the shortest way of sending a message. Because of this, the wormhole attack can be used for compromising the routing algorithms in systems. The attacker can control the data flow traffic and stop the transmission at its will [26].

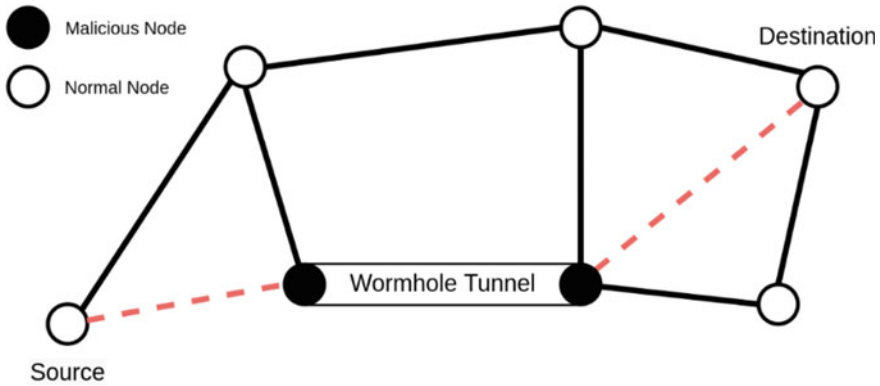


Fig. 5 Wormhole attack

4.2.5 Sinkhole Attack

In a sinkhole attack, attackers either hack a node or create a fake fabricated node in the target network. The malicious node convinces other nodes in the network that it is the closest node to the base station. The objective here is to direct all the network traffic to the malicious node which creates a sinkhole around itself. The sinkhole attack not only enables monitoring of the network but also creates a setup for eavesdropping and selective forwarding attacks by leaving new messages in the network. Similar to the wormhole attack, the sinkhole attack also compromises the routing algorithms [27, 28].

4.3 Application Layer Attacks

The Application layer is the uppermost layer of the IoT infrastructure. It is responsible for the interaction with the end user and enables access to data in the network. The vulnerabilities in this layer are in the software and the attackers have the purpose of capturing the credentials of the users.

4.3.1 Phishing Attack

The goal is to mislead the users to enter sensitive information into mock-up interfaces or lead them to install applications that contain malware that infects the system upon installation. These types of attacks rely on the lack of active security measures in the system or the user's lack of knowledge or attention. Web interface forgery, link manipulation, filter evasion, zero-day malware, etc. are the most common methods.

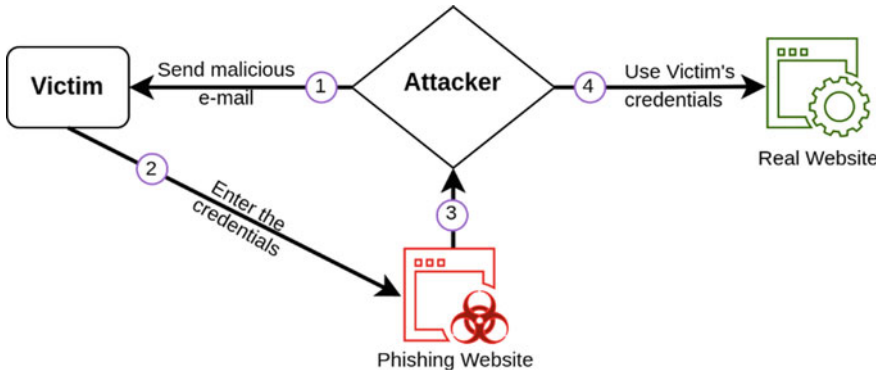


Fig. 6 Phishing attack

A typical phishing attack scheme is shown in Fig. 6 [20]. At first, the victim sends some malicious mail containing a phishing website that looks like the original one. The phishing website is made to look like a legit website such that the user does not suspect its legitimacy. Users then enter their credentials into the fake website and let the attacker capture sensitive information and credentials, which can be used by the attacker to access the original website later on.

4.3.2 HTTP Floods

An HTTP Flood is a DDoS attack where the attacker sends large quantities of HTTP GET or POST requests to a target server in order to consume its available resources and thereby prevent it from serving legitimate users. HTTP flood attacks require less bandwidth than other attacks and they are often carried out via several interconnected devices that are infected by malware. POST attacks typically use more resources from the target server compared to GET attacks. However, GET attacks are simpler to create and they can scale more efficiently [29].

4.3.3 SQL Injections

SQL injection attack is a type of code-injection attack where malicious SQL commands are injected into the user input and submitted to the target system by leveraging the target system's vulnerabilities that are mainly caused by "insufficient validation" of user input. SQL injection attacks can have a range of side effects, depending on the nature and severity of the attack. Here are some of the common side effects:

1. **Unauthorized data access:** SQL injection attacks can allow attackers to gain unauthorized access to sensitive data, such as user credentials, financial information, and personal information.

2. **Data manipulation:** Attackers can modify, delete or insert data into the database, which can result in data loss, corruption, or inaccurate records.
3. **Server compromise:** If the attacker gains complete access to the database server, they may be able to execute commands on the server or even gain access to other systems on the network.
4. **Reputation damage:** SQL injection attacks can damage the reputation of an organization, especially if sensitive data is compromised or customer data is lost.
5. **Legal and regulatory repercussions:** Depending on the nature of the data lost or compromised, organizations may face legal and regulatory repercussions for failing to protect sensitive information.
6. **Financial loss:** Organizations may face financial loss due to lost revenue, legal fees, and costs associated with repairing the damage caused by the attack.

SQL is often carried out using a combination of different injection types depending on the goal of the attacker. Injection mechanisms can be through user input, cookies, and server variables. The objective of the injection can also be indirect where the attacker seeds malicious input into the system for later use. This allows the attacker to separate the injection from the attack, meaning that the injection and attack time and place can be different thus making it more difficult to detect [30].

4.3.4 Slowloris Attack

Slowloris is a DoS attack that attempts to overwhelm web servers by sending slow and incomplete HTTP requests. The attacker sends a piece of an HTTP request to be responded by the server, but the final line of the request is not sent which causes the server to remain in waiting mode. When there is no response from the user side to send the final part, the server will timeout. In order to avoid this, the attacker sends these requests slowly enough to keep the server alive. It is difficult to detect the Slowloris attack since the sent HTTP request is legit [31, 32].

5 Security Solutions for Smart Manufacturing

Ensuring the security of smart manufacturing (SM) is a continuous process, not only a feature on a single device. Right from the very beginning of conceptualizing an idea, all the way to the final product, security must be considered one of the top priorities in SM systems. Some of the solutions specific to all IIoT layers are provided in this section.

5.1 Perception/Physical Layer Security

Defending a smart manufacturing system from cyber-attacks is challenging since such systems tend to be consisting of complex components and infrastructure. The classic “keeping attackers out” principle is still a good approach but it’s not sufficient to prevent such attacks. Perception layer attacks often include accessing the hardware devices physically, therefore several measures are taken. *Physical hardening* is one measure that helps achieve security by making the hardware tamper resistant. It includes restricting physical access to only a few authorized people, especially for unsupervised devices. Perception security can also include physical port locks; or the camera, USB and Ethernet covers, etc. [33]. Meanwhile, vulnerabilities may be discovered after devices are deployed. In such cases, all the devices must be designed in a way that they must be able to receive updates and patches post-deployment. *Upgradability* is another measure against such attacks. A common approach is to use a proper digital signature to prevent unauthorized modification of the firmware upgrades. Eventually, old IoT devices will become obsolete, with the fast-evolving technology and constantly changing requirements. These devices with expired lifespans must be destroyed properly so that no private data could be accessible by the attackers.

5.2 Network Layer Security

Multipath routing is an efficient way of defending the network system against cyberattacks. Unlike single-path routing techniques, multipath routing techniques eliminate the overhead that comes from constructing a new path when the path of the network fails [34]. This technique constructs a network with multiple paths between the source and destination. Therefore, in the case of any collapse in one of the paths, another path within the network can take over the transmission. Spoofing, wormhole, and sinkhole attacks can be tackled with the multipath routing technique. Some techniques for network layer security are shown in Fig. 7.

Another common method for network layer security is authentication. There are several ways of authentication, such as physical and virtual multipath authentication also hop-by-hop and end-to-end authentication [35]. End-to-end authentication is the most efficient way of building a secure network, but it comes at a high cost. In [36], authors designed a network model that consists of a couple of base stations and numerous nodes with limited resources. The study claims that the nodes are trustable enough to not allow internal attacks. The message authentication is used in order to prevent external attacks and a hierarchical 3-way handshake routing tree from the nodes to avoid wormhole attacks. Directional antennas and package leash mechanisms are also used for the defense of wormhole attacks. The authors in [37] adopted an antenna model and came up with three protocols in order to detect and prevent attacks. However, this defense mechanism is not as efficient when the

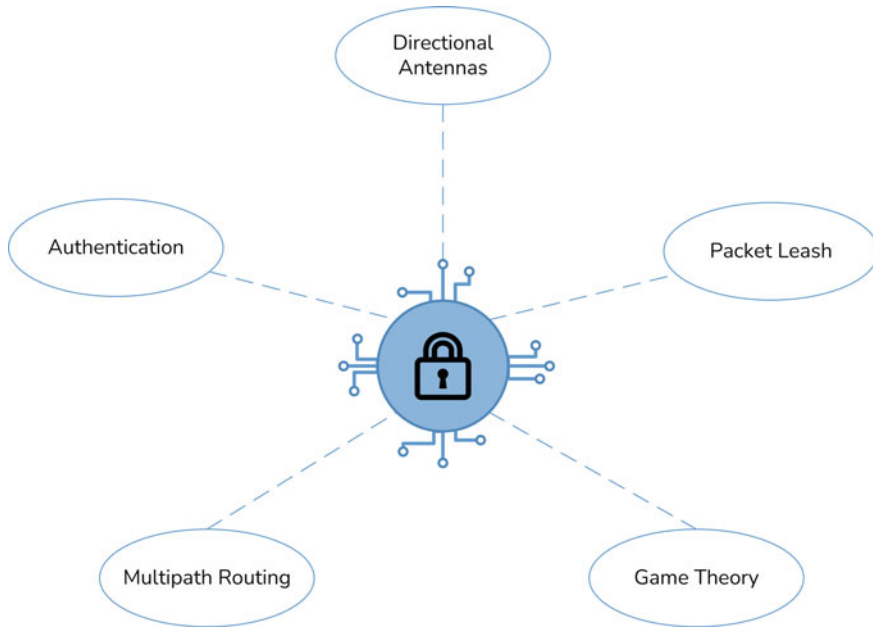


Fig. 7 Various techniques for network layer security

attacker has multiple endpoints. Another disadvantage of this method is that it causes performance degradation as a consequence of preventing legitimate nodes from being established.

In [38], the authors proposed an approach called packet leashes. Here, every packet has its own leash and it keeps any kind of information in order to prevent violation of the allowed transmission distance. This mechanism consists of geographical leashes and temporal leashes. In geographical leashes, the distance between the sender and receiver is limited. The packet leash includes the locations of the nodes and the times the packet was sent and received. The speed of light is accepted as the maximum speed of the packet. Based on this information, an attack is detected if it exceeds the upper bound of the distance as shown in Fig. 8 [38]. Temporal leashes require synchronization between the clocks of all nodes in the network. It sets an expiration time for the packet. The receiver checks the time when it receives the packet and compares it with the time it was sent. An attack is detected if this time exceeds the upper bound transmission time. Another way of detecting the attack is to set the sent time as an offset in the packet and not accept it on the receiver side if it exceeds the limit. This method is used against wormhole attacks.

Game theory is also an important research area to develop methods for preventing DDoS attacks. In [39], the authors use the UDSR protocol to maintain network security. It detects the nodes' misbehavior, identifies them as malicious nodes, and does not receive their messages. Therefore, any attacks that might be able to come from these malicious nodes are avoided and network security is established.

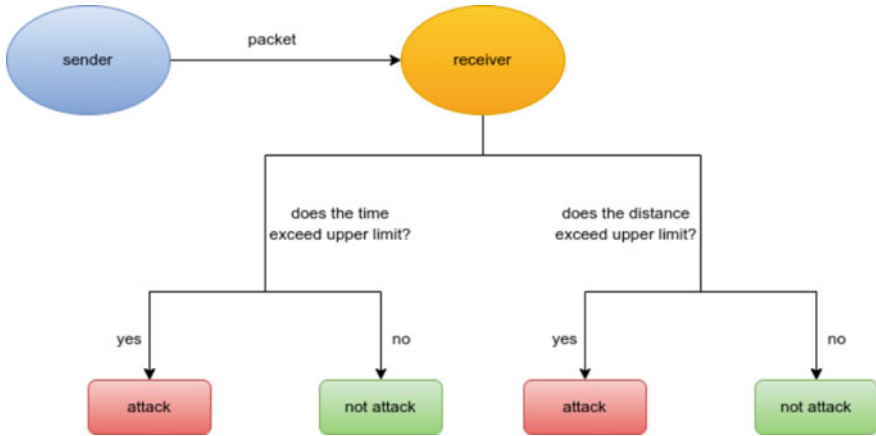


Fig. 8 Packet leash defense mechanism

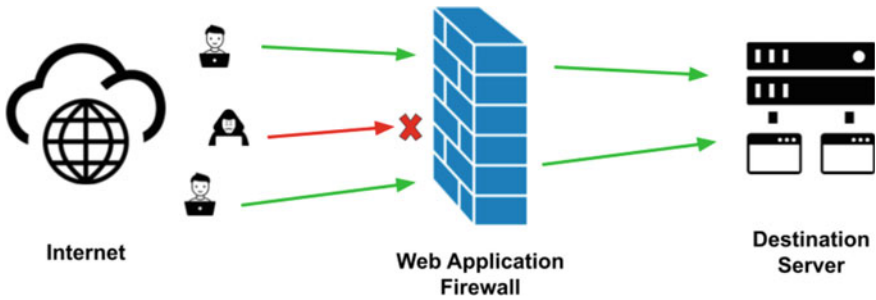


Fig. 9 Web application firewall

5.3 Application Layer Security

As the application layer is the closest layer in the whole system to the end user, it enables many waypoints for attackers with different purposes to target. Unauthorized access, data theft, data loss, and stability issues are potential consequences of weak application layer security. Web Application Firewalls (WAF) shown in Fig. 9 are often used by organizations in order to secure the system against most malicious attacks. WAF protects web applications by monitoring and filtering the network traffic between the application and the internet and stops bad traffic and malicious requests.

Human factor is also very important when considering application layer security. System users must have strong and secure passwords, and devices should not be left unlocked. Used apps can also have some vulnerabilities which can be hacked by using Cross-Site Scripting(XSS) and SQL Injection(SQLi). Hence, these web applications should be tested against vulnerabilities.

Not only the applications developed by organizations make the system vulnerable. Third-party software and firmware used on the system can also create a vulnerable waypoint therefore receiving the updates and security patches for such software in a timely manner is important. Moreover, unlicensed/pirated software should never be used. Inappropriate role access, lack of multifactor authentication, insecure password recovery mechanism, and insufficient authentication, and authorization are some other application layer security issues.

In [40], a defense mechanism against DDoS attacks namely Defense System Against Tilt (DAT) is proposed. In this method, the behavior of the sender node is observed and analyzed during the connection. It detects the malicious node when the sender misbehaves. It is designed such that the packet transmission by the legitimate user is not interrupted even if the system is hacked.

DAAD (DNS Amplification Attacks Detector) is a tool developed by the authors in [41] against DDoS attacks. It is designed to build a secure system for DNS local servers, but it also works for local network hosts. DNS packets are monitored using tools such as IPtraf which are helpful in monitoring network traffic and providing network statistics as well. In each DNS message transmission, DAAD checks the message and decides whether it is a response or request. For the responses, DAAD searches for the corresponding request in the database and if there is no corresponding request, the transmission is identified as suspicious. When the suspicious transmissions reach the threshold value, an alert is generated and message transmission from the suspicious node is blocked.

6 Approaches from Recent Studies for Cyber-Security in Manufacturing

In the InSecTT project, partners are collaborating on various industrial areas to develop AI-enhanced IoT solutions to address different cybersecurity concerns in a well-organized structure. One use case provided by Arçelik within the project directly focuses on wireless reliability and cybersecurity in manufacturing. Wireless IoT systems are exposed to several malicious attacks not only because they use shared transmission mediums but also because most IoT devices have minimal security features [42]. To cope with wireless communication issues, multiple QoS parameters should be considered. Within the project, various applications are developed to perform deep packet inspection and provide the hardware parameters, link quality, network performance, protocol-specific features, and delay measurements along with active and passive network monitoring data. The infrastructure is realized in the lab environment to investigate the OPC-UA protocol and communication aspects. For this purpose, OPC UA server and client applications are developed and deployed on IoT devices in the lab.

Monitoring and analyzing tools and services developed for the use case are allowing real-time and historical data to be inspected. They provide interfaces for people

and for the machines to interact with the data sources under reliable network protocols. Additionally, alerts and warnings based on QoS and performance metrics provide visualization and statistics. When the “data collection”, and “data visualization” aspects are achieved, more insights about the network data can be explored. The aim is to benefit from machine learning algorithms, statistical methods, and time-series analysis approaches to detect anomalies, and identify possible security attacks, more specifically “jamming”. Considering connectivity problems which is a by-nature effect of these types of attacks, the lack of data should also be taken into account while designing an appropriate model and algorithm. This way the data can be further analyzed to detect anomalies in terms of quality, reliability, and availability of the services. Within the use case, AI-enhanced approaches are explored to detect connectivity problems, performance degradation, and jamming whether it is an intentional attack or unintentional interference. Radio jamming and de-authentication attacks are inspected to address two common types of attacks.

6.1 Radio Jamming Attack

Radio Jamming is a subset of Denial of service (DoS) attacks where an attacker can obstruct legitimate packets by transmitting interference signals in the wireless channel in which the wireless devices are operating. Jammers interrupt wireless communication by producing high-power noise near the transmitting and receiving nodes across the entire bandwidth. Jamming attacks can interrupt communication, cause connectivity problems, avoid the availability of services, and eventually, degrade the performance of IoT devices significantly both regarding energy consumption, as well as network throughput. An intentional jamming attack is when someone would deliberately tries to obstruct the wireless operation. On the other hand, when IoT devices are exposed to undesirable wireless transmissions by nearby devices (mobile phones, satellites, other IoT devices), they may unintentionally be obstructed by these devices. There are different types of jamming attacks applied in the wireless medium namely, constant jammers, deceptive jammers, reactive jammers, intelligent jammers, and random jammers [43]. All these types of attacks have varying detection probabilities and can fully or partially block communication [44]. Because of that, it is of utmost priority to design effective mechanisms to detect jamming attacks and to apply countermeasures.

6.1.1 Jamming Attack Scenario Setup

To detect jamming attacks and network anomalies in the manufacturing industry with robots working in the production line an effective jamming/anomaly detection system is developed. OPC-UA is an industrial standard protocol used in such systems. Keeping this in mind, an infrastructure is realized in the lab environment to investigate the OPC-UA protocol and its communication aspects. For this purpose, OPC-UA

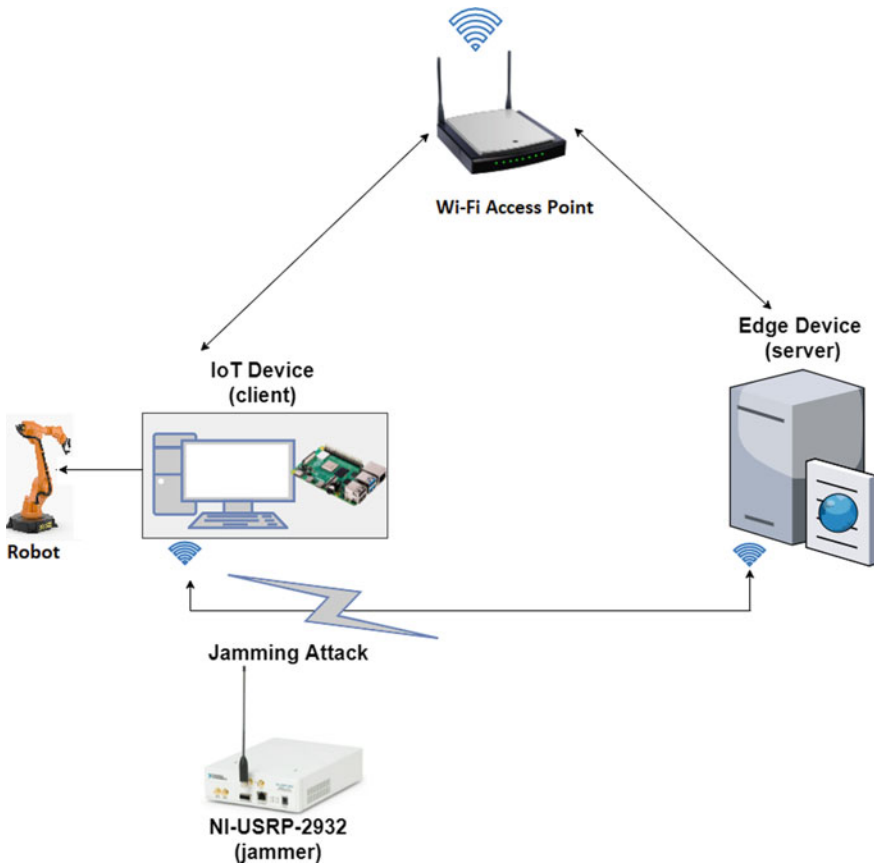


Fig. 10 Experimental setup for jamming detection

server and client applications are developed and deployed on IoT devices in the lab. A data collection application that collects certain QoS parameters from an industrial robot and an edge device is designed. The robot parameters on the client side are controlled with this application. An SDR is configured in a way that it can be used to send high-power jamming signals across 2.4 GHz WiFi channels. These signals cause severe noise in the communication channel and make it difficult for the client to communicate with the server. The experimental setup is displayed in Fig. 10.

6.1.2 Insights and Results from the Experiments

The effects of jamming were studied on the communication network on the application layer. A significant increase in application delay (approximately ten times the normal delays) was observed during the jamming attack. Also, the network through-

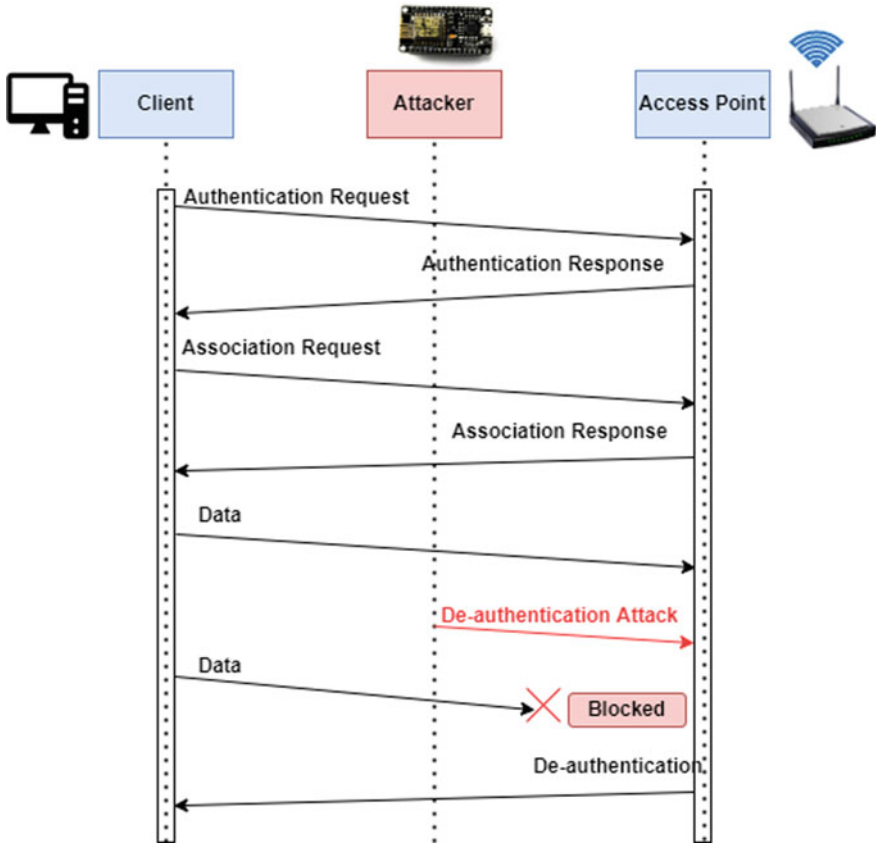


Fig. 11 Representation of De-authentication attack

put gets adversely affected by the attacks. Statistical models were developed to detect jamming using these parameters regarding network delays and throughput. Later, by utilizing deep learning, a state-of-the-art stacked LSTM model is trained using the data generated to identify jamming attacks and anomalies from the normal network data. The trained deep learning model for jamming detection has a False-Positive rate of 0.28% and a False-Negative rate of 0.21% according to the dataset generated from experiments on the test setup. It provides an accuracy of 99.5% with 99.44% precision. The use of machine learning and AI models to construct a comprehensive analysis and model design for real-time jamming detection mechanisms are ongoing studies.

6.2 Deauthentication Attack

Another type of Wi-Fi jamming attack utilizes a Wi-Fi Deauther to generate de-authentication and dissociation packets to block communication. The 802.11 WiFi protocol includes a de-authentication feature to detach users from the network. This type of attack exploits the vulnerability of this feature as it does not require any encryption for this frame even if the session is established. These packets can be leveraged as anyone can transmit them while pretending that these packets are coming from the WiFi router. Upon receiving these packets, the connected device immediately disconnects from the network. By repeatedly sending these packets connected devices encounter denial of service which is considered jamming. The de-authentication attack is illustrated in Fig. 11.

The Wi-Fi de-authentication application developed by SpacehuhnTech [45] was used to create the jamming scenario. Due to internal timers and reliability mechanisms implemented in the transport layers of the network, an increased rate of retransmission of packets as well as lower packet density as compared to the normal operation is observed on both sides of the connection. Using the network interface card in the monitoring mode, it was observed that the density of de-authentication and dissociation packets significantly increases. Based on these observations, a statistical method to detect de-authentication attacks on-device to increase local awareness of the IoT device. In this method, the packet drop rate (PDR) and connectivity continuously are monitored along with the density of de-authentication packets in a certain period of time. If the PDR and de-authentication packets increase by a certain threshold, a de-authentication attack alert is generated. The de-authentication attack causes connectivity loss as well so connectivity is constantly monitored and reported as well.

7 Conclusion

Smart Manufacturing is the future of the industry due to massive opportunities and benefits it offers. At the same time, the security of systems and the subversive outcomes of security attacks cannot be ignored: harm to physical infrastructure, damage to important equipment, leak of critical information, injuries, and even death can occur merely because of the lack of sufficient security measures. Manufacturing enterprises and industrial organizations pay attention to these concerns and commit to making security a fundamental feature while keeping in mind the fact that the security of manufacturing systems is not a product or a single feature, it is an application of series of countermeasures on the whole process in general. Although the improvement in manufacturing systems is proceeding slowly, the IT security is advancing towards two-factor authentication, trained detection, and prevention models.

As Smart Manufacturing is moving towards autonomy and system complexity, it is becoming extremely important to invest in security experts and develop effective security solutions specially customized for the specific industry needs and demands.

The scientific community and industries should work together to build robust, reliable, and efficient security solutions which are able to cope with the increasing deployments and run-time requirements of Smart Manufacturing Systems. Cyber security in manufacturing is paramount for the future of durable systems, and more investments and research is required to keep advancing the right track to secure them.

Acknowledgements The authors would like to express their sincere gratitude and appreciation to the individuals who have contributed to the completion of this book chapter. First and foremost, we would like to thank all the VeNIT Lab members for providing an exceptional work environment. Their dedication and hard work have been instrumental in ensuring the success of this project. In particular, we would like to extend our heartfelt thanks to Fatmanur Ozdemir, Anilcan Bulut, Mert Ilik and Omer Faruk Caki for their invaluable assistance and support throughout the writing process. We are also grateful to Stefan Marksteiner and Leander Hoermann for providing their unique perspectives, insightful feedback, and expertise in their respective fields which have been invaluable in shaping the content of this chapter.

References

1. Phuyal, S., Bista, D., Bista, R.: Challenges, opportunities and future directions of smart manufacturing: a state of art review. *Sustain. Futur.* **2**, 100023 (2020)
2. Zheng, P., Wang, H., Sang, Z. et al.: Smart manufacturing systems for Industry 4.0: conceptual framework, scenarios, and future perspectives. *Front Mech. Eng.* **13**, 137–150 (2018)
3. Wang, B., Tao, F., Fang, X., et al.: Smart manufacturing and intelligent manufacturing: a comparative review. *Engineering* **7**, 738–757 (2021)
4. Xu, J., Kovatsch, M., Mattern, D., Mazza, F., Harasic, M., Paschke, A., Lucia, S.: A review on AI for smart manufacturing: deep learning challenges and solutions. *Appl. Sci.* **12**, 8239 (2022). <https://doi.org/10.3390/app12168239>
5. Parliment, E.: Smart manufacturing (2016). <https://ec.europa.eu/digital-singlemarket/smart-manufacturing>
6. Tuptuk, N., Hailes, S.: Security of smart manufacturing systems. *J. Manuf. Syst.* **47**, 93–106 (2018)
7. Gerodimos, A., Maglaras, L., Ferrag, M.A., Ayres, N., Kantzavelou, I.: IoT: communication protocols and security threats. *Internet Things Cyber-Phys. Syst.* **3**, 1–13 (2023)
8. Yang, X., Shu, L., Liu, Y., Hancke, G.P., Ferrag, M.A., Huang, K.: Physical security and safety of IoT equipment: a survey of recent advances and opportunities. *IEEE Trans. Ind. Inform.* **18**(7), 4319–4330 (2022)
9. Rossmann, M. et al.: Smart factories: how can manufacturers realize the potential of digital industrial revolution? *Cappgemini.com.* (2017)
10. Yee, T., Yen, N., Onathan, S.: Internet of things real-time monitoring of energy efficiency on manufacturing shop floors. *Procedia CIRP* **61**, 376–381 (2017)
11. Juarez, F.A.B.: Cybersecurity in an industrial internet of things environment (IIoT) challenges for standards systems and evaluation models. In: *Proceedings of the 2019 8th International Conference On Software Process Improvement (CIMPS)*, Leon, Guanajuato, Mexico, 23–25 October 2019, pp. 1–6
12. Wang, L., Shih, A.: Challenges in smart manufacturing. *J. Manuf. Syst.* **40**, 1 (2016)
13. Santhosh, N., et al.: Internet of Things (IoT) and industry 4.0 applications in manufacturing: a review. *IOP Conf. Ser.: Mater. Sci. Eng.* **764**, 012025 (2020)
14. Abosata, N., Al-Rubaye, S., Inalhan, G., Emmanouilidis, C.: Internet of things for system integrity: a comprehensive survey on security, attacks and countermeasures for Industrial Applications. *Sensors* **21**(11), 3654 (2021)

15. Yildirim, M., Demiroglu, U., Şenol, B.: An in-depth exam of IOT, IOT core components, IOT layers, and attack types. *Eur. J. Sci. Technol.* **21**, 1–10 (2021)
16. TATA Consultancy Services: the complete reimaginative force. TCS Global Trend Study (2015). <https://www.tcs.com/content/dam/tcs/pdf/Industries/global-trend-studies/iot/Internet-of-Things-The-Complete-Reimaginative-Force.pdf>. Accessed Aug 2022
17. Atzori, L., Iera, A., Morabito, G., Nitti, M.: The Social Internet of Things (SIOT)—when social networks meet the internet of things: concept, architecture and network characterization. *Comput. Netw.* **56**(16), 3594–3608 (2012)
18. Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., Zhao, W.: A survey on internet of things: architecture, enabling technologies, security and privacy, and applications. *IEEE Internet Things J.* **4**(5), 1125–1142 (2017)
19. Zhao, K., Ge, L.: A survey on the internet of things security. In: 2013 Ninth International Conference on Computational Intelligence and Security, vol. 14, pp. 663–667 (2013). <https://doi.org/10.1109/CIS.2013.175>.
20. Obaidat, M.A., Obeidat, S., Holst, J., Al Hayajneh, A., Brown, J.: A comprehensive and systematic survey on the internet of things: security and privacy challenges, security frameworks, enabling technologies, threats, vulnerabilities and countermeasures. *Computers* **9**, 44 (2020). <https://doi.org/10.3390/computers9020044>
21. Gebende, E., Campegianni, P., Czachórski, T., Katsikas, S.K., Komnios, I., Romano, L., Tzouvaras, D. (eds.): Security in computer and information sciences. *Commun. Comput. Inf. Sci* (2018)
22. Gelenbe, E., Kadioglu, Y.M.: Energy life-time of wireless nodes with network attacks and mitigation. In: 2018 IEEE International Conference on Communications Workshops (ICC Workshops) (2018)
23. Lomné, V., Prouff, E., Roche, T.: Behind the scene of side channel attacks. In: Sako, K., Sarkar, P. (eds.) *Advances in Cryptology—ASIACRYPT 2013*. ASIACRYPT 2013. Lecture Notes in Computer Science, vol. 8269, pp. 542–559. Springer (2013)
24. Gokhale, P., Bhat, O., Bhat, S.: Introduction to IOT. *Int. Adv. Res. J. Sci.* **5**(1) (2018)
25. Syed, N.F., Baig, Z., Ibrahim, A., Valli, C.: Denial of service attack detection through machine learning for the IoT. *J. Inf. Telecommun.* **4**(4), 482–503 (2020)
26. Fazeldehkordi, E., Amiri, I.S.: Wormhole attack. In: Akanbi, O.A. (ed.) *A Study of Black Hole Attack Solutions*, 1st edn., pp. 51–52. Syngress, Waltham, Massachusetts (2016)
27. Abdullahi, M., Baashar, Y., Alhussian, H., Alwadain, A., Aziz, N., Capretz, L.F., Abdulkadir, S.J.: Detecting cybersecurity attacks in internet of things using artificial intelligence methods: a systematic literature review. *Electronics* **11**, 198 (2022). <https://doi.org/10.3390/electronics11020198>
28. Baronti, P., Pillai, P., Chook, V.W.C., Chessa, S., Gotta, A., Hu, Y.F.: Wireless sensor networks: a survey on the state of the art and the 802.15.4 and zigbee standards. *Comput. Commun.* **30**(7), 1655–1695 (2007)
29. Sreeram, I., Vuppala, V.: HTTP flood attack detection in application layer using machine learning metrics and bio inspired bat algorithm. *Appl. Comput. Inf.* **15**, 59–66 (2019)
30. Halfond, W.G., Viegas, J., Orso, A.: A classification of SQL-injection attacks and countermeasures. In: *Proceedings of the IEEE International Symposium on Secure Software Engineering*, p. 1 (2006)
31. Shorey, T., Subbaiah, D., Goyal, A., Sakxena, A., Mishra, A.K.: Performance comparison and analysis of Slowloris, goldeneye and Xerxes DDos Attack tools. In: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (2018)
32. Cambiaso, E., Papaleo, G., Chiola, G., Aiello, M.: Slow dos attacks: definition and categorisation. *Int. J. Trust. Manag. Comput. Commun.* **1**(3/4), 300 (2013)
33. Maggi, F., Pogliani, M.: Attacks on smart manufacturing systems. *Trend Micro Research* (2020)
34. Zin, S.M., Anuar, N.B., Kiah, M.L.M., Ahmady, I.: Survey of secure multipath routing protocols for WSNs. *J. Netw. Comput. Appl.* **55**, 123–53 (2015)

35. Vogt, H.: Exploring message authentication in sensor networks. In: 1st European Workshop on Security in Ad Hoc and Sensor Networks (ESAS 2004) (2004)
36. El Kaissi, R.Z., Kayssi, A., Chehab, A., Dawy, Z.: Dawwsen: a defense mechanism against wormhole attacks in wireless sensor networks [Ph.D. dissertation]. American University of Beirut, Department of Electrical and Computer Engineering (2005)
37. Hu, L., Evans, D.: Using directional antennas to prevent wormhole attacks. NDSS 241–245 (2004)
38. Hu, Y.-C., Perrig, A., Johnson, D.B., Packet leashes: a defense against wormhole attacks in wireless networks. In: INFOCOM: Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, vol. 3, pp. 1976–1986. IEEE (2003)
39. Patil, S., Chaudhari, S.: DoS attack prevention technique in wireless sensor networks. Procedia Comput. Sci. **79**, 715–721 (2016). <https://doi.org/10.1016/j.procs.2016.03.091>
40. Liu, H.I., Chang, K.C.: Defending systems against tilt DDos attacks. Telecommun. Syst. Serv. Appl. (TSSA) Oct 20–21, 22–27 (2011)
41. Kambourakis, G., Moschos, T., Geneiatakis, D., Gritzalis, S.: Detecting DNS amplification attacks. In: Samarati, P., De Capitani di Vimercati, S. (eds.) Critical Information Infrastructures Security. Lecture Notes in Computer Science, vol. 5141, pp. 185–196. Springer, Berlin (2008)
42. Mukherjee, A.: Physical-layer security in the internet of things: sensing and communication confidentiality under resource constraints. Proc. IEEE **103**(10), 1747–1761 (2015). <https://doi.org/10.1109/JPROC.2015.2466548>
43. Xu, W., Ma, K., Trappe, W., Zhang, Y.: Jamming sensor networks: attack and defense strategies. IEEE Netw. **20**(3), 41–47 (2006). <https://doi.org/10.1109/MNET.2006.1637931>
44. Babar, S.D., Prasad, N.R., Prasad, R.: Jamming attack: behavioral modeling and analysis. Wirel. VITAE 1–5 (2013). <https://doi.org/10.1109/VITAE.2013.6617054>
45. Spacehuhn Technologies: *esp8266_deauther*. (2020). https://github.com/SpacehuhnTech/esp8266_deauther

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

