

*International Conference on*



*Computational Intelligence in  
Security for Information Systems*

*International Conference on*



*EUROPEAN  
Transnational Education*

Pablo García Bringas, Hilde Pérez García, Francisco Javier Martínez de Pisón, José Ramón Villar Flecha, Alicia Troncoso Lora, Enrique A. de la Cal, Álvaro Herrero, Francisco Martínez Álvarez, Giuseppe Psaila, Héctor Quintián, Emilio Corchado (Eds.)

---

## **International Joint Conference**

**15<sup>th</sup> International Conference on Computational Intelligence in Security for Information Systems (CISIS 2022)**

**13<sup>th</sup> International Conference on European Transnational Education (ICEUTE 2022)**

## Editors

Pablo García Bringas  
University of Deusto  
Bilbao, Spain

Hilde Pérez García  
University of Leon  
León, Spain

Francisco Javier Martínez de Pisón  
University of La Rioja  
Logroño, Spain

José Ramón Villar Flecha  
University of Oviedo  
Oviedo, Spain

Alicia Troncoso Lora  
Pablo Olavide University  
Seville, Spain

Enrique A. de la Cal  
University of Oviedo  
Oviedo, Spain

Álvaro Herrero  
University of Burgos  
Burgos, Spain

Francisco Martínez Álvarez  
Pablo Olavide University  
Seville, Spain

Giuseppe Psaila  
University of Bergamo  
Bergamo, Italy

Héctor Quintián  
University of A Coruña  
A Coruña, Spain

Emilio Corchado  
University of Salamanca  
Salamanca, Spain

---

## Preface

This volume contains accepted papers pre-sented at the *15<sup>th</sup> International Conference on Computational Intelligence in Security for Information Systems* (CISIS 2022) and the *13<sup>th</sup> International Conference on EUropean Transnational Education* (ICEUTE 2022). These conferences were held in the beautiful city of Salamanca, Spain, in September 2022.

The aim of the CISIS 2022 conference is to offer a meeting opportunity for academic and industry-related researchers belonging to the various, vast communities of Computational Intelligence, Information Security, and Data Mining. The need for intelligent, flexible behaviour by large, complex systems, especially in mission-critical domains, is intended to be the catalyst and the aggregation stimulus for the overall event.

After a through peer-review process, the CISIS 2022 International Program Committee selected 20 papers, which are published in these conference proceedings. In this edition, three special session were organized: Cybersecurity in Future Connected Societies, Cybersecurity and Trusted Supply Chains of ICT and Intelligent Solutions for Cybersecurity Systems.

The aim of ICEUTE 2022 conference is to offer a meeting point for people working on transnational education within Europe. It provides a stimulating and fruitful forum for presenting and discussing the latest works and advances on transnational education within European countries.

In the case of ICEUTE 2022, the International Program Committee selected 5 papers, which are also published in these conference proceedings.

The selection of papers was extremely rigorous to maintain the high quality of the conferences. We want to thank the members of the Pro-gram Committees for their hard work during the reviewing process. This is a crucial process for creating a high-standard conference; the CISIS and ICEUTE conferences would not exist without their help.

CISIS 2022 and ICEUTE 2022 enjoyed outstanding keynote speeches by distinguished guest speakers: Prof. Ajith Abraham, Director of Machine Intelligence Research Labs (MIR Labs), Prof. Guy De Tré head of the research group on Database, Document, and Content Management (DDCM) at Ghent University, Belgium, and Felix Barrio General Director at INCIBE (Spain).

CISIS 2022 has teamed up with “Logic Journal of the IGPL” (Oxford University Press), and Expert Systems (Wiley) for a suite of special issues, including selected papers from CISIS 2022.

Particular thanks go as well to the conference's main sponsors, Startup Olé, the CYL-HUB project financed with NEXT-GENERATION funds from the European Union, the Ministry of Labor and Social Economy, the Recovery, Transformation and Resilience Plan and the State Public Employment Service, channelled through the Junta de Castilla y León, BISITE research group at the University of Salamanca, CTC research group at the University of A Coruña, and the University of Salamanca. They jointly contributed in an active and constructive manner to the success of this initiative.



We would like to thank all the special session organizers, contributing authors, as well as the members of the Program Committees and the Local Organizing Committee for their hard and highly valuable work. Their work has helped to contribute to the success of the CISIS and ICEUTE 2022 events.

The editors

Pablo García Bringas  
Hilde Pérez García  
Francisco Javier Martínez de Pisón  
José Ramón Villar Flecha  
Alicia Troncoso Lora  
Enrique A. de la Cal  
Álvaro Herrero  
Francisco Martínez Álvarez  
Giuseppe Psaila  
Héctor Quintián  
Emilio Corchado

September, 2022

# CISIS 2022

---

## Organization

### General Chair

Emilio Corchado	University of Salamanca, Spain
-----------------	--------------------------------

### Program Committee Chair

Pablo García Bringas	University of Deusto, Spain
Hilde Pérez García	University of León, Spain
Francisco Javier Martínez de Pisón	University of La Rioja, Spain
José Ramón Villar Flecha	University of Oviedo, Spain
Alicia Troncoso Lora	Pablo Olavide University, Spain
Enrique A. de la Cal	University of Oviedo, Spain
Álvaro Herrero	University of Burgos, Spain
Francisco Martínez Álvarez	Pablo Olavide University, Spain
Giuseppe Psaila	University of Bergamo, Italy
Héctor Quintián	University of A Coruña, Spain
Emilio Corchado	University of Salamanca, Spain

### Program Committee

Adam Wójtowicz	Poznań University of Economics and Business, Poland
Agustin Martin Muñoz	C.S.I.C., Spain
Alberto Peinado	University of Malaga, Spain
Álvaro Herrero Cosío	University of Burgos, Spain
Álvaro Michelena Grandío	University of A Coruña, Spain
Amparo Fuster-Sabater	C.S.I.C., Spain
Anca Avram	Technical University of Cluj-Napoca, North University Center at Baia Mare, Romania
Andreea Vescan	Babes-Bolyai University, Cluj-Napoca, Romania
Andrysiak Tomasz	University of Technology and Life Sciences (UTP), Poland
Angel Arroyo	University of Burgos, Spain
Angel Martin Del Rey	University of Salamanca, Spain
Araceli Queiruga-Dios	University of Salamanca, Spain
Borja Sanz	University of Deusto, Spain
Carlos Cambra	University of Burgos, Spain
Carlos Pereira	ISEC, Portugal
Carmen Benavides	University of León, Spain
Ciprian Pungila	West University of Timișoara, Romania
Cristina Alcaraz	University of Malaga, Spain
Daniel Urda	University of Burgos, Spain
David Arroyo	C.S.I.C., Spain
David Garcia-Retuerta	University of Salamanca, Spain
Eduardo Solteiro Pires	UTAD University, Portugal
Elena Hernández Nieves	University of Salamanca, Spain
Emilia Cioroica	Fraunhofer IESE, Germany
Eneko Osaba	TECNALIA Research & Innovation, Spain

Enrique Onieva	University of Deusto, Spain
Esteban Jove	Univeristy of A Coruña, Spain
Esteban Maurin	University of Salamanca, Spain
Fernando Ribeiro	EST, Portugal
Fernando Tricas	University of Zaragoza, Spain
Francisco Martínez-Álvarez	Pablo de Olavide University, Spain
Francisco Zayas-Gato	University of A Coruña, Spain
Guillermo Morales-Luna	CINVESTAV-IPN, Mexico
Héctor Quintián	University of A Coruña, Spain
Hugo Sanjurjo-González	University of Deusto, Spain
Hugo Scolnik	ARSAT SA, Argentina
Igor Santos	Mondragon Unibertsitatea, Spain
Ioannis Sorokos	Fraunhofer IESE, Germany
Isaias Garcia	University of León, Spain
Javier Nieves	Azterlan, Spain
Jesús Díaz-Verdejo	University of Granada, Spain
Jose A. Onieva	University of Malaga, Spain
Jose Barata	NOVA University Lisbon, Portugal
Jose Carlos Metrolho	IPCB, Portugal
José Francisco Torres Maldonado	Pablo de Olavide University, Spain
Jose Luis Calvo-Rolle	University of A Coruña, Spain
José Luis Casteleiro-Roca	University of A Coruña, Spain
José Luis Imaña	Complutense University of Madrid, Spain
Jose M. Molina	University Carlos III of Madrid, Spain
Jose Manuel Lopez-Guede	University of the Basque Country, Spain
Josep Ferrer	University of les Illes Balears, Spain
Juan J. Gude	University of Deusto, Spain
Juan Jesús Barbarán	University of Granada, Spain
Juan Pedro Hecht	University of Buenos Aires, Argentina
Lidia Sánchez-González	Universidad de León, Spain
Luis Alfonso Fernández Serantes	FH-Joanneum University of Applied Sciences, Spain
Luis Hernandez Encinas	C.S.I.C., Spain
Manuel Castejón-Limas	Universidad de León, Spain
Manuel Graña	University of the Basque Country, Spain
Marc Ohm	Universität Bonn, Germany
Marina Castaño Ishizaka	University of Salamanca, Spain
Michal Choras	ITTI, Poland
Michal Wozniak	Wroclaw University of Technology, Poland
Míriam Timiraos Díaz	University of A Coruña, Spain
Nuño Basurto	University of Burgos, Spain
Oliviu Matei	North University of Baia Mare, Romania
Oscar Llorente-Vazquez	University of Deusto, Spain
Ovidiu Cosma	Technical University Cluj Napoca, Romania
Pablo García Bringas	University of Deusto, Spain
Petrica Pop	Technical University of Cluj-Napoca, North University Center at Baia Mare, Romania
Rafael Alvarez	University of Alicante, Spain
Rafael Corchuelo	University of Seville, Spain
Raúl Durán	University of Alcalá, Spain
Robert Burduk	University Wroclaw, Poland

Roberto Casado-Vara	University of Burgos, Spain
Rogério Dionísio	Polytechnic Institute of Castelo Branco, Portugal
Rudolf Erdei	Holisun SRL, Romania
Salvador Alcaraz	Miguel Hernandez University, Spain
Sorin Stratulat	Université de Lorraine, France
Wenjian Luo	Harbin Institute of Technology, China
Wilson Rojas	El Bosque University, Colombia

## **CISIS 2022: Special Sessions**

### **Cybersecurity in Future Connected Societies**

#### **Program Committee**

Koji Ishibashi (Organizer)	University of Electro-Communications, Japan
Víctor Gayoso Martínez (Organizer)	C.S.I.C., Spain
Kazuo Sakiyama	The University of Electro-Communications, Japan
Lorenzo Mucchi	Dept. of Information Engineering, University of Florence, Italy
Mitsugu Iwamoto	University of Electro-Communications, Japan
Naoki Ishikawa	Yokohama National University, Japan
Satyanarayana Vuppala	CITI Bank, Ireland
Seyedomid Motlagh	RWTH Aachen University, Germany
Slobodan Petrovic	Norwegian University of Science and Technology, Norway

### **Cybersecurity and Trusted Supply Chains of ICT**

#### **Program Committee**

Anca Andreica (Organizer)	Babes-Bolyai University, Romania
Antonio Skarmeta Gomez (Organizer)	Universidad de Murcia, Spain
Camelia Chira (Organizer)	Babes-Bolyai University, Romania
Eda Marchetti (Organizer)	Isti-cnr, Italy
Ileana Buhan (Organizer)	RU, Netherlands
Jose Barata (Organizer)	UNINOVA, Portugal
Oliviu Matei (Organizer)	HOLISUN, Romania
Ovidiu Cosma (Organizer)	Technical Univesity of Cluj-Napoca, Romania
Petrica Pop (Organizer)	Technical Univesity of Cluj-Napoca, Romania
Ricardo Silva Peres (Organizer)	Universidade NOVA de Lisboa, Portugal

### **Intelligent Solutions for Cybersecurity**

#### **Program Committee**

Bernabé Dorronsoro (Organizer)	University of Cádiz, Spain
Daniel Urda Muñoz (Organizer)	University of Burgos, Spain
Meelis Kull (Organizer)	University of Tartu, Estonia
Roberto Magán Carrión (Organizer)	University of Granada, Spain

Ángel Canal Alonso	University of Salamanca, Spain
Carlos Cambra	University of Burgos, Spain
Carlos Fernandes	Universidade de Lisboa, Portugal
Eduardo Feo Flushing	Carnegie Mellon University in Qatar, Qatar
Esther-Lydia Silva-Ramírez	University of Cádiz, Spain
Jamal Toutouh	Massachusetts Institute of Technology, United States
Javier González-Enrique	University of Cádiz, Spain
José Manuel García Nieto	University of Málaga, Spain
Lipika Deka	De Montfort University, United Kingdom
Margarita Robles Carrillo	University of Granada, Spain
Nuño Basurto	University of Burgos, Spain
Pablo García	University of Granada, Spain
Pablo Ruiz	OriGen AI, Spain
Rafael A. Rodríguez-Gómez	International University of La Rioja, Spain
Rafael M. Luque-Baena	University of Extremadura, Spain

## **CISIS 2022 Organising Committee Chairs**

Emilio Corchado	University of Salamanca, Spain
Héctor Quintián	University of A Coruña, Spain

## **CISIS 2022 Organising Committee**

Álvaro Herrero Cosío	University of Burgos, Spain
José Luis Calvo Rolle	University of A Coruña, Spain
Ángel Arroyo	University of Burgos, Spain
Daniel Urda	University of Burgos, Spain
Nuño Basurto	University of Burgos, Spain
Carlos Cambra	University of Burgos, Spain
Esteban Jove	University of A Coruña, Spain
José Luis Casteleiro Roca	University of A Coruña, Spain
Francisco Zayas Gato	University of A Coruña, Spain
Álvaro Michelena	University of A Coruña, Spain
Míriam Timiraos Díaz	University of A Coruña, Spain

# ICEUTE 2022

---

## Organization

### General Chair

Emilio Corchado	University of Salamanca, Spain
-----------------	--------------------------------

### Program Committee Chair

Pablo García Bringas	University of Deusto, Spain
Hilde Pérez García	University of León, Spain
Francisco Javier Martínez de Pisón	University of La Rioja, Spain
José Ramón Villar Flecha	University of Oviedo, Spain
Alicia Troncoso Lora	Pablo Olavide University, Spain
Enrique A. de la Cal	University of Oviedo, Spain
Álvaro Herrero	University of Burgos, Spain
Francisco Martínez Álvarez	Pablo Olavide University, Spain
Giuseppe Psaila	University of Bergamo, Italy
Héctor Quintián	University of A Coruña, Spain
Emilio Corchado	University of Salamanca, Spain

### Program Committee

Alessandra Raffaetà	Ca' Foscari University of Venezia, Italy
Álvaro Michelena Grandío	University of A Coruña, Spain
Ana Maria Lara Palma	University of Burgos, Spain
Ana Rosa Pereira Borges	ISEC-Coimbra Polytechnic Institute, Portugal
Andreea Vescan	Babes-Bolyai University, Romania
Angel Arroyo	University of Burgos, Spain
Antonio Morales-Esteban	University of Seville, Spain
Carlos Cambra	University of Burgos, Spain
Carlos Pereira	ISEC, Portugal
Carmen Benavides	University of León, Spain
Daniel Urda	University of Burgos, Spain
Daniela Zaharie	West University of Timisoara, Romania
David Becerra Alonso	University of Loyola, Spain
Diego Pedro Pinto Roa	Universidad Nacional de Asunción, Paraguay
Domingo S. Rodríguez Baena	Pablo de Olavide University, Spain
Dragan Simic	University of Novi Sad, Serbia
Eduardo Solteiro Pires	UTAD University, Portugal
Eloy Irigoyen	University of the Basque Country, Spain
Esteban Jove	University of A Coruña, Spain
Estibaliz Apiñaniz	University of the Basque Country, Spain
Federico Divina	Pablo de Olavide University, Spain
Francisco Gomez Vela	Pablo de Olavide University, Spain
Francisco Martínez-Álvarez	Pablo de Olavide University, Spain
Francisco Zayas-Gato	University of A Coruña, Spain

Guillermo Santamaria	CONACYT-INEEL, Mexico
Héctor Quintián	University of A Coruña, Spain
Hugo Sanjurjo-González	University of Deusto, Spain
Isaias Garcia	University of León, Spain
J. David Nuñez-Gonzalez	University of the Basque Country, Spain
Jairo Ortiz-Revilla	University of Burgos, Spain
Jorge Barbosa	ISEC, Portugal
José Francisco Torres Maldonado	Pablo de Olavide University, Spain
Jose Luis Calvo-Rolle	University of A Coruña, Spain
José Luis Casteleiro-Roca	University of A Coruña, Spain
Jose Luis Vazquez Noguera	Universidad Nacional de Asunción, Paraguay
José Manuel Galán	University of Burgos, Spain
Jose Manuel Lopez-Guede	University of the Basque Country, Spain
José-Lázaro Amaro-Mellado	University of Seville, Spain
Juan J. Gude	University of Deusto, Spain
Juan Pavón	Complutense University of Madrid, Spain
Julián Estévez	University of the Basque Country, Spain
Laura Fernández-Robles	University of León, Spain
Laura Melgar-García	Pablo de Olavide University, Spain
Lidia Sánchez-González	University of León, Spain
Luis Alfonso Fernández Serantes	FH-Joanneum University of Applied Sciences, Spain
Manuel Castejón-Limas	University of León, Spain
María Fernanda Gambarini	Francisco de Vitoria University, Spain
Maria Jose Marcelino	University of Coimbra, Portugal
Maria Teresa Godinho	Polytechnic Institute of Beja, Portugal
Maria Victoria Requena	University of Seville, Spain
Marián Queiruga-Dios	Francisco de Vitoria University, Spain
Miguel Ángel Queiruga-Dios	University of Burgos, Spain
Miguel Carriegos	RIASC, Spain
Miguel Garcia Torres	Pablo de Olavide University, Spain
Miriam Timiraos Díaz	University of A Coruña, Spain
Nuño Basurto	University of Burgos, Spain
Pablo García Bringas	University of Deusto, Spain
Paola Clara Leotta	University of Catania, Italy
Paulo Moura Oliveira	UTAD University, Portugal
Pedro Mauricio Acosta Castellanos	Santo Tomás University, Colombia
Ramona Mihaila	Dimitrie Cantemir Christian University, Romania
Richard Duro	University of A Coruña, Spain
Sebastian Alberto Grillo	Universidad Autonoma de Asunción, Paraguay
Soraya Muñoz Pérez	University Francisco de Vitoria, Spain
Sorin Stratulat	Université de Lorraine, France
Viviana Elizabeth Jiménez Chaves	Universidad Nacional de Itapúa, Paraguay

## ICEUTE 2022 Organising Committee Chairs

Emilio Corchado	University of Salamanca, Spain
Héctor Quintián	University of A Coruña, Spain

## **ICEUTE 2022 Organising Committee**

Álvaro Herrero Cosío	University of Burgos, Spain
José Luis Calvo Rolle	University of A Coruña, Spain
Ángel Arroyo	University of Burgos, Spain
Daniel Urda	University of Burgos, Spain
Nuño Basurto	University of Burgos, Spain
Carlos Cambra	University of Burgos, Spain
Esteban Jove	University of A Coruña, Spain
José Luis Casteleiro Roca	University of A Coruña, Spain
Francisco Zayas Gato	University of A Coruña, Spain
Álvaro Michelena	University of A Coruña, Spain
Miriam Timiraos Díaz	University of A Coruña, Spain



# INDEX

<b>CISIS Conference .....</b>	<b>16</b>
<b>CISIS Applications .....</b>	<b>17</b>
<b>Analysis of long-range forecast strategies for IoT on urban water consumption prediction task .....</b>	<b>18</b>
<i>Krzysztof Pałczyński, Tomasz Andrysiak, Marcin Głowacki, Michał Kierul and Tomasz Kierul</i>	
<b>Genetic Algorithm based Aggregation for Federated Learning in Industrial Cyber Physical Systems .....</b>	<b>28</b>
<i>Souhila Badra Guendouzi, Samir Ouchani and Mimoun Malki</i>	
<b>Hand SOS gesture detection by computer vision .....</b>	<b>38</b>
<i>Roberto Viejo-López, Virginia Riego Del Castillo and Lidia Sánchez-González</i>	
<b>Prediction of Smart Energy Meter Network Traffic Features for Anomaly Detection .....</b>	<b>46</b>
<i>Tomasz Andrysiak and Łukasz Saganowski</i>	
<b>An anomaly detection approach for realtime identification systems based on centroids .....</b>	<b>56</b>
<i>Álvaro Michelena Grandío, Francisco Zayas-Gato, Esteban Jove, José Luis Casteleiro-Roca, Héctor Quintián, Oscar Fontenla-Romero and Jose Luis Calvo-Rolle</i>	
<b>Powerful Biogeography-Based Optimization algorithm with local search mechanism for Job Shop Scheduling Problem with additional constraints .....</b>	<b>68</b>
<i>Madiha Harrabi, Olfa Belkahla Driss and Khaled Ghedira</i>	
<b>Dimensionality-Reduction Methods for the Analysis of Web Traffic .....</b>	<b>78</b>
<i>Nuño Basurto, Álvaro Michelena, Daniel Urda, Hector Quintián, José Luis Calvo-Rolle, and Álvaro Herrero</i>	
<b>Special Session on Cybersecurity in Future Connected Societies .....</b>	<b>89</b>
<b>About the Fujisaki-Okamoto Transformation in the Code-based Algorithms of the NIST Post-Quantum Call .....</b>	<b>90</b>
<i>Miguel Ángel González de la Torre and Luis Hernández Encinas</i>	
<b>Analysis of Secret Key Agreement Protocol for Massive MIMO Systems .....</b>	<b>100</b>
<i>Seiya Otsuka, Hiroki Imori, Kengo Ando, Giuseppe Abreu, Koji Ishibashi and Naoki Ishikawa</i>	
<b>Efficient implementation of stream cipher SNOW3G for resource-constrained devices .....</b>	<b>110</b>
<i>Guillermo Cotrina, Alberto Peinado and Andrés Ortiz</i>	

<b>State of the art of cybersecurity in cooperative, connected and automated mobility.....</b>	<b>120</b>
<i>Óscar Castillo Campo, Víctor Gayoso Martínez, Luis Hernandez Encinas, Agustin Martin Muñoz and Roberto Álvarez Fernández</i>	
<b>Cryptographic protocols in advanced metering infrastructures in smart grids .....</b>	<b>130</b>
<i>Luis Hernández-Álvarez, Juan J. Bullón Pérez and Araceli Queiruga-Dios</i>	
<b>Special Session on Cybersecurity and Trusted Supply Chains of ICT ....</b>	<b>140</b>
<b>Orchestrator Architecture and Communication Methodology for Flexible Event Driven Message Based Communication.....</b>	<b>141</b>
<i>Rudolf Erdei, Daniela Delinschi, Emil Pasca and Oliviu Matei</i>	
<b>A comparative study of Machine Learning algorithms for the detection of vulnerable Python libraries .....</b>	<b>153</b>
<i>Laura Pérez-Vilarelle, Eva Sotos Martínez and Javier Yépez Martínez</i>	
<b>Evaluation of the reliability index of IP addresses in reputation lists.....</b>	<b>163</b>
<i>Alberto Miranda-García, Ignacio Samuel Crespo-Martínez, Ángel Manuel Guerrero-Higueras and Vicente Matellán-Olivera</i>	
<b>Forecasting the Number of Bugs and Vulnerabilities in Software Components using Neural Network Models .....</b>	<b>173</b>
<i>Ovidiu Cosma, Petrica Pop, Cosmin Sabo and Laura Cosma</i>	
<b>Special Session on Intelligent Solutions for Cybersecurity Systems ....</b>	<b>183</b>
<b>Reinforcement Learning model free with GLIE Monte-Carlo on policy update for network topology discovery .....</b>	<b>184</b>
<i>Roberto Casado-Vara, Marcos Severt Silva, Angel Martin Del Rey, Héctor Quintián and Jose Luis Calvo-Rolle</i>	
<b>Obfuscating LLVM Intermediate Representation Source Code with NSGA-II.....</b>	<b>194</b>
<i>Juan Carlos de la Torre, José Miguel Aragón-Jurado, Javier Jareño, Sébastien Varrette and Bernabé Dorronsoro</i>	
<b>A Deep Learning-based approach for Mimicking Network Topologies: the Neris Botnet as a Case of Study .....</b>	<b>204</b>
<i>Francisco Álvarez Terribas, Roberto Magán-Carrión, Gabriel Macia-Fernandez and Antonio Mora</i>	
<b>Evaluating Classifiers' Performance to Detect Attacks inWebsite Traffic .....</b>	<b>214</b>
<i>Daniel Urda, Nuño Basurto, Meelis Kull and Álvaro Herrero</i>	

## **ICEUTE Conference..... 225**

### **Evaluation of an interactive guide for robotics self-learning ..... 226**

*Álvaro Ovejero, Jesús Enrique Sierra-García and Matilde Santos*

### **Gamifying the classroom for the acquisition of skills associated with Machine Learning: a two-year case study..... 235**

*Antonio Manuel Durán-Rosal, David Guijo-Rubio, Víctor Vargas, Antonio Manuel Gómez-Orellana, Pedro Antonio Gutiérrez and Juan Carlos Fernández*

### **Hackathon in teaching: applying machine learning to Life Sciences tasks ..... 246**

*David Guijo-Rubio, Víctor Manuel Vargas, Javier Barbero-Gómez, Jose Vicente Die and Pablo González-Moreno*

### **Digital platforms for education. The case of e4you ..... 256**

*Javier Parra-Domínguez, Sergio Manzano, Susana Herrero and Pablo Chamoso*

### **3D Virtual Laboratory for Control Engineering using Blended Learning Methodology ..... 266**

*Francisco Zayas-Gato, Álvaro Michelena, Esteban Jove, José-Luis Casteleiro-Roca, Héctor Quintián, Elena Arce and José Luis Calvo-Rolle*

# **CISIS Conference**

# **CISIS Applications**

# Analysis of long-range forecast strategies for IoT on urban water consumption prediction task

Krzysztof Pałczyński, Tomasz Andrysiak, Marcin Głowacki, Michał Kierul,  
Tomasz Kierul

**Abstract:** With the rapid development of technology, researchers worldwide have applied the Internet of Things to effectively transmit and monitor water levels and detect anomalies in real time. The data obtained enables numerical methods to predict water consumption as well. In the presented paper, an attempt has been made to predict water consumption for various problems forward using dedicated models and a system using an iterative approach. For this purpose, neural network algorithms such as Random Forest, XGBoost, Decision Tree, and Support Vector Regression were tested and used to train the prediction models. The results presented allowed to indicate the difference between the examined methods through the Mean Absolute Percentage Error of prediction. The used set of algorithms allowed to show the problem of estimating water prediction from different points of view. Thus, determining the tested systems' seasonality and short-term and long-term trends. This allowed to choose the two best algorithms, one that needs less computational power to work; this seems to be a better solution.

**Keywords:** Prediction Systems, Machine Learning, Water Consumption Forecast

---

Krzysztof Pałczyński  
Bydgoszcz University of Science and Technology, email: krzpal004@pbs.edu.pl

Tomasz Andrysiak  
Bydgoszcz University of Science and Technology, email: andrys@pbs.edu.pl

Marcin Głowacki  
Bydgoszcz University of Science and Technology email: marglo010@pbs.edu.pl

Michał Kierul  
Research and Development Center SOFTBLUE S.A., email: mkierul@softblue.pl

Tomasz Kierul  
Research and Development Center SOFTBLUE S.A., email: tkierul@softblue.pl

## 1 Introduction

The issues involved in managing water distribution networks are based on a large number of processes. The demand for water for service or residential purposes affects the amount of water withdrawn. In both cases, it is necessary to estimate consumption based on the services provided by the company or the inhabitants' behavior. The topic of demand analysis is essential due to the number of variables depending on the type of facility to which water is supplied, which provides the basis for the proper design of water management strategies. Forecasts are of great importance in the context of decisions related to introducing procedures to optimize water treatment plants or pumping stations.

In the presented work, an attempt was made to predict water consumption forward by different numbers of probes. The water consumption signal under study was divided into 15-minute intervals, with one step covering the 15 minutes elapsed. For the long-range prediction calculations, systems were used that could attempt to estimate water consumption using two methods; using specially trained models for forecasting each step into the future and using only one model trained for next-value prediction and using it iteratively to find demanded forecasts. The first involved a dedicated model designed to look one, two, or several time steps ahead. The second worked based on performing predictions using earlier forecasting results. This system used an iterative approach involving only one model, with each forward time step depending on the previous one. It goes forward when the prediction turns out to be true after predicting the following sample. The comparison between systems was established based on their mean errors and suitability for Internet of Things (IoT) purposes.

The thesis in this article is that long-range prediction obtained from the chain of short-range predictions made using the results of previous forecasts is an acceptable predictive strategy for specific time ranges. IoT applications would benefit from having only one trained model and using it for long-range predictions instead of keeping many models, each trained for forecasting on its specific time range.

This paper aims to use selected algorithms to train forecasting models to predict the following sample and evaluate their performance using Mean Absolute Percentage Error (MAPE). Short and long-term forecast results of all tested algorithms are presented, highlighting the performance of iterative systems in short-term and dedicated models in long-term forecasts.

The data came from the waterworks and was taken thanks to a telemetry overlay on the water meters using IoT. In simple terms, it is a system of electronic devices that can automatically communicate and exchange data over a network without human intervention. IoT-based solutions in water level monitoring and storage are a novelty, as well as the analysis of long-range forecasting strategies suited for low-computational power devices commonly used for IoT purposes. The proper use of the received data can help to reduce the occurrence of anomalies and mitigate the possible effects of a water crisis [1].

This paper is organized as follows: an introduction, section 2, in which work on water use forecasting to detect anomalies and determine the causes of fluctuations in consumption is discussed. The third section includes a detailed presentation of the systems used and the rationale for choosing particular algorithms. Section 4 discusses the experimental results. The paper concludes with a sub-summary.

## 2 Related work

Several approaches to the problem of water demand forecasting, or drought forecasting, can be found in the literature. Typically, statistical models have been used to estimate losses and detect undesirable anomalies.

In paper [2], an algorithm was tested to detect and locate water leaks at an early stage of demand forecasting. The algorithm used was based on pattern similarity, and the demonstration area of the project was a medium-sized city in Spain. The algorithm was tested in three different scenarios over two years (2014-2016). The scenarios were selected based on other profiles, including industrial areas, city centers, and suburbs. The prediction analysis yielded good results. The predictions indicated the most accurate results for the downtown area due to the many people living there. The authors suggest that the method is suitable for predicting daily water flow. The signals proved highly unpredictable for the time range studied due to water flow spikes caused by anomalous days such as holidays. As the historical database of similar past situations increases, it is possible to increase the prediction accuracy. The task and problems related to water flow spikes are identical to the subject of our work. However, our analyses focused on the approaches to long-term prediction and divergence rate of forecast from the base signal.

Another way of determining the unit water demand and consumption distribution was addressed in work [3], which focused on the hourly analysis of water consumption in selected residential buildings after declaring an epidemic emergency in the country. The objective was to verify the application of clustering using mean class clustering. The research objects were three similar five-story residential buildings in the same housing estate in Bydgoszcz. The study time included 464 measurement days, of which 276 days covered the period before the COVID-19 pandemic and 188 during the pandemic, which was included in the calculations. The application of cluster analysis through clustering by the mean squared method allowed us to develop a pattern of hourly water demand by working and non-working days. The results clearly show the profound changes in daily water demand patterns exerted by the COVID-19 pandemic. Continuous monitoring is needed to obtain up-to-date consumption patterns. The averaged histograms obtained can be used to determine nodal water consumption in the mathematical modeling of water supply networks. The data analyzed in this work is also from times of the COVID pandemic. However, in this work, the predictions were conducted using machine learning algorithms and Deep Learning.



Another interesting paper [4] addressed clustering and Support Vector Regression for water demand forecasting and the possibility of anomaly detection. The data used for the analysis was obtained from a municipal water company in Milan, Italy, from a Supervisory Control And Data Acquisition (SCADA) system and 26 sensors, i.e., meters used at the individual level. The proposed approach offers several innovations, from increasing the forecasting accuracy at the municipal and individual classes to using a time series clustering procedure on the raw data. The second innovation is to learn different Support Vector Machine (SVM) regression models separately for each cluster and each hour of water consumption. This results in the optimization of the pumping schedule and the ability to identify online anomalies resulting from wise meter faults, cyber-physical attacks, or changes in habits. The proposed approach has obtained favorable results both in the municipal application and by individual consumers. The SVR model was also employed in our work. However, the preliminary results for our data set did not indicate any benefit in dividing water consumption signals into specific periods.

In [5], the authors attempted to change the operating conditions of water supply systems using empirical exponents. The focus was mainly on modifying the flow rate and hydrostatic pressure. Many failures were observed in managing the water supply network's stress. The study area covered the Municipal Water Supply and Sewerage Company S.A. in Wrocław, and the recording time was between April 2017 and January 2018. The conducted research enabled the determination of an exponent linking the flow rate-pressure relationship to the overall assessment of the water supply network. Changes in the form of a steady, cascading decrease in pressure through the night flow period and a cascading increase in the morning resulted in the implementation of new analytical procedures for pressure and flow rate analysis. This work has a similar subject to this one. However, it is based on theoretical models and expert knowledge. This paper tackles the problem from a data science point of view and aims to define approaches that may be useful outside this particular problem involving water consumption forecasting.

In work [6], an analysis of water consumption variability based on combined long-term and short-term data was performed. The aim of the work was based on the development of a new deterministic method that would allow the data from periodic measurements to be collected together. To then serially determine the degree of variation in water consumption. The object of study was three blocks of flats located in Świdnica, and the observation time was five years. The experiment results show that it is impossible to analyze the cumulative fluctuations of water consumption for data without preprocessing. To categorize the measurements, it is necessary to supplement the data sets with water consumption records for several sufficiently long periods. Research conducted in our paper also deemed preprocessing necessary as water consumption signal tends to have random spikes of activity, which disrupts mean squared error-based loss function driven neural networks. To perform the experiments, we had to draw the same conclusions.

On the other hand, in the paper [7], seasonal forecasts were analyzed, and the main interest was developing a drought forecast. Drought in 1998-2002, which occurred in South Carolina areas, is considered the most severe problem for the state. The analysis performed was extended to include other drought indicators,

and the information to calculate these indicators is provided every month in terms of precipitation and temperature. Thus, different scenarios can be adapted to determine the amount of rain needed to end drought conditions and the probability of receiving those amounts. The article bases its methodology on bootstrapping, which we also used in our Decision Tree-based algorithms and can confirm its efficiency in forecasting.

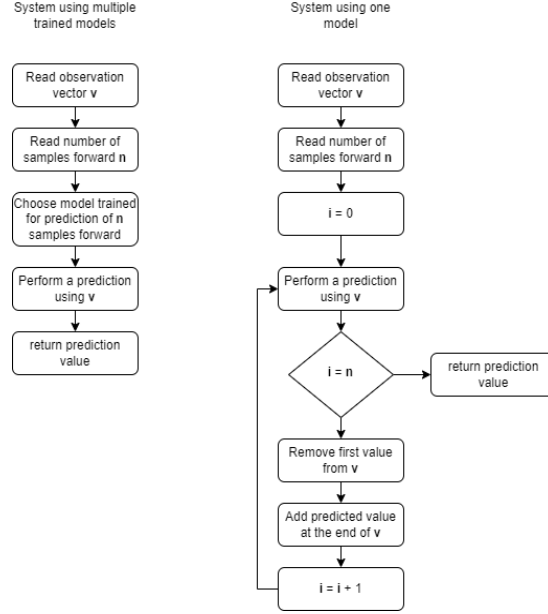
### 3 Methodology

#### 3.1 Dataset

The dataset used in this research was obtained from the measurements of the pump room, generating an average water supply of 8 m<sup>3</sup> per hour. The data is the property of Softblue S.A company. Access to the data was provided to conduct this research. The signal was initially encoded as a timestamp between subsequent transfers of 10 liters of water through the pump house. For this research, the signal was resampled to contain the amount of water transferred every 15 minutes. Due to temporal anomalies in water supply provision, data cleaning was required. Each value of material water supply exceeded the 94 percentile of water flow distribution and was reduced to the value associated with the 94 percentile.

#### 3.2 Experiment design

The cleaned data are formed into observation vectors. Each observation vector contains the last 20 values representing the previous 5 hours of the observed water consumption. Two forecasting systems use these data. The first system uses models explicitly trained to predict a fixed amount of future samples. For example, the system that can predict the next 1, 2, 3 ... 6 samples in the future contains six models, each trained to forecast its respective amounts of samples in the future. The second system has only one model trained to predict the following sample. The system uses this model to predict an arbitrarily chosen number of samples in the future by iteratively predicting the following sample and uses the value of the prediction to update the observation vector for another sample prediction. The systems' algorithms are presented in figure 1.



**Fig 1.** Presentation of the algorithms forming prediction systems evaluated in this work. On the left, the system using dedicated models is presented. On the right, the system employing an iterative approach is described.

### 3.3 Machine Learning algorithms

Each system employs machine learning models to forecast the following sample. The algorithms selected for training prediction models in this study are: neural networks with two fully connected layers (referred further in the text as FC2), recurrent neural networks [8] with two recurrent layers and one fully connected (referred further in the text as RNN), Random Forest [9], XGBoost [10], Decision Tree [11] and Support Vector Regression [12] (SVR).

These algorithms were chosen due to their different approach to the forecasting problem. FC2 and SVR models perform the linear transformation of the input vector to obtain the forecast. RNN attempts to extract temporal patterns. The Decision Tree is the algorithm that divides solution space into subsets based on the partition purity. Random Forest and XGBoost perform an ensemble of decision trees in different ways. Such a collection of algorithms presents the forecasting problem from different angles and allows comparison of varying decision-making philosophies.

The critical factor to be considered during model selection is its ability to be implemented into the hardware commonly used in IoT applications. The machine learning algorithms (Random Forest, XGBoost, Decision Tree, and SVR) are implemented in the scikit-learn library widely used in Data Science. This library is lightweight enough to be used on the higher end of IoT devices like Raspberry microcomputers. Although frameworks for Deep Neural Networks may be too computationally expensive for IoT devices, the Fully Connected Neural Networks are

algorithms simple enough to be implemented with reduced NumPy computational stack for micro-Python booted on the microcontrollers like ESP.

## 4 Experimental results

The algorithms and techniques were evaluated using the Mean Absolute Percentage Error (MAPE) metric given by equation 1. The purpose of this metric is to present relative differences between the observed values predicted by the systems.

$$MAPE = \sum_{i=0}^{|Y|} \frac{|\hat{Y}_i - Y_i|}{Y_i} \quad (1)$$

Where  $Y$  denotes the vector of observed values and  $\hat{Y}$  is the vector of predicted values.

The models were evaluated by averaging MAPE results from the 30 iterations of the experiment conducted for every triplet of the model algorithm, version of the system, and several samples to look into in the future. During each iteration, train, validation, and test datasets were randomly chosen from the available dataset.

The results are presented in tables 1 and 2. The values were obtained from evaluations conducted on test datasets. In each table, rows contain values for tested models, and columns present models' performance on the number of samples of the future forecast. Table 1 contains values from the models dedicated to predicting a specific number of samples forward. Table 2 presents the evaluation results of the system using an iterative approach.

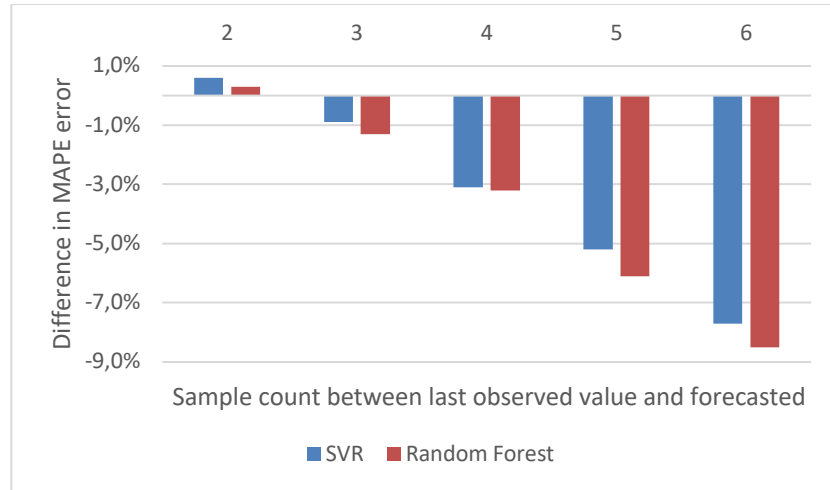
**Table 1.** MAPE metric values of the system using models dedicated to forecasting the fixed number of samples ahead. The columns denote the number of samples the model looks ahead.

Model	Sample count between last observed value and forecasted					
	1	2	3	4	5	6
SVR	9.4%	11.1%	12.1%	13.2%	14.6%	15.9%
RNN	9.4%	11.2%	12.2%	13.3%	14.4%	15.4%
Random Forest	9.5%	11.0%	11.8%	12.8%	13.8%	15.0%
FC 2	9.9%	12.1%	13.7%	15.4%	17.1%	19.5%
XGBoost	10.0%	11.6%	12.4%	13.4%	14.5%	15.5%
Decision Tree	13.7%	15.9%	16.9%	18.1%	19.3%	20.7%

**Table 2.** MAPE metric values of the system using one trained model iteratively. The columns denote the number of samples the model looks ahead.

Model	Sample count between last observed value and forecasted					
	1	2	3	4	5	6
SVR	9.4%	10.4%	13.0%	16.3%	19.9%	23.7%
RNN	9.4%	11.0%	13.7%	17.3%	21.6%	25.8%
Random Forest	9.5%	10.6%	13.2%	16.1%	19.9%	23.6%
FC 2	9.9%	11.5%	14.3%	17.9%	21.8%	26.0%
XGBoost	10.0%	11.2%	13.7%	16.7%	20.2%	24.2%
Decision Tree	13.7%	15.8%	18.4%	21.5%	24.8%	29.1%

According to the data stored in tables 1 and 2, two algorithms with the best MAPE score on both the short and long-range forecasts are SVR and Random Forest. Figure 2 depicts the differences between the system using the dedicated model for forecasting and its iterative counterpart for both algorithms.



**Fig 2.** Difference between MAPE error scores between systems using dedicated models for forecasting and their iterative counterparts.

## 5 Conclusion

Surprisingly, the iterative system proved slightly better than the system using dedicated models across all modeling algorithms. The improvement of the score ranges from 0.1% to 0.7%. Although the difference may not be considered significant,

the repeating tendency on all tested algorithms may indicate that the water consumption signal measured in 15 minutes intervals may have a short-range trend in addition to its evident seasonality. In such a case, an iterative approach may stabilize the prediction, improving its average performance, albeit not for much.

Iterative systems' performance quickly deteriorates from fourth sample forward forecasting compared to the model with dedicated models. Assessment of further prediction is pointless due to error difference reaching almost nine percentage points. However, the iterative approach may be helpful for a relatively reliable long-range forecast to the fourth sample forward, for example, as a basis for anomaly detection systems. For such cases, the reduced number of models is preferable to better precision due to the limited memory of low-computational devices commonly used for IoT purposes. The thesis of this work has been proven suitable for forecasting up to four samples forward.

Systems with dedicated models provide a stable increase in the error of about one percentage point per number of forecast samples. The two best algorithms for this particular task are Random Forest and SVR. Their performance was the best for both short and long-range forecasts. However, SVR is an algorithm of much less computational complexity and memory dependence than Random Forest, resulting in a better solution for low-power devices like microcomputers.

Despite its more straightforward structure, the SVR model proved to have significantly better results than Neural Network with two fully connected layers (FC2). Overfitting is unlikely because a Recurrent Neural Network with two RNN layers obtained better results than FC2 despite a more complex structure. The margin function employed in SVR may be responsible for the improvement of the model. The margin function reduces the impact of the outliers by associating the constant penalty for values outside specified prediction bounds instead of computing loss for each outlier proportional to the squared difference between prediction and observed value. Further investigation of this phenomenon is required.

## References

1. Jan, F., Min-Allah, N., Saeed, S., Iqbal, S. Z., & Ahmed, R. (2022). IoT-Based Solutions to Monitor Water Level, Leakage, and Motor Control for Smart Water Tanks. *Water*, 14(3), 309.
2. Benítez, R., Ortiz-Caraballo, C., Preciado, J. C., Conejero, J. M., Sánchez Figueroa, F., & Rubio-Largo, A. (2019). A short-term data based water consumption prediction approach. *Energies*, 12(12), 2359.
3. Dzimińska, P., Drzewiecki, S., Ruman, M., Kosek, K., Mikołajewski, K., & Licznar, P. (2021). The use of cluster analysis to evaluate the impact of COVID-19 pandemic on daily water demand patterns. *Sustainability*, 13(11), 5772.
4. Candelieri, A. (2017). Clustering and support vector regression for water demand forecasting and anomaly detection. *Water*, 9(3), 224.

5. Stańczyk, J., & Burszta-Adamiak, E. (2019). The Analysis of Water Supply Operating Conditions Systems by Means of Empirical Exponents. *Water*, 11(12), 2452.
6. Wawrzosek, J., Ignaciuk, S., Stańczyk, J., & Kajewska-Szkudlarek, J. (2021). Water Consumption Variability Based on Cumulative Data From Non-simultaneous and Long-term Measurements. *Water Resources Management*, 35(9), 2799-2812.
7. Carbone, G. J., & Dow, K. (2005). WATER RESOURCE MANAGEMENT AND DROUGHT FORECASTS IN SOUTH CAROLINA 1. *JAWRA Journal of the American Water Resources Association*, 41(1), 145-155.
8. Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, 404, 132306.
9. Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5-32.
10. Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).
11. Swain, P. H., & Hauska, H. (1977). The decision tree classifier: Design and potential. *IEEE Transactions on Geoscience Electronics*, 15(3), 142-147.
12. Basak, D., Pal, S., & Patranabis, D. C. (2007). Support Vector Regression Neural Information Processing—Letters and Reviews.

# Genetic Algorithm based Aggregation for Federated Learning in Industrial Cyber Physical Systems

Souhila Badra GUENDOUZI and Samir OUCHANI and Mimoun MALKI

**Abstract** During the last decade, Industrial Cyber-Physical Systems (ICPS ) have attracted a significant amount of interest from industries as well as academic institutions. These kinds of systems have proved to be very complicated, and it may be a difficult task to get a handle on their architecture and make sure everything works properly. By putting up a framework for federated learning that we’ve dubbed FedGA-ICPS the purpose of this study is to address some of the difficulties that are associated with the performance and decision-making aids provided by ICPS. To begin, we launch an ICPS modeling formalism with the goal of specifying the structure and behaviour of such systems. FedGA-ICPS then conducts an analysis of the performance of the industrial sensors based on the data supplied by the ICPS from the *industrial* sensors by putting forth locally integrated learning models. Following that, a genetic algorithm drives federated learning in order to quicken and enhance the aggregation process. In the end, transfer learning is used so that the learned parameters of the models may be distributed across a variety of limited entities. FedGA-ICPS has been implemented on MNIST, and the results have been rather significant.

## 1 Introduction

The Internet of Things (IoT) has emerged as one of the most essential technologies of the 21st century during the years. There are many different fields, such as the Internet of Industrial Things (IoIT) and the Internet of Medical Things (IoMT), which refer to the improvement of industrial processes and medical applications through the

---

Souhila Badra GUENDOUZI

ESI Engineering School, Sidi bel Abbès, Algeria, e-mail: b.guendouzi@esi-sba.dz

Samir OUCHANI

LINEACT CESI, Aix-en-Provence, France, e-mail: souchani@cesi.fr

Mimoun MALKI

LabRI-SBA Laboratory, Ecole Supérieure en Informatique, Sidi Bel Abbès, Algeria, e-mail: m.malki@esi-sba.dz



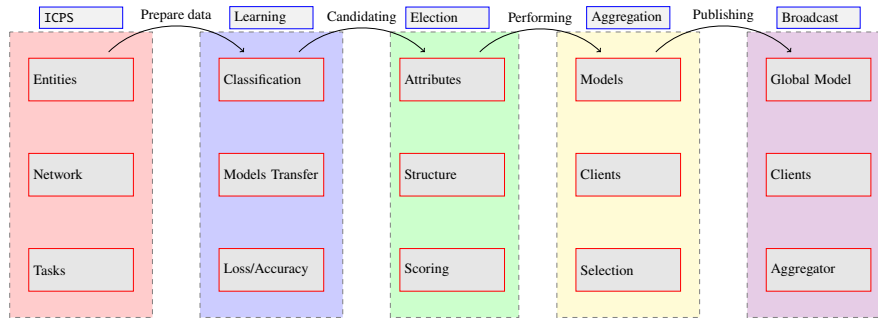
utilization of smart sensors, actuators, quick communication protocols, and effective cybersecurity mechanisms. In vast networks, intelligent devices create a significant quantity of data; hence, IoT frameworks demand methods that are both clever and resilient for analyzing massive volumes of data.

Precisely, deep learning (DL) approaches have achieved promising results in networks application owing to their intelligent learning and processing skills. These qualities have enabled these techniques to create positive outcomes. Historically, the application of AI in CPS has been done in a centralized manner. This means that there is a central server that trains the model by using all of the data from the end devices that are linked to it. The transmission of such a massive volume of data from these end devices to the cloud, however, leads to congestion on the bandwidth, delays in the processing of data, and perhaps a breach of privacy. Recently, this method of learning has been modified to be either distributed learning or federated learning, which brings together a collection of clients (end devices) that each have their own set of data that is kept private and is not shared with other clients. They coordinate their efforts with one another so that they may achieve the same degree of learning by exchanging information on the learning parameters.

Google first presented the concept of federated learning (FL), which is a decentralized and collaborative method of machine learning, in the year 2016. The goal of the FL process is to produce a model that is suitable for a certain application in its entirety. Choose the appropriate machine learning model to use and select devices candidates to participate in the learning process where each device initializes its embedded model parameters and trains the model until it converges. This is an example of a typical workflow for its Life cycle, which includes: choosing the appropriate machine learning model to use. Following a training session on the local level, each of the linked devices then uploads its model parameters to the centralized server via a secure communication method that we will go over in more detail later. Then, after the aggregator (for example, a central server) has received all of the local models, it will aggregate the parameters in order to update the new global and optimum model. Once again, the server will re-share the parameters of the global model, and the devices will update their own parameters based on the new information. This procedure is carried out in a loop until the whole of the training process has been completed.

In the literature, McMahan et al. [1] introduced a new aggregated model called FedAVG, which stands for federated average. In this model, the weights of the several local models are averaged by the server, which then provides new weights and, therefore, a new model.

Also, FedPer proposed by Arivazhagan et al. [2] similar to the FedAVG in the way it computes new weights in the aggregated models [3]. However, the clients communicate the neural model's base layers to the server instead of the totality of the model and retain the other layers. The underlying idea is that the base layers deal with representation learning and can be advantageously shared by clients through aggregation. The upper layers are more concerned with decision-making, which is more specific to each client [3]. FedPer can be seen as an adaption of the transfer learning methodology into a federated learning scheme.



**Fig. 1** The Interoperability and Integrity Validation and Evaluation.

The main goal of this work is to overcome the limitations of FedAvg and FedPer by introducing genetic algorithm to enhance the aggregation process of FL and deploy it for ICPS. As shown in Figure 1, FedGA-ICPS develops the following five steps.

- (1) FedGA-ICPS will create and put into effect a CPS in the form of a composition of entities and components of varying forms and kinds. Each entity possesses the correct behavior and structure for itself. The beings are capable of communication and can move between different environments.
- (2) Through simulation and run-time execution, FedGA-ICPS collects, formats, cleans and normalizes streaming data 'D' generated and communicated between different CPS components. 'D' will be used for the learning step that relies on convolution neural network that is assigned to a given device.
- (3) FedGA-ICPS provides a series of components in order to select the most qualified candidate to federate the learning that occurs between embedded CNNs. During the voting process, a variety of criteria, including processing and memory capacity, latency, availability, and security, among others, are taken into consideration.
- (4) FedGA-ICPS utilizes genetic algorithms in order to carry out the aggregation. This latter method takes into consideration: the weights of the various local models. Then, in the component that had been elected, the optimum weight vector for broadcasting was constructed.
- (5) The optimal weights that were determined by FedGA-ICPS after using transfer parameter learning are communicated to the various customers, edges, and components.

In a nutshell, we present our main contributions in the present paper.

- Surveying the main contributions related to the application of FL on CPS.
- Develop a framework called FedGA-ICPS to enhance the performance to ICPS.
- Model formally ICPS components and their compositions.
- Propose a federated solution to enhance the collaborative learning phases between ICPS components through genetic algorithms.
- Compare FedGA-ICPS within the existing solutions and validates it on benchmarks.

The remainder of this paper is organized as follows. Section 2 surveys contributions related to federated learning and its applications. Section 3 develops our

proposed framework, FedGA-ICPS that is validated in 4. Finally, Section 5 concludes the paper and give hints on our FedGA-ICPS related perspectives.

## 2 Related Work

This section reviews and discusses approaches that deal with performance analysis and decision supports for ICPS. Połap et al. [4] proposed an agent-based system to analyze medical data collected from IoMT stored in a blockchain. The solution implements three agents: learning, indirect, and data management (DM). Unfortunately, this solution is not real-time since DMA takes time to get information from patients and doctors. Also, despite the use of FL methods, selecting the best classification method for this kind of noisy and heterogeneous data is challenging.

Tian et al. [5] used an asynchronous FL-based anomaly detection for resources constrained IoT devices based on deep learning techniques. Their approach is established through: pre-initialization of the parameters, and deploy an asynchronous model training. Unfortunately, they did not study the reliability of the participating nodes. Chen et al. [6] proposed FedHealth framework that applies federated transfer learning approach on cloud computing architecture for wearable healthcare. Their framework deals with the issues of data isolation and model personalising in FL. Initially, the cloud server develops the global model using public datasets, distributes it to the clients using homomorphical encryption, then the clients retrain local models and upload them to the cloud server. Monotonously, the aggregator builds the global model using fedAVG algorithm, distributes it to the clients and perform transfer learning. Hao et al. [7] They took the initiative of Privacy-enhanced FL (PEFL), which aims to improve the accuracy of local models while simultaneously boosting the safety of model gradients that are transmitted between the server and clients. Their system architecture includes the following components: (1) a Key Generation Center (KGC) that is responsible for distributing private keys to each participant; (2) a Cloud Service that acts as an Aggregator; and (3) participants. In the first step of the process, each member will acquire their own local model, estimate their own local gradients, and then disrupt them by utilizing the Differential Privacy (DP) technique to introduce local disturbances. Homomorphic encryption is used to cipher the disturbed gradients into the BGV ciphertext, and the ciphertext BGV encryption is then integrated into enhanced learning. After that, the CS will carry out the opposite procedure of encryption by carrying out a series of decryption steps in preparation for aggregation. The results indicate that there is only a modest decline in accuracy when just a few of the participants are impacted by the challenge.

Zhou et al. [8] developed a federated learning approach for fog computing that would protect users' privacy. Each fog node has the capability to act as a participant and collect data from IoT devices in order to accomplish the learning objective. This design effectively improves the low training effectiveness and model accuracy that are generated by the unequal distribution of data and the large difference in computer resources. They enabled the data from IoT devices to satisfy differential privacy in order to thwart data attacks and leveraged the combination of blinding and Paillier

homomorphic encryption in order to defend against model attacks, which resulted in the realization of the security aggregation of model parameters.

### 3 FedGA-ICPS Framework

This section follows FedGA-ICPS steps represented in Fig. 1 by introducing the ICPS formalism.

#### 3.1 Industrial CPS

We think of a system  $S$  as being composed of a collection of entities  $\mathcal{E}$  that collaborate and intertwine with one another through a network ( $\mathcal{N}$ ) of physical and logical channels in order to carry out a predetermined job ( $\mathcal{T}$ ) inside a particular context  $\mathcal{CT}$ . A system  $S$  is a tuple  $\langle \mathcal{E}, \mathcal{N}, \mathcal{T}, \mathcal{CT} \rangle$ .

##### 3.1.1 Modeling the entities

An entity  $\varepsilon \in \mathcal{E}$  may be an Internet of Things (IIoT), an edge node, a fog node, or a cloud server. These are all examples of entities that are able to carry out certain activities on their own or work together with other entities to build a system that carries out a global job. To evaluate the guard related to an action, the entity  $\varepsilon$  run its associated machine learning  $\text{ml}_\varepsilon$  that evaluates the variables of the specified guard. In order to improve the quality of decisions made by an entity, the FedGA-ICPS provides strategies to assist entities in regularly updating the related  $\text{ml}$  with which they are associated. However,  $\varepsilon$  is the main entities describing ICPS, and it is defined by the tuple  $\langle id, attr, Actuator, \Sigma, Beh \rangle$ , where:

- $id$  is a finite set of tags,
- $attr : id \rightarrow 2^T$  returns the attributes of an entity,
- $Actuator$  specifies the status of an entity by evaluating its attributes,
- is a function that associate to an entity  $\varepsilon$  its  $\text{ml}$ .
- $\Sigma = \{\text{Send}, \text{Receive}, \text{Update}, \text{Predict}, \text{Train}, \text{Aggregate}\}$  is a finite set of atomic actions that depend on the type of entity  $\varepsilon_i$  and executed by the latter.
- $Beh : id \times \Sigma \rightarrow \mathcal{L}$  returns the expression written in the language  $\mathcal{L}$  that describes the behavior of an entity. The syntax of  $\mathcal{L}$  is given by:  $B ::= \alpha \mid B \cdot B \mid B +_g B$ , where  $\alpha \in \Sigma$ , “ $\cdot$ ” composes sequentially the actions and  $+_g$  is a guarded choice decision that depends on the evaluation of the guard, a propositional formula,  $g$ , by the functionality  $\text{Predict}$ . When  $g \stackrel{\Delta}{=} \top$  the guarded decision become a non-deterministic choice.

### 3.1.2 Modeling the Network

The  $\mathcal{N}$  describes how the entities are linked to one another and how they interact with one another. It is possible for one entity, denoted by  $\varepsilon_i$ , to be linked to another entity by means of a logical or physical communication channel, or even to a subsystem. The term  $\mathcal{N}$  refers to a graph in which the vertices represent the entities and the edges refer to the ways in which the entities interact with one another and are linked  $\mathcal{N} = \langle \mathcal{E}, \text{Chan}, \text{Prot}, \text{Rel} \rangle$ , where:

- $\text{Chan}$  is a finite set of channels,
- $\text{Prot}$  is a finite set of protocols where  $\epsilon_{\text{prot}}$  is the empty protocol.
- $\text{Rel} : \mathcal{E} \times \mathcal{E} \rightarrow \text{Chan} \times \text{Prot}$  relies two entities with a channel and a protocol. When  $\epsilon_{\text{prot}}$  is assigned, it means both nodes are physically connected.

### 3.1.3 Modeling the Tasks

The system's primary purpose is to complete the task  $\mathcal{T}$ , which is denoted by tasks. It details the order of activities that each entity ought to carry out in order to achieve the desired results. A task is represented as a tree in our definition, with the root standing for the overarching objective of the system  $\mathcal{S}$ , the offspring standing for subsidiary objectives of the entities, and the leaves standing for the completed output for each entity. The task  $\mathcal{T}$  is the tuple  $\langle \text{Goals}, \leq \rangle$ , where:

- $\text{Goals}$  is a finite set of goals where  $G \in \text{Goals}$  is the root (the main goal),
- $(\text{Goals}, \leq)$  is a preorder relation on  $\text{Goals}$ .

### 3.1.4 Context

It is possible to see it as a container for entities that are capable of undergoing dynamic change via the incorporation or exclusion of entities, the modification of protocols, and the updating of tasks; nonetheless, these entities should adhere to a set of predetermined rules and policies  $\mathcal{P}\mathcal{L}$ . A context  $C\mathcal{T}$  is the tuple  $\langle \mathcal{E}' \subseteq \mathcal{E}, \mathcal{T}' \subseteq \mathcal{T}, \mathcal{P}\mathcal{L} \rangle$ .

## 3.2 Learning

### 3.2.1 Classification

CNN, short for convolutional neural network, is a kind of deep neural network [9] that is used in the learning phase of the FedGA-ICPSThe structure of CNN is made up of three layers: the first is called convolution, which extracts features from the data, the second is called pooling, which reduces the number of dimensions that feature maps have, and the third is called fully connected, which classifies the data into several categories according to the attributes that were gathered by the layers that came before it [10]. In the convolutional and pooling layers of a neural network,

ReLU functions are often employed to classify the inputs. On the other hand, in the FC layers, a softmax activation function is typically utilized to give a probability ranging from 0 to 1. Because it is reliant on the structure of its own dataset, each edge client in our system has its own CNN model. This model does not have to be similar to other local models in terms of the number of filters, layer design, function activation, and so on.

### 3.2.2 Models Transfer

Transfer learning, also known as TL, is the practice of reusing a model that has already been trained on another learning task that is related to the first. Since a deep neural network requires a large amount of data and resources with computing power to accelerate learning in order to make good decisions, transfer learning is known as a solution that can help deal with these challenges. In this work, we concentrate on the process of fine-tuning pre-trained models. When a new edge node is added to the industrial network, the cloud server searches for pre-trained models that are comparable to those already deployed on other nodes of the same kind. These models must have been trained using a large dataset and must have reached a high level of accuracy. As a consequence of this, only the base layers of the network need to be retrained as opposed to the whole network, since the feature extractor layers are the only ones that are sent to the target node.

## 3.3 Election

For federated learning, FedGA-ICPS seeks for the most powerful Fog or Edge node. As default, we consider the cloud server as the aggregator. Depending on a component's free memory and processing capabilities, the cloud server may elect it as a secondary aggregator. FedGA-ICPS reorders all system components into a prioritized list.

## 3.4 Aggregation

Unlike FedAVG [11] and FedPer [2], FedGA simply uploads encrypted base layer (classification layer) weights to the designated aggregator. The latter creates new weights by invoking the genetic algorithm (line 9) and using a weight vector to describe the system across chromosomes.

- (1) Define an adequate *chromosome* which the weight vectors.
- (2) Select a large set of chromosomes, *population* takes into account all weight vectors collected from the different components.
- (3) Apply the reproduction operators (*selection*, *crossover*, *mutation*). *selection* is applied on on vectors with high ranking (fitness evaluation). In our case, the fitness is the loss function. The *crossover* operation is based on the single point

---

**Algorithm 1** Federated Genetic Algorithm (FedGA),  $C$  is the fraction of clients selected randomly to participate in each communication round. The  $\mathbf{K}$  clients are indexed by  $\mathbf{k}$ ;  $\mathbf{B}$  is the local mini-batch size,  $\mathcal{D}_k$  is the dataset available to client  $\mathbf{k}$ ,  $\mathcal{D}_t$  is the dataset used for the test which is available on the aggregator,  $W_{\mathbf{B}}$  is the vector of base layers (classification layers),  $\mathbf{E}$  is the number of local epochs, and  $\eta$  is the learning rate.

---

```

1: Procedure FedGA                                     ▶ Run on the server.
2: Initialize  $W_{\mathbf{B}}^0$ ;
3: for each round  $\mathbf{t} = 1, 2, 3, \dots$  do
4:    $m \leftarrow \max(C.K, 1)$ ;
5:    $S_t \leftarrow$  (random set of  $m$  clients);
6:   for each client  $\mathbf{k} \in S_t$  do
7:      $W_{\mathbf{B},\mathbf{k}}^{t+1} \leftarrow \text{ClientUpdate}(\mathbf{k}, W_{\mathbf{B},\mathbf{k}}^t)$ ;           ▶ In Parallel.
8:   end for
9:    $W_{\mathbf{B}}^{t+1} = \text{GA}(\mathcal{D}_t, W_{\mathbf{B}}^t)$ ;                               ▶ Only base layers are aggregated
10: end for
11: End procedure FedGA
12: Procedure ClientUpdate( $k, w_{\mathbf{B}}^t$ )                       ▶ Run on client  $\mathbf{k}$ .
13:  $\beta \leftarrow$  (Split  $\mathcal{D}_k$  into mini-batches of size  $\mathbf{B}$ );
14: for each local epoch  $i$  from 1 to  $\mathbf{E}$  do
15:   for batch  $\mathbf{b} \in \beta$  do
16:      $w_{\mathbf{B}}, w_{\mathbf{P}} \leftarrow w - \eta \Delta \mathcal{L}(w_{\mathbf{B}}, w_{\mathbf{P}}, b)$ ;       ▶ Only base layers are aggregated
17:   end for
18: end for
19: return  $\mathbf{t}$  to the Server
20: End procedure ClientUpdate( $k, w_{\mathbf{B}}$ )

```

---

paradigm. It means that a vector is divided into two parts to be exchanged with another vector to form a new population. Finally, the *mutation* operation collects randomly only 10% of weights to reproduce new vectors.

- (4) FedGA-ICPS repeats this process until the accuracy of all edge nodes models is more than 99%.

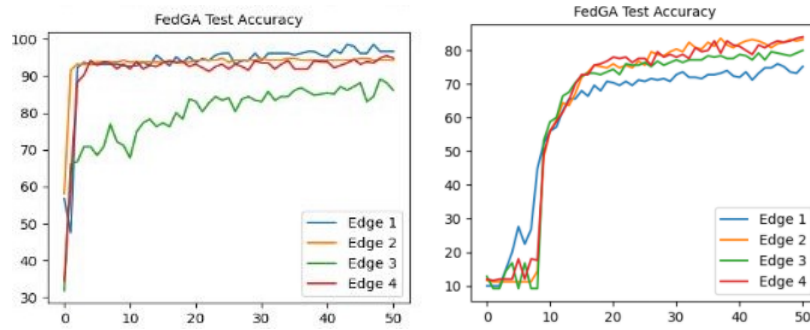
### 3.5 Broadcasting

After aggregation, FedGA-ICPS transmits model base layer weights to learning phase components. In the training phase, a high-accuracy, low-loss model is created. Homomorphic encryption is used for broadcasting.

## 4 Experimental results

We evaluated our FedGA-ICPS structure using MNIST [12]. We subdivide it. Each customer has an unequal-sized and distribution (non-iid data). We used a 1000-sample window. Our experiments were done using CNN to compare the federated

learning combined with genetic algorithm results against FedAVG and FedPer. Our CNN model contains two convolutional layers followed by a max-pooling layer. Mini-batch SGD of size 1000, ReLU activation function, and dropout are utilized to train the models. They are trained using a mini-batch SGD of size 1000, ReLU activation function and to counter over-fitting, a dropout is used. For implementation, we used Pytorch. We fixed FL round to 50. The results show that is a perturbation of the accuracy for each customer due to data heterogeneity and dataset size variance using FedAVG. The obtained results using FedPer present that there is an improvement in accuracy compared to FedAVG by stabilizing the convergence of the models. Finally, using FedGA-ICPS, we obtain a rapid convergence for all clients with an average accuracy greater than 95% of all edge nodes, except of edge node  $\varepsilon_4$  that has irrelevant dataset and weak model as shown in Figure 2 (left). The aggregation results are obtained within rounds in 300 sec, the transfer step has been proceeded after five training phases where each model of a given edge takes 23 sec. Figure 2 (right) shows the results when data are heterogeneous.



**Fig. 2** FedGA Aggregation Algorithm Results.

## 5 Conclusion

In this work, we have provided a first step toward a comprehensive methodology for strengthening performance testing in order to construct a more resilient ICPS. This step was accomplished by giving an overview of the approach. A system is defined as a set of items, each of which has its own structure and behavior for the goal of fulfilling out a given function, according to the architecture that was developed by FedGA-ICPS. In order to speed up the processes of analysis and learning, FedGA-ICPS plans to leverage federated learning and genetic algorithms to assist restricted devices in locally embedding their decision models. This will allow the procedures to be completed more quickly. By utilizing a well-known standard, we were able to provide evidence that the proposed framework is effective. In the future, we intend to grow the functionality of the framework by (1) incorporating additional machine learning techniques and automatically selecting the best agent, primarily through reinforcement learning; (2) decentralizing the system through the implementation



of a blockchain architecture; and (3) evaluating the framework on use cases and benchmarks that are more complex.

## References

- [1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.
- [2] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019.
- [3] Sannara Ek, François Portet, Philippe Lalanda, and German Vega. A federated learning aggregation algorithm for pervasive computing: Evaluation and comparison. In *19th IEEE I. C. on Perv. Comp. and Comm. PerCom*, 2021.
- [4] Dawid Połap, Gautam Srivastava, and Keping Yu. Agent architecture of an intelligent medical system based on federated learning and blockchain technology. *Journal of Information Security and Applications*, 58:102748, 2021.
- [5] Pu Tian, Zheyi Chen, Wei Yu, and Weixian Liao. Towards asynchronous federated learning based threat detection: A dc-adam approach. *Computers & Security*, 108:102344, 2021.
- [6] Yiqiang Chen, Xin Qin, Jindong Wang, Chaohui Yu, and Wen Gao. Fed-health: A federated transfer learning framework for wearable healthcare. *IEEE Intelligent Systems*, 35(4):83–93, 2020.
- [7] Meng Hao, Hongwei Li, Xizhao Luo, Guowen Xu, Haomiao Yang, and Sen Liu. Efficient and privacy-enhanced federated learning for industrial artificial intelligence. *IEEE Trans. on Ind. Informatics*, 16(10):6532–6542, 2019.
- [8] Chunyi Zhou, Anmin Fu, Shui Yu, Wei Yang, Huaqun Wang, and Yuqing Zhang. Privacy-preserving federated learning in fog computing. *IEEE Internet of Things Journal*, 7(11):10782–10793, 2020.
- [9] Chapter 4 - multi-category classification problem. In Shih-Chia Huang and Trung-Hieu Le, editors, *Principles and Labs for Deep Learning*, pages 81–116. Academic Press, 2021.
- [10] Wenbo Zhu, Yan Ma, Yizhong Zhou, Michael Benton, and Jose Romagnoli. Deep learning based soft sensor and its application on a pyrolysis reactor for compositions predictions of gas phase components. In *13th International Symposium on Process Systems Engineering (PSE 2018)*, volume 44, pages 2245–2250. Elsevier, 2018.
- [11] Tao Sun, Dongsheng Li, and Bao Wang. Decentralized federated averaging. *arXiv preprint arXiv:2104.11375*, 2021.
- [12] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

# Hand SOS gesture detection by computer vision

Roberto Viejo-López, Virginia Riego del Castillo, and Lidia Sánchez-González

**Abstract** Computer vision has been applied widely in cybersecurity, specially for authentication purposes like iris or fingerprint recognition. Image processing techniques also allow to understand hand gestures of the sign language alphabet, among others. Combining both approaches, in this paper, a system to detect the hand SOS gesture is proposed. By training a model to understand hand gestures, the detection of a certain sequence of hand gestures make possible to identify the SOS signal. The proposed method can be deployed in surveillance systems and others devices with a camera such as social robots. So, victims can ask for help silently and alarms can inform the authorities automatically.

## 1 Introduction

Computer vision has been employed in many fields in order to automatize procedures. In cybersecurity, there are many approaches that use image processing in authentication systems by recognizing iris [10, 23], face [7] or fingerprints [22].

Additional biometric features have been also studied for authentication such as the way of walking, signature, retina, ear or voice, among others [26, 3, 14, 9]. There are more innovative solutions in which the element to be analysed is represented as an image and by processing such image an anomaly can be detected. This can

---

Roberto Viejo-López

Departamento de Ingenierías Mecánica, Informática y Aeroespacial, Universidad de León, 24071, León, Spain, e-mail: rviejl00@estudiantes.unileon.es

Virginia Riego del Castillo

Departamento de Ingenierías Mecánica, Informática y Aeroespacial, Universidad de León, 24071, León, Spain, e-mail: vriegc@unileon.es

Lidia Sánchez-González

Departamento de Ingenierías Mecánica, Informática y Aeroespacial, Universidad de León, 24071, León, Spain, e-mail: lidia.sanchez@unileon.es

be used in traffic anomaly detection [1], malware detection [18] or for file analysis, among others [27].

More applications of cybersecurity that employ computer vision methods are the use of unsupervised learning for the detection of malware by analysing its visual representation [5]. Another example applies image descriptors as the wavelet Hashing or the Scale-Invariant Feature Transform (SIFT) to determine image similarity in order to detect phishing websites [8].

Gesture or pose has been also considered for cybersecurity authentication [13]. Methods to detect hand gestures involve traditional techniques as morphological operations [4] or by combining features extracted from the histogram of oriented gradients (HOG) with local binary pattern (LBP) ones, as in [17]. More recently, the use of Convolutional Neural Networks has also been applied with a high accuracy over 90% [19, 2, 12]. A complete review of the existing methods for hand gesture recognition is provided in [21], classifying them mainly in two groups, one to interact with a system without using a device and another for sign language recognition. Recently, a new dataset has been published (HAnd Gesture Recognition Image Dataset, HAGRID) with more than 550,000 samples of hand gestures but it only has 18 classes [15]. Furthermore, different datasets are cited in [11] and they consider two new hand gestures, but the dataset they proposed is focused on human-computer interfaces (HCI) to control multimedia systems by a sequence of gestures. Regarding applications that recognize sign gestures are mainly focused on hearing-impaired people, using i.e. CNN [16], although it is a challenging task since sign language is not universal and each region has its own set of signs.

A certain sequence of hand gestures is internationally known as a way to alert that someone needs help. This gesture can be reproduced silently in order to make it visible to anyone, including a video camera system like a surveillance camera. However, the authors are not aware that a system exists that, using computer vision techniques, can detect the SOS signal automatically.

In this paper, a method to detect the SOS hand gesture by analysing real time video is proposed. This application can be run on a camera system including the ones included in social robots in order to identify if anyone is asking for help.

The structure of the paper is as follows. In section 2, the problem is described. Section 3 explains the architecture of the proposed model. In Section 4, the obtained results are discussed. Finally, Section 5 presents the achieved conclusions.

## 2 Problem description

The Women's Funding Network and the Canadian Women's Foundation developed a signal to ask for SOS that can be showed at any situation, for instance, answering the door, walking on the street or during a video call. This signal has been adopted internationally as it allows a victim to ask for help silently. This is formed by a sequence of hand gestures as it is explained in [6]. So, first the hand is open and the thumb is tucked into the palm (Fig. 1 (left)). Next, the fist is closed (Fig. 1 (right)).

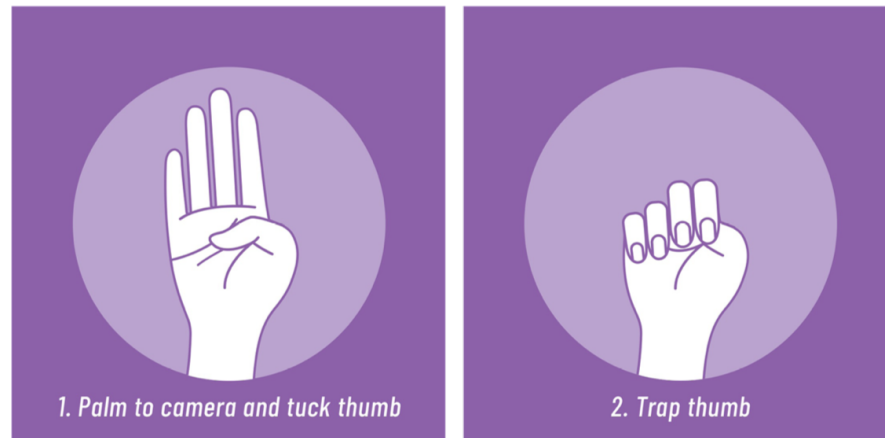
Unfortunately, it is not uncommon to read in the news that someone has made this signal in order to ask for help due to a situation of kidnapping or abuse, among others. Nowadays there are many cameras available on the streets or other IoT devices or surveillance systems with cameras; social robots also collaborate with humans, i.e., in care or performing functions as hotel receptionist [24]. Any such device with a camera could be a silent watchdog on which this method could be deployed.

For that reason, a system to identify this hand SOS gesture is presented. By using existing approaches of hand gesture recognition, the signal sequence is detected and alarms can automatically triggered.

### 3 Model architecture

The proposed method is based on the use of MediaPipe hand tracking [20] that is a framework based on OpenCV and Tensorflow, among others. Basically, it recognizes 21 keypoints of the hand or hands by analysing every frame of a video in real time. The keypoints or landmarks are showed in Fig. 2.

By following the approach implemented in [25], from the keypoints detected by MediaPipe library [20], a MultiLayer Perceptron (MLP) is used to classify the data. So, the model is formed by the layers Fig. 3 shows. The considered classes are 7, in order to detect the SOS signal but also other similar signals to try to distinguish between them: open hand, close hand, pointing with a finger, OK signal, Scissors signal, first gesture of SOS signal (as it is shown in Fig. 1 left) and second gesture of SOS signal (see Fig. 1 right). The last two classes have been added to the dataset in order to adapt it to the hand SOS gesture detection. By analysing the output of the model, if the sequence of SOS gestures is detected, the alarm is triggered. This



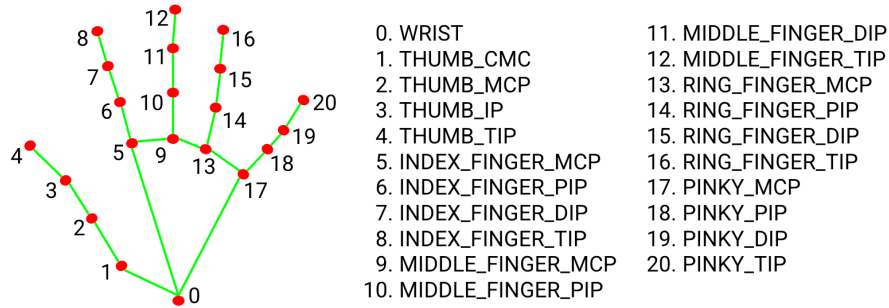
**Fig. 1** Hand SOS gesture [6]

model processes images with a frame rate of 30 images per second, so real-time analysis can be carried out.

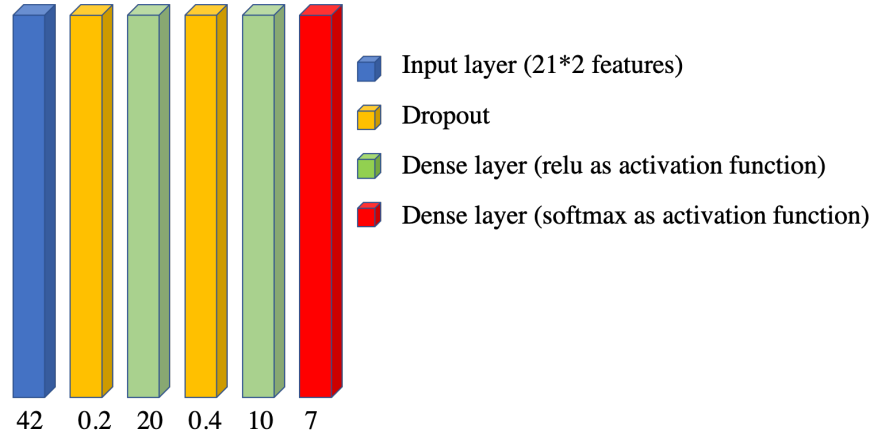
The output yielded by the model is analysed in order to detect the sequence of hand gestures that represent the hand SOS gesture.

#### 4 Model evaluation

In order to evaluate the method, the dataset used to train the model for hand gesture detection available in [25] has been increased with the hand SOS gesture. Using the hand gesture application, 433 images have been acquired (213 samples of the first hand gesture and 220 sample of the second hand gesture of the SOS sign). For those images, the keypoints are computed so as to form the feature vector of 21 elements. These images are in different orientations and both hands -right and left- are sampled.



**Fig. 2** Landmarks used for hand gesture recognition [20]



**Fig. 3** Architecture of the model

Dataset is split: for training is used the 70% and the remaining 30% for test. The used optimizer is Adam, the function loss is sparse categorical cross-entropy and the considered metric to evaluate the model is its accuracy. Training has been carried with 1000 epochs and a batch size of 128.

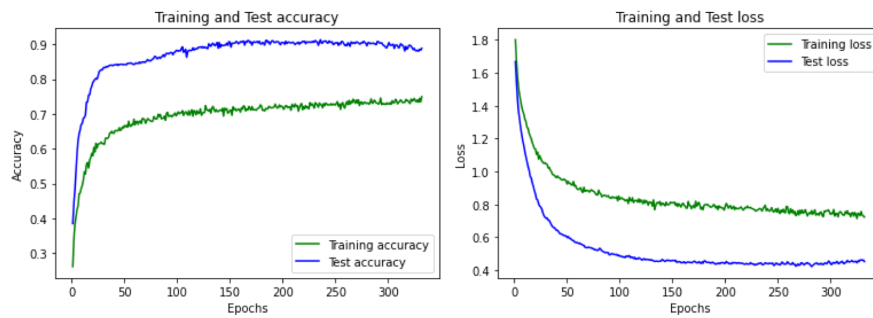
Fig.4 shows the accuracy and the loss for training and test. As it can be observed, hand gestures are quite well recognized with an accuracy around 0.90 in test images. All these results have been obtained in real-time, processing 30 frames per second.

Experiments with real users and different backgrounds have been carried out. So, they were presented in front of the camera and the application show in real time if the signals have been detected, showing the keypoints in the image as well as the name of the gesture as Fig. 5 shows.

To evaluate the recognition of the hand SOS gesture, 600 samples formed by the sequence of the two gestures that compose the sign have been acquired considering three different persons. Each person repeats 100 times the two signs that make up the SOS gesture, 50 times with the right hand and 50 times with the left hand, varying the orientation of the hand, since the developed application is capable of recognizing the gesture sequence from both sides of the hand. Table 1 gathers the obtained results. As can be seen, when the gesture is performed with the right hand the results are worse than with the left hand because most of the images that compose the dataset are recorded with the left hand. In general, the second sign is better identified than the first one, which tends to be misclassified as 'open hand'. In overall, the SOS gesture is correctly recognized in 75.67% of the cases, achieving a 92.67% if the gesture is performed by the left hand.

## 5 Conclusions

This paper proposes a vision-based system in order to detect the hand SOS gesture. First, 21 landmarks on a hand are detected with the MediaPipe library. This feature vector is classified with a MLP that determines the corresponding hand gesture.

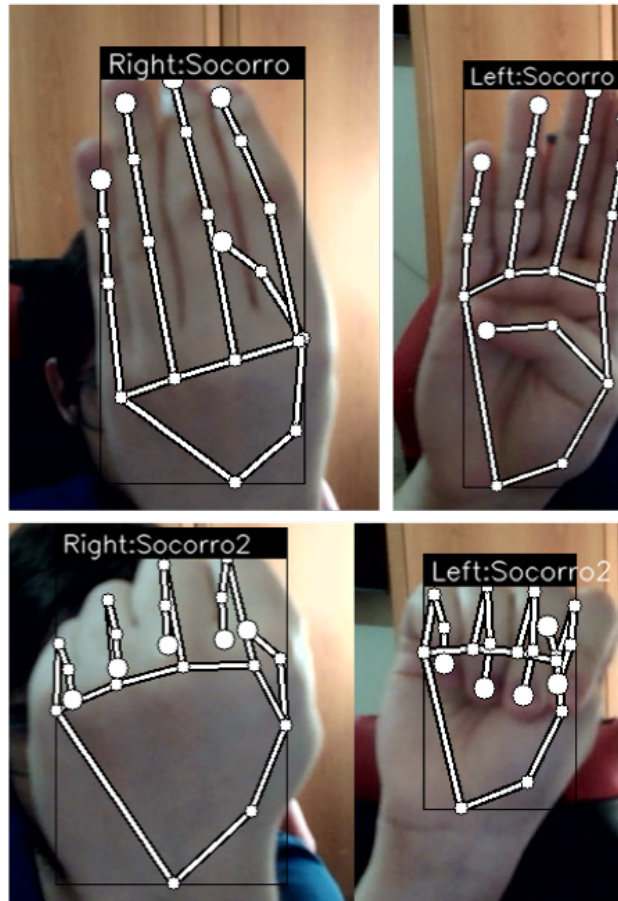


**Fig. 4** Accuracy and loss of training and test

**Table 1** Results of the evaluation with 3 different persons and 50 hand SOS gestures with each hand expressing the results as hit rates.

Hand	User	First gesture (%)	Second gesture (%)	SOS gesture (%)	Mean (%)
Right	1	66%	80%	66%	58.67%
	2	74%	68%	68%	
	3	42%	54%	42%	
Left	1	96%	98%	96%	92.67%
	2	88%	100%	88%	
	3	94%	94%	94%	
Total		76.67%	82.33%	75.67%	

By analysing the sequence of gestures, the hand SOS signal can be identified and



**Fig. 5** Output of the proposed application. It recognizes the SOS hand gestures from both sides

therefore the alarm procedure can be triggered. Experimental results show that the hand gesture is recognized rightly 75.67% of the time, achieving a 92.67% when the left hand is used. Therefore, this method can be deployed in a surveillance system or a camera-based system in order to make possible to help victims of kidnapping or abuse, among others.

**Acknowledgements** Virginia Riego would like to thank Universidad de León for its funding support for her doctoral studies.

## References

1. Aboah, A.: A vision-based system for traffic anomaly detection using deep learning and decision trees. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4207–4212 (2021)
2. Alnujaim, I., Alali, H., Khan, F., Kim, Y.: Hand gesture recognition using input impedance variation of two antennas with transfer learning. *IEEE Sensors Journal* **18**(10), 4129–4135 (2018). DOI 10.1109/JSEN.2018.2820000
3. Álvarez Aparicio, C., Guerrero-Higueras, M., Rodríguez-Lera, F.J., Ginés Clavero, J., Martín Rico, F., Matellan, V.: People detection and tracking using lidar sensors. *Robotics* **8**(3) (2019). DOI 10.3390/robotics8030075. URL <https://www.mdpi.com/2218-6581/8/3/75>
4. Azad, R., Azad, B., Kazerooni, I.T.: Real-time and robust method for hand gesture recognition system based on cross-correlation coefficient. *ACSIJ Advances in Computer Science: an International Journal* **2**(6) (2013)
5. Baptista, I., Shiaeles, S., Kolokotronis, N.: A novel malware detection system based on machine learning and binary visualization. In: *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6. IEEE (2019)
6. Berting, N.: A hand signal for help. Online campaign to support those experiencing violence in isolation. <https://www.whatdesigncando.com/stories/a-hand-signal-for-help/> (2020). [Online; accessed 6-May-2022]
7. Bhele, S.G., Mankar, V., et al.: A review paper on face recognition techniques. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* **1**(8), 339–346 (2012)
8. Chen, J.L., Ma, Y.W., Huang, K.L.: Intelligent visual similarity-based phishing web-sites detection. *Symmetry* **12**(10) (2020). DOI 10.3390/sym12101681. URL <https://www.mdpi.com/2073-8994/12/10/1681>
9. Dargan, S., Kumar, M.: A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities. *Expert Systems with Applications* **143**, 113114 (2020). DOI <https://doi.org/10.1016/j.eswa.2019.113114>. URL <https://www.sciencedirect.com/science/article/pii/S0957417419308310>
10. Daugman, J.: How iris recognition works. In: *The essential guide to image processing*, pp. 715–739. Elsevier (2009)
11. Fronteddu, G., Porcu, S., Floris, A., Atzori, L.: A dynamic hand gesture recognition dataset for human-computer interfaces. *Computer Networks* **205**, 108781 (2022). DOI <https://doi.org/10.1016/j.comnet.2022.108781>. URL <https://www.sciencedirect.com/science/article/pii/S1389128622000172>
12. Gadekallu, T.R., Srivastava, G., Liyanage, M., M., I., Chowdhary, C.L., Koppu, S., Maddikunta, P.K.R.: Hand gesture recognition based on a harris hawks optimized convolution neural network. *Computers and Electrical Engineering* **100**, 107836 (2022). DOI <https://doi.org/10.1016/j.compeleceng.2022.107836>. URL <https://www.sciencedirect.com/science/article/pii/S004579062200129X>



13. Gunes, H., Piccardi, M., Jan, T.: Face and body gesture recognition for a vision-based multimodal analyzer. In: M. Piccardi, T. Hintz, S. He, M.L. Huang, D.D. Feng (eds.) *Visual Information Processing 2003, Proceedings of the Pan-Sydney Area Workshop on Visual Information Processing, VIP2003, CRPIT*, vol. 36, pp. 19–28. Australian Computer Society (2003). URL <http://crpit.scem.westernsydney.edu.au/abstracts/CRPITV36Gunes.html>
14. Ito, K., Aoki, T.: [invited paper] recent advances in biometric recognition. *ITE Transactions on Media Technology and Applications* **6**(1), 64–80 (2018). DOI 10.3169/mta.6.64
15. Kapitanov, A., Makhlyarchuk, A., Kvanchiani, K.: Hagrid - hand gesture recognition image dataset (2022). DOI 10.48550/ARXIV.2206.08219. URL <https://arxiv.org/abs/2206.08219>
16. KASAPBAŞI, A., ELBUSHRA, A.E.A., AL-HARDANEE, O., YILMAZ, A.: Deep-aslr: A cnn based human computer interface for american sign language recognition for hearing-impaired individuals. *Computer Methods and Programs in Biomedicine Update* **2**, 100048 (2022). DOI <https://doi.org/10.1016/j.cmpbup.2021.100048>. URL <https://www.sciencedirect.com/science/article/pii/S2666990021000471>
17. Lahiani, H., Neji, M.: Hand gesture recognition method based on hog-lbp features for mobile devices. *Procedia Computer Science* **126**, 254–263 (2018). DOI <https://doi.org/10.1016/j.procs.2018.07.259>. URL <https://www.sciencedirect.com/science/article/pii/S1877050918312353>. Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 22nd International Conference, KES-2018, Belgrade, Serbia
18. Lin, W.C., Yeh, Y.R.: Efficient malware classification by binary sequences with one-dimensional convolutional neural networks. *Mathematics* **10**(4) (2022). DOI 10.3390/math10040608. URL <https://www.mdpi.com/2227-7390/10/4/608>
19. Liu, H., Wong, A.M.H., Kang, D.K.: Stationary hand gesture authentication using edit distance on finger pointing direction interval. *Scientific Programming* **2016**, 7427980 (2016). DOI 10.1155/2016/7427980. URL <https://doi.org/10.1155/2016/7427980>
20. LLC, G.: MediaPipe Hands. <https://google.github.io/mediapipe/solutions/hands> (2020). [Online; accessed 26-November-2021]
21. Oudah, M., Al-Naji, A., Chahl, J.: Hand gesture recognition based on computer vision: A review of techniques. *Journal of Imaging* **6**(8) (2020). DOI 10.3390/jimaging6080073. URL <https://www.mdpi.com/2313-433X/6/8/73>
22. Pakutharivu, P., Srinath, M.V.: A comprehensive survey on fingerprint recognition systems. *Indian Journal of Science and Technology* **8**(35), 1–7 (2015)
23. Singh, G., Singh, R.K., Saha, R., Agarwal, N.: Iwt based iris recognition for image authentication. *Procedia Computer Science* **171**, 1868–1876 (2020). DOI <https://doi.org/10.1016/j.procs.2020.04.200>. URL <https://www.sciencedirect.com/science/article/pii/S1877050920311819>. Third International Conference on Computing and Network Communications (CoCoNet'19)
24. Sætra, H.S.: The foundations of a policy for the use of social robots in care. *Technology in Society* **63**, 101383 (2020). DOI <https://doi.org/10.1016/j.techsoc.2020.101383>. URL <https://www.sciencedirect.com/science/article/pii/S0160791X20303262>
25. Takahashi, K.: Hand gesture recognition using MediaPipe. <https://github.com/Kazuhito00/hand-gesture-recognition-using-mediapipe> (2021). [Online; accessed 26-November-2021]
26. Toral-Álvarez, V., Álvarez-Aparicio, C., Guerrero-Higueras, Á.M., Fernández-Llamas, C.: Gait-based authentication using a rgb camera. In: J.J. Gude Prego, J.G. de la Puerta, P. García Bringas, H. Quintián, E. Corchado (eds.) *14th International Conference on Computational Intelligence in Security for Information Systems and 12th International Conference on European Transnational Educational (CISIS 2021 and ICEUTE 2021)*, pp. 126–135. Springer International Publishing, Cham (2022)
27. Zhao, J., Masood, R., Seneviratne, S.: A review of computer vision methods in network security. *IEEE Communications Surveys & Tutorials* (2021)

# Prediction of Smart Energy Meter Network Traffic Features for Anomaly Detection

Łukasz Saganowski and Tomasz Andrysiak

**Abstract.** In this article, we present the solution of anomaly detection in the network traffic features for critical smart metering infrastructure, performed with the use of a radio sensory network. The structure of the examined measurement network is presented and described. The signals analysed in the article represent selected features of the smart metering network traffic. The initial step involves the use of Isolation Forest algorithm for detection and elimination of outlier observations. In the next step, the motion features are converted into one-dimensional time series that are used for prediction. The predictions are achieved by means of one dimensional convolutional neural network model. For every traffic feature a multi-step prediction is calculated online in sliding time windows which consist of a few hundred samples from a single smart meter. Anomalies are detected by means of parameter estimation of the analysed signal and its comparative analysis to network traffic features which are used by maintenance staff. Efficiency of our method is examined with the use of an extended set of test traces from real network traffic. The obtained experimental results confirm effectiveness of the presented method.

**Keywords:** Time series analysis, outliers detection, neural networks, smart energy meter traffic prediction, anomaly/attack detection.

## 1 Introduction

The Smart Metering Communication Networks (SMCN) constitute one of most important parts of the smart grid system [1]. Typically, such networks consist of last-mile networks, access networks and backbone network. The backbone and access networks are realised with the use of classic methods i.e. using the Internet Protocol (IP) network as backbone and most often the General Packet Radio Service (GPRS) technology to access it. It should be noted that the mentioned classic solutions are not the only ones. There can also be other rather original solutions e.g. ones described in [2]. Last-mile smart metering networks use Power Line Communications (PLC), Radio Frequency (RF) or hybrid of these technologies. In this article, like in [3], the RF technology is taken into consideration. Anomalies in communication are caused by various factors, e.g. human or non-human activity, unintentional or intentional actions

---

Łukasz Saganowski

University of Science and Technology, email: luksag@pbs.edu.pl

Al. Prof. S. Kaliskiego 7, 85-796 Bydgoszcz, Poland

Tomasz Andrysiak

University of Science and Technology, email: andrys@pbs.edu.pl

Al. Prof. S. Kaliskiego 7, 85-796 Bydgoszcz, Poland

such as theft, for instance. There is a number of works dedicated to anomaly detection in communication networks, also in Smart Grid (SG) communications systems [4][6], including the last-mile area of their communication networks. However, in these works, the authors focus on anomaly detection in an IP network [7]. In our dissertation, we used RF technology for last-mile network where IP technology implementation was impossible because it would lengthen the radio frames and make the radio transmission unreliable. In the present article we suggest a network anomaly detection system based on the technique of outliers detection and prediction model, using One Dimensional Convolutional Neural Network (1D CNN). The process of anomaly detection (a network attack) is realized by comparison of parameters of a normal behavior (predicted from neural network responses) and parameters of a real network traffic.

This article is organized as follows: after the introduction, in Section 2, the security problems in smart metering network are presented. In Section 3, the proposed solution for anomaly detection using data traffic prediction is developed. Section 4 includes implementation details and experimental results. Conclusions are presented thereafter.

## 2 Security risks in smart metering networks

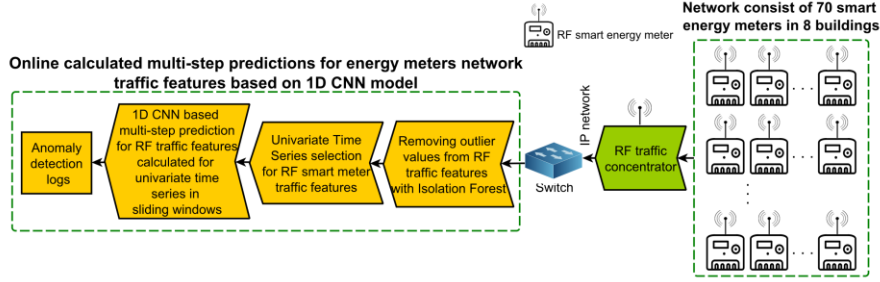
Assuring the security and protection of data collected by the smart metering systems is by no means as essential element of the smart metering communication solutions. Naturally, data gathered by smart meters can provide much information about the recipients' private lives. Moreover, the audience themselves can perform destructive activities towards the Advanced Metering Infrastructure (AMI), for instance: disturbing data saved in the meter, reconfiguration of settings and parameters of the counter, or interruption of data transmission [8]. Nevertheless, what appears to be a more serious problem is protection against cyberattacks. Application on a large scale of smart metering methods creates new entering possibilities for an unauthorized use by informational systems. Cyberattacks onto the SMCN security may be divided into two basic groups: passive and active. The former ones are all the attempts of an unauthorized access to data or the SMCN infrastructure, in which the attacker does not use emission of signals which may disturb or even disable correct operation of the system. The latter attacks, on the other hand, are all the attempts of an unauthorized access by the attacker on data or the SMCN infrastructure with the use of emission of any signals or activities that can be detected [9].

By performing a passive attack on SMN, the access to the transmitted data is gained through passive network monitoring where the attacker hides his presence. Another passive form of an attack on SMN are activities aimed at collection of network traffic analysis data in order to gain knowledge about its topology [10].

Active attacks can be divided into three subgroups [9]: *(i)* physical – destruction of a node, a node manipulation, Electromagnetic Pulse (EMP), *(ii)* attacks onto integrity, confidentiality or privacy of data (including unauthorized access to data), *(iii)* attacks on services (Denial-of-Service (DoS) or Distributed Denial of Service (DDoS)) – attacks directed at each SMCN network layer.

### 3 The methodology for the SMCN anomaly/attack detection

The main aim of the proposed solution is to detect anomalies or attacks in smart energy meters network. The examined network consists of 70 smart meters spread into 8 university buildings. A block representation of the proposed anomaly detection algorithm is presented in Fig. 1.



**Fig. 1.** Block representation of Smart Energy Meters Network with subsequent steps of the proposed algorithm for prediction of traffic features for anomaly detection purposes.

As presented in Fig. 1, traffic from smart meters is gathered by an RF traffic concentrator which is responsible for collecting traffic from smart energy meters and transforming traffic into IP packets. In the next step, RF traffic from smart meters in the form of IP packets is analyzed by the proposed anomaly detection method. Firstly, traffic is cleaned from outlier values by the Isolation Forest (see Chapter 3.1) algorithm in order to prepare traffic features for prediction in subsequent steps of the proposed methodology. Next, traffic features are transformed into a form of univariate time series (it is a time series where subsequent values appear in a constant period of time) where each series represent one RF traffic feature (see Table 1).

Part of the selected traffic features connected to physical quality of wireless signal such as SFW1 RSSIW: received signal strength indication [dBm] and SFW2 LQIW: link quality indicator parameter depends on the possibilities of hardware module embedded in the RF smart meter. Other traffic features connected to data link or network layers (SFW3-SFW8) are generated based on proprietary routing protocol that has been designed for the proposed smart meter network. Subsequently, univariate time series are prepared for prediction process.

Two modules are responsible for generation and transmission of traffic features in the proposed solution. The first module acting as a translation of traffic feature to IP packets is embedded in the RF traffic concentrator. The second module that calculates all traffic features SFW1-SFW8 based on IP packets from the RF concentrator in the first step of the proposed method removes outlier values from the RF traffic based on the Isolation Forest algorithm and eventually forms these features into univariate time series that are passed to prediction module. In order to achieve predictions, we proposed to use 1D CNN model. For every traffic feature we calculate a multi-step pre-

diction (3 steps ahead) calculated online in sliding time windows which consist of a few hundred samples. We can calculate subsequent prediction online in sliding time windows because traffic samples from a single smart meter are sent in 15 minutes' intervals. Based on prediction samples, we estimated prediction intervals and compared them with an actual traffic sample from the network (see Chapter 3.2 for details). In the end of algorithm we indicate detection logs for anomaly or attack detection which is used by maintenance staff.

### **3.1 Detection and elimination of outliers, based on the Isolation Forest algorithm**

Due to the nature of the SMCN infrastructure, there is high risk of significant fluctuations in the analyzed network traffic parameters, that is, high probability of outliers. The source of these fluctuations may be diverse, e.g. propagation of radio waves (environmental source), changes in the infrastructure (technical source), intentional user deception, hardware damage, network attack consequences, etc. Detection and elimination of outliers from the analyzed data set would constitute the basis for correct prediction processes. In the proposed solution, recognition of the outliers in the analyzed SMCN network traffic parameters is performed with the use of the Isolation Forest (IF) algorithm [11]. There are two stages of anomaly detection with the IF algorithm. The first, called the training stage, depends on building isolation trees by recursively partitioning the training set until the instances isolate or a certain tree height is obtained. Note that the tree height limit is automatically set based on the sub-sample size, which is approximately the average tree height. The reason for growing trees to an average height is that we are solely interested in data points that are shorter than average path length, as such points are more likely to be anomalous. The second, called the testing step, is based on running test instances through the isolation trees to obtain an anomaly result for each example. The detected anomalies depend on the expected path length for each test instance. In contrast, path lengths are determined by traversing each individual tree in the Isolated Forest example. Finding the highest anomalies is simply based on sorting the data in descending order. The first instances are the biggest anomalies.

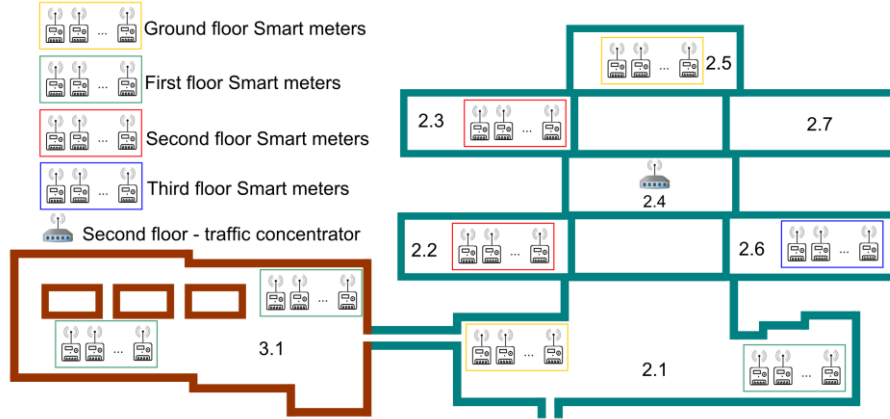
### **3.2 Calculation of multi-step prediction for anomaly detection with 1D CNN**

In the next step of the proposed solution, the traffic features are transformed into one-dimensional time series (i.e. time series in which successive values appear at constant intervals), where each series represents one RF traffic feature described in Table 1. In this context, one-dimensional time series are a specific class in which an attempt has been made to model and predict variable network traffic parameters using only the information contained in its own past values and possibly the current and past values of the erroneous components. Another activity is the prediction process carried out on the transformed one-dimensional time series. In order to obtain reliable forecasts (prediction results), we proposed the use of the 1D Convolutional Neural Networks model [12]. These networks are specific types of Artificial Neural Networks (ANN)

that focus on many deep layers of neural networks which use convolutional and pool strategies [13]. 1D CNN typically uses two- or three-dimensional neural layers for prediction problems that take the input in the same dimension to undergo a feature learning process. Developing CNN starts with a configuration of the external parameters known as hyper-parameters [14]. Adjusting the hyper-parameters is significant to get a better model with high predictive power. For each traffic feature, we calculate a multi-step prediction (i.e. 3 steps ahead) computed online in sliding time windows consisting of several hundred (300 – 500) samples [15]. We can calculate another prediction online in sliding time windows, as the traffic samples from one smart meter are sent in 15-minute intervals. It enables to eliminate possible prediction errors and improve its accuracy. Based on the predictive samples, we estimate the predictive intervals and compare them with the sample of real traffic from the network of metering meters. At the end of the algorithm, we indicate the anomaly detection and / or attack logs used by maintenance staff. They are the basis for monitoring and corrective actions carried out in the SMCN network.

#### 4 Experimental results

Experimental results were obtained with the use of real world smart meter network consisting of 70 smart meters located on four floors in eight buildings of Bydgoszcz University of Science and Technology in Poland. The RF Traffic concentrator was placed on the second floor. Localization of smart meters groups is presented in Fig. 2.



**Fig. 2.** Localization of RF smart energy meters groups in university campus (2.1-2.7 and 3.1 represents sub building numbers).

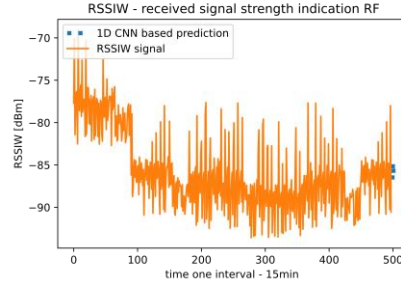
Traffic from smart meter network is transformed by the RF concentrator into IP packets. In subsequent step of the proposed solution, data from these packets is converted into univariate time series (it is a time series where subsequent values appear in con-

stant period of time) where every traffic feature is represented by time series. We captured set of traffic features which are presented in Table 1.

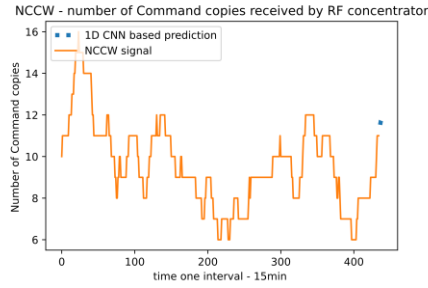
**Table 1.** Smart energy meters network RF traffic features' specification.

RF traffic Feature	RF traffic feature specification
SFW1	RSSIW: received signal strength indication for RF smart meter [dBm]
SFW2	LQIW: link quality indicator parameter (changes from: 0–127)
SFW3	PERPMW: packet error rate per time interval for smart meter RF node in [%]
SFW4	NPPMW: number of packets per time interval
SFW5	TTLW: packet time to live value for RF smart meter node
SFW6	NCCW: number of Command copies received by RF concentrator
SFW7	NRCW: number of RESPONSE copies received by RF concentrator
SFW8	NACW: number of ACK/CANCEL copies received by RF concentrator

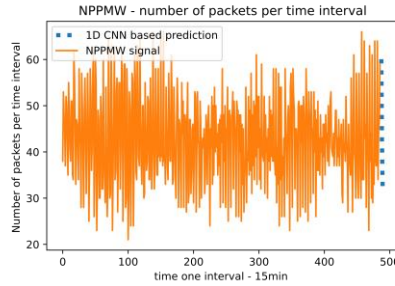
Traffic features represent for example wireless signal quality like SFW1 – RSSIW received signal strength indication, SFW2 – LQIW link quality indicator or parameters connected to transmission and routing protocols that are used in smart meter network: SFW6 – NCCW number of Command copies, NRCW: number of RESPONSE copies or NACW: number of ACK/CANCEL copies received by the RF concentrator. Traffic features have to be prepared for passing them into 1D CNN neural network model without outlier values. For removing these values we used the Isolation Forest algorithm (see Chapter 3.1). Subsequently, traffic features are formed into univariate time series where for each one of them we calculate multi-step prediction by means of 1D CNN neural network model. One dimensional CNN model in our case consists of a convolutional hidden layer for one-dimensional sequence of traffic. The next layer is a polling layer responsible for clearing of the convolutional layer outcome to achieve the most important values. The subsequent flattened layer, placed between a convolutional layer and a dense layer, is responsible for reducing feature maps into a vector of one-dimensional values. Dense and fully connected layer interprets features from the convolutional layer. In the proposed methodology we calculate multi-step forecasting separately for every univariate time series. We perform 3-step-ahead prediction for every traffic feature from Table 1. We recalculate predictions online for every traffic in sliding time windows consisting of few hundreds (300 – 500 samples) traffic feature samples. We can constantly calculate new predictions because traffic samples arrive relatively rarely (every smart meter sends data in 15-minute periods). Basing on the achieved results, we can calculate prediction intervals (see Chapter 3.2) for every traffic feature in order to compare online the extracted real traffic sample values. If the real traffic values extracted online exceed the estimated prediction interval, we indicate possible anomaly or attack. Examples of multi-step predictions are presented in Fig. 3 – Fig. 5. In these figures we can observe prediction values (presented as blue rectangular dots) calculated for SFW1 - RSSIW: received signal strength indication, SFW6 - NCCW: number of Command copies and SFW4 - NPPMW: number of packets per time interval.



**Fig. 3.** Multi-step (3 samples ahead) prediction example for RSSIW univariate time series RF traffic feature achieved by 1D CNN based on prediction model algorithm (presented as blue dots).



**Fig. 4.** Multi-step (3 samples ahead) prediction example for NCCW univariate time series RF traffic feature achieved by 1D CNN based prediction model algorithm (presented as blue dots).



**Fig. 5.** Multi-step (3 samples ahead) prediction example for NPPMW univariate time series RF traffic feature achieved by 1D CNN based prediction model algorithm (presented as blue dots).

In order to evaluate prediction accuracy achieved by the proposed method we calculated RMSE [16] and Scatter Index (SI) [%] [16] values. Additionally, we compared them to the results obtained by Neural Network AutoRegressive (NNAR) model [17]. RMSE and SI were calculated for three different signals (RSSIW, NCCW and



NPPMW). We can see from Table 2 that more accurate predictions were achieved for 1D CNN based model.

**Table 2.** Comparisons of prediction accuracy for 1D CNN and NNAR neural network based models for RSSIW (Received signal strength indication wireless), NCCW (Number of Command copies wireless) and NPPMW (Number of Packets per Time Interval wireless) RF traffic features using RMSE (Root Mean Square Error) and SI [%] (Scatter Index).

Prediction model	RMSE-RSSIW	SI [%]-RSSIW	RMSE-NCCW	SI [%]-NCCW	RMSE-NPPMW	SI [%]-NPPMW
NNAR	4.77	5.82	3.85	25.46	23.56	28.62
1D CNN	0.84	0.98	0.68	6.25	5.89	13.49

Most of SI values for the predicted interval are approximately below 6% so we can state that the proposed model fits very well to the characteristics of the examined signal. SI values lower than 15% for signals with noise character (e.g. NPPMW) are also acceptable. Evaluation of the proposed anomaly detection methodology was performed for real world network presented schematically in Fig. 2. Generation of anomalies in real world environment of our test bed was determined by technical possibilities that could be applied to different physical locations of the RF smart meter clusters that were spread in our university buildings. As a disturbance generation factor we used existing natural or human-made obstacles. We installed clusters of the RF smart meters in places where different RF Internet of Things (IoT) networks or different RF devices from Industrial, Scientific and Medical (ISM) bands were active. Additionally, in some of smart meter locations, we installed redundant RF smart meters that operate as disturbance devices by generating random packets that interfere with smart meter routing protocol in frequencies used by our network. In some locations we also added devices that indirectly interfere with the RF smart meters groups by injecting disturbances into AC power mains in close proximity of the smart meters. As a disturbance device we used a generator that is used during EMC compatibility test according to the IEC standard 61000-4-4[18]. The anomaly generation methods are summarized in points a) – f):

- a) Radio frequency interferences that come from sources existing in university buildings: existing RF Internet of Things (IoT) networks, different RF devices from Industrial, Scientific and Medical (ISM) bands.
- b) Different obstacles: natural and human-made.
- c) Intentional damages of smart power meter groups.
- d) Intentionally generated RF frequency disturbances by different RF devices that operate in frequencies used by smart energy meters network (impact on selected smart meter groups).
- e) Attacks that have direct impact on routing protocol used in the smart meter network.
- f) Conducted disturbances injected by means of Active Power (AC) power mains (according to IEC standard 61000-4-4 [18]) that have indirect impact on smart power meter groups.

Taking into consideration all types of anomalies and attacks that come from environment and deliberately generated attacks or anomalies, we achieved results summarized in Table 3. We compared results achieved for the proposed model based on 1D CNN neural network to the model utilizing NNA.

**Table 3.** Detection Rate DR[%] and False Positive FP[%] results comparison for 1D CNN prediction based model and NNAR neural network model.

Smart Meter Feature	NNAR DR[%]	1D CNN DR[%]	NNAR FP[%]	1D CNN FP[%]
SFW1	89.21	98.57	4.77	2.84
SFW2	85.34	97.84	4.86	3.68
SFW3	84.65	98.45	6.11	3.58
SFW4	88.24	95.63	5.25	4.22
SFW5	86.26	94.26	4.82	3.62
SFW6	90.72	98.75	5.22	3.21
SFW7	86.47	94.42	6.12	4.24
SFW8	84.78	96.15	6.74	5.52

Due to the fact that 1D CNN model gives us better prediction accuracy than NNAR model, we obtained better DR [%] values and lower values of FP [%] parameter. Anomalies or attacks that have a direct impact on the RF signal quality can be observed on traffic features directly connected to signal quality like SFW1 - RSSIW or SFW2 - LQIW but also have indirect influence on traffic features from data link or network layers like SFW5 - TTLW, SFW6 - NCCW, SFW7 - NRCW and SFW8 - NACW. Taking into consideration all results from Table 3. we achieved DR [%] that changes between 98.75 - 94.26 and FP [%] 2.84 - 5.52 for all source of anomalies and attacks generated in our smart energy meter testbed so we can state that the proposed methodology can be considered for maintenance purposes in this type of metering network. For anomaly detection purposes, false positive values less than 10% in smart metering or more generally in IoT network can be considered as an acceptable result for anomaly detection purposes [19][20].

## 5 Conclusions

In the article we proposed a solution for anomaly and attack detection in smart energy meter network. The proposed explication was tested with real world smart meter network that consists of 70 meters placed in 8 university buildings. We captured and extracted RF traffic features from a RF concentrator. We proposed a model with the main steps consisting of: RF traffic capture, traffic features extraction, outlier values removal based on the Isolation Forest algorithm, preparation of univariate time series to the requirements of neural network, model calculation based on 1D CNN Convolutional Neural Network, multi-step forward prediction, estimation of prediction intervals based on prediction values and, finally, comparison of the actual extracted traffic feature values to the estimated prediction interval. 1D CNN prediction intervals are

calculated online in sliding time windows consisting of approximately few hundreds of samples. We achieved acceptable results where DR [%] changes between 98.75 - 94.26 and FP [%] 2.84 - 5.52 for all source of anomalies and attacks generated in our smart energy meter testbed. The proposed solution provides promising results so it can be considered applicable for maintenance purposes in smart energy meter network or, more generally, for IoT network with similar types of devices.

## References

1. Finster, S., Baumgart.: Privacy-Aware Smart Metering: A Survey, *IEEE Communications Surveys & Tutorials*, 17(2), pp. 1088–1101, (2015)
2. Bilgin, B.E., Baktir, S., Gungor, V.C.: Collecting smart meter data via public transportation buses, *IET Intelligent Transport Systems*, 10(8), pp. 515–523, (2016)
3. Kulkarni, P., Gormus, S., Fan, Z.: A mesh-radio-based solution for smart metering networks, *IEEE Communications Magazine*, 50(7), pp. 86–95, (2012)
4. Guo Y, Ten, C-W., Hu, S.: Preventive Maintenance for Advanced Metering Infrastructure Against Malware Propagation, *Transactions on Smart Grid*, 7(3), pp.1314–1328, (2016)
5. Berthier, R., I. Urbina, D.I., Cárdenas A.A., Guerrero M., Herberg U., Jetcheva J. G., Mashima, D., Huh, J.H., Bobba R.B.: On the Practicality of Detecting Anomalies with Encrypted Traffic in AMI, *Conference on Smart Grid Communications*, pp. 890–895, (2014)
6. Giani A., Bitar E., Garcia M., McQueen M, Khargonekar P. and Poolla K.: Smart Grid Data Integrity Attacks, *IEEE Trans. on Smart Grid*, 4(3), pp. 1244–1253, (2013)
7. ITU-T Recommendation G.9904 (10/2012): Narrowband orthogonal frequency division multiplexing power line communication transceivers for PRIME networks, (2013)
8. Balakrishnan, M.: Security in Smart Meters. Free scale Semiconductor Inc. Doc. number: SEC s. MTMTRWP REV0, Arizona (2012)
9. Liu, J.; Xiao, Y.; Li, S.; Liang, W.; Chen, C.; ,Cyber Security and Privacy Issues in Smart Grids, *Communications Surveys & Tutorials*, IEEE, vol. PP, no.99, pp.1-17, (2012)
10. Sarma, H.K.D., Kar, A.: Security Threats in Wireless Sensor Networks, Elsevier, (2006)
11. Fei, T. L., Kai, M.T., Zhi-Hua, Z.: Isolation Forest, *Eighth IEEE International Conference on Data Mining*, pp. 413–422, (2008)
12. Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M. 1D convolutional neural networks and applications: A survey. *Mech. Syst. Signal Process.* 151, 107398, 2019
13. Yoo, Y.: Hyperparameter optimization of deep neural network using univariate dynamic encoding algorithm for searches, *Knowledge-Based Systems*, 178, pp. 74–83, (2019)
14. Aszemi, N.M., Dominic, P.: Hyperparameter optimization in convolutional neural network using genetic algorithms. *Int. J. Adv. Comput. Sci.*, 10, 269–278, (2019)
15. Harbola, S., Coors., V.: One dimensional convolutional neural network architectures for wind prediction, *Energy Conversion and Management*, 195, pp. 70-75, 2019
16. Bryant, Mary A., Hesser Tyler J., Jensen Robert E.: Evaluation Statistics Computed for the Wave Information Studies (WIS), Technical Report ERDC/CHL CHETN-I-91, (2016)
17. Cogollo, M. R., Velasquez, J. D.: Are Neural Networks Able To Forecast Nonlinear Time Series With Moving Average Components?, *Latin America Transactions*, 13(7), (2015)
18. IEC 61000-4-4 [http://www.iec.ch/emc/basic\\_emc/basic\\_emc\\_immunity.htm](http://www.iec.ch/emc/basic_emc/basic_emc_immunity.htm)
19. Garcia-Font, V., Garrigues, C., Rifà-Pous, H.: A Comparative Study of Anomaly Detection Techniques for Smart City Wireless Sensor Networks, *Sensors*, 16(6), (2016)
20. Xie, M., Han, S., Tian, B., Parvin, S.: Anomaly detection in wireless sensor networks: A survey, *Journal of Network and Computer Applications*, 34(4), pp. 1302–1325, (2011)

# An anomaly detection approach for realtime identification systems based on centroids

Álvaro Michelena 0000-0003-0134-5660, Francisco Zayas-Gato  
0000-0002-0994-1961, Esteban Jove 0000-0002-0625-359X, José-Luis  
Casteleiro-Roca 0000-0001-9740-6477, Héctor Quintián 0000-0002-0268-7999,  
Óscar Fontenla-Romero 0000-0003-4203-8720 and José Luis Calvo-Rolle  
0000-0002-2333-8405

**Abstract** The present research describes a novel adaptive anomaly detection method to optimize the performance of nonlinear and time-varying systems. The proposal is based on combining the real-time identification algorithm, Recursive Least Squares, with a centroid-based methodology. For anomaly detection, the method compares the current system dynamics with the average (centroid) of the dynamics identified in previous states for a specific setpoint. If the dynamics difference exceeds a certain threshold, the system classifies it as an anomaly. Otherwise, the centroid is updated by introducing the newly identified data. Finally, the proposed method was tested on a real system, in this case, on the level control plant, obtaining a good performance in anomaly detection.

## 1 Introduction

Nowadays, industrial systems are becoming heavily instrumented, involving many sensors and actuators. Therefore, all its components must work properly to ensure a correct operation process and improve and optimize system performance. Also, detecting abnormal behavior or any data that is different from the normal pattern is becoming necessary to achieve robust and safe system operations.

---

Álvaro Michelena, Esteban Jove and José Luis Calvo-Rolle  
University of A Coruña, CITIC. Campus de Elviña, s/n, 15008 A Coruña, Spain, e-mail: al-  
varo.michelena@udc.es, esteban.jove@udc.es, jlcalvo@udc.es

Francisco Zayas-Gato, José-Luis Casteleiro-Roca and Héctor Quintián,  
University of A Coruña. CTC, Department of Industrial Engineering. Avda. 19 de febrero s/n,  
15405, Ferrol, A Coruña, Spain, e-mail: f.zayas.gato@udc.es, jose.luis.casteleiro@udc.es, hec-  
tor.quintian@udc.es

Óscar Fontenla-Romero  
University of A Coruña, LIDIA, Faculty of Computer Science, Campus de Elviña, s/n, 15071 A  
Coruña, Spain, e-mail: oscar.fontenla@udc.es

In any process or application, anomalies can be produced for many reasons, such as measurement errors in the sensors, actuators failures, human mistakes and so on. For this reason, anomaly detection is a key factor in many industrial processes and applications, such as, fault detection in power cells [17, 23], in biocomponents production [13, 18], fault detection in medical field [25, 22], in the computer science field [10, 21, 27] and so on [11, 12, 32, 14, 29, 19, 2].

Recently, the complexity of today's industrial systems and the improvement of computational resources have led many researchers to focus on optimizing and developing new anomaly detection techniques and their implementation in many applications, processes, and systems. Depending on the application and process characteristics, various techniques can be used to detect anomalous behavior [6, 33, 26, 31]. For example, in [16], the authors applied semi-supervised (also known as one-class) techniques to detect anomalies in industrial control loops. In addition, in [24], supervised machine learning techniques are used to detect anomalies in industrial control systems. These two investigations require a labeled dataset to train their proposals, which can sometimes be challenging.

The complexity of obtaining quality labeled datasets in many processes has led to the increasing use of unsupervised techniques for anomaly detection. For example, in [15] this kind of unsupervised techniques is used to detect anomalies in industrial processes. Although a good system performance was obtained, the proposal was not wholly automatic because it required expert analysis to define the boundaries of the clusters obtained according to whether they correspond to anomalous behaviors.

In addition, all these investigations presented are not adaptive, causing temporary changes in the systems' dynamics to lead to losses in performance when detecting anomalies. Therefore, developing adaptive techniques whose performance does not depend on the temporal system variations is necessary.

For all these reasons and considering the relevance of anomaly detection systems, this paper proposes a novel proposal to detect anomalies on industrial control loops. The method is based on combining a real-time identification algorithm, Recursive Least Squares, and centroid estimation to detect anomalies in nonlinear and time-varying systems.

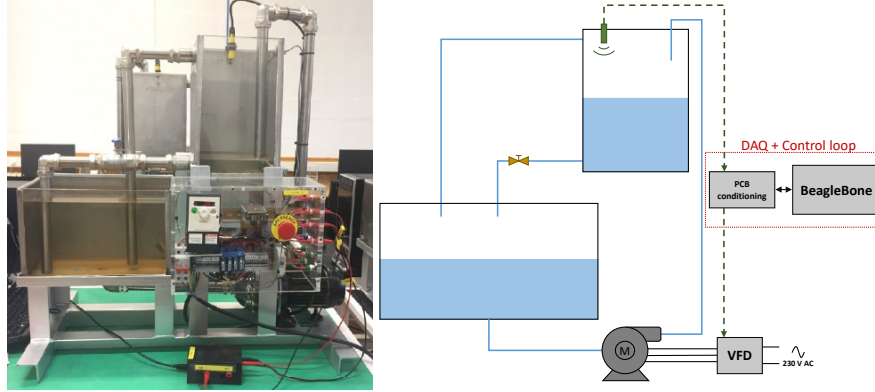
The present paper is structured as follows: after the Introduction, Section 2 describes the case of study. Then, Section 3 presents the proposed anomaly detection method. Section 4 lists the experiments and results. Finally, the conclusions and future works are exposed in Section 5.

## 2 Case of Study

This section describes the level control plant and the control loop used for its correct operation. In addition, the dataset used to perform the functional tests is also described.

## 2.1 Level control plant

The level control plant is one of the educational plants of the Optimization and Control Laboratory of the Faculty of Engineering of the University of A Coruña. This system is used to control the tank filling level. Figure 1 shows the real system and its scheme.



**Fig. 1** Level control plant.

This system has two tanks that are placed at different heights. The filling level is controlled in the upper tank, while the lower tank is used to store the water discharged from the upper tank through a pipe. A three-phase centrifugal pump controlled by a variable frequency drive (VFD) is used to propel the water from the lower tank to the upper tank. An ultrasonic sensor is used to measure the amount of water stored in the upper tank. In addition, the system also includes a manual valve to control the emptying flow rate of the upper tank.

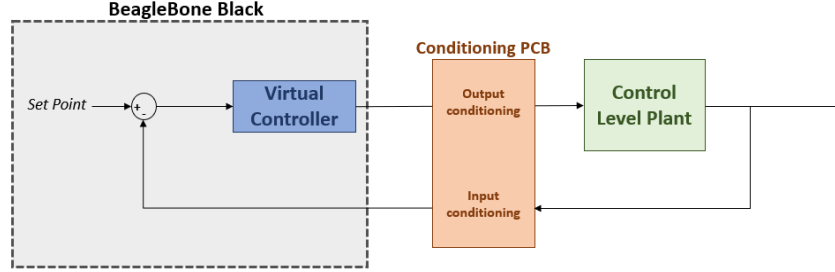
## 2.2 System integration and its control implementation

A virtual Proportional Integrative and Derivative (PID) controller is implemented to control the water level plant. This controller corresponds to a python algorithm which is executed on a BeagleBone Black board. Since the input and output system signals are 0/10V DC and the BeagleBone Black features, it is necessary to add a conditioning circuit to ensure the correct functionality of the control loop.

The control loop follows the diagram shown in Figure 2. In this way, the virtual controller reads the fill level measured through the input conditioning circuit. Depending on the setpoint, which corresponds to the desired filling level, the PID controller generates a control signal sent to the VFD control input through the con-

ditioning circuit. Finally, the VFD generates a three-phase AC signal proportional to the control value, thus controlling the water flow rate propelled to the upper tank.

To simplify the control loop, the filling level and centrifugal pump power correspond to percentage values.



**Fig. 2** Control loop diagram.

### 2.3 Dataset

The control signal, set point, and process value were recorded during 35 minutes of normal plant operation to obtain the data set used. An adaptive PID controller based on the Dahlin PID [3] was implemented for data collection. For this purpose, the manually piloted valve was fully open. The sampling time was 0.5 seconds, so 4200 samples were recorded for different operating points from 25% to 85% in steps of 10%. The range limitation is necessary since the plant does not perform well for below or above percentage values due to its design. The 10% increment is implemented since the change in system dynamics is not very appreciable for smaller increments.

The real data registered correspond to a correct operation, and this work aims to detect anomalies. Therefore, a total of 30 anomalies have been generated by modifying the process value signal by deviating a random percentage between 4 and 10% of the total filling. Due to the control loop used to control the plant, the system output signal affects the control signal, so this signal is also modified in the same samples in an inverse way to the output signal deviation. Since a variation of the control signal affects the system output to a greater extent, this signal is modified the half percentage varied in the process value signal. These anomalies simulate small interferences in the signals or problems of valve obstruction, leaks, etc. Likewise, the sensor output signal has also been modified for 0-100% values to emulate ultrasonic sensor measurement failures.

### 3 Methodological Approach

The main goal of this research is to develop a novel anomaly detection system by means of on-line identification algorithms. Generally speaking, the system uses the RLS on-line identification algorithm to detect the system dynamic in a specific time, and this value is compared to the centroid of the other identification dynamics for that setpoint. If the measured distance between the centroid and the new data is greater than a given threshold, an anomaly is detected. Therefore, this method consist of two main stages: the RLS identification algorithm to identify the system dynamics and the fault detection process based on centroids. The proposal scheme is shown in Figure 3 where SP correspond to setpoint value, CP is the control process signal, PV is the process value and  $a_0$ ,  $a_1$  and  $b_0$  are the transfer function parameters.

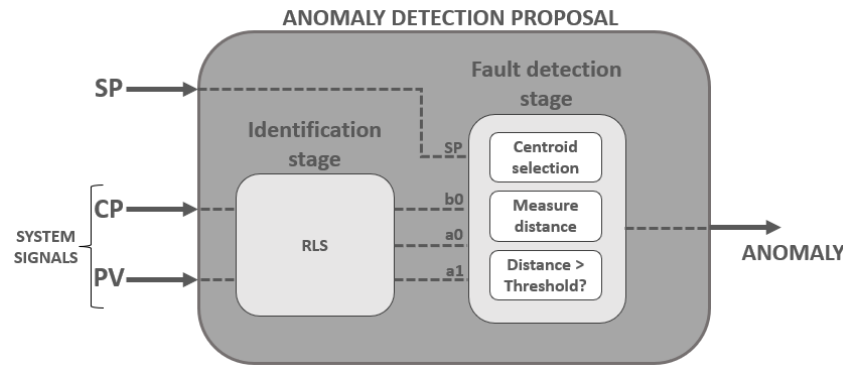


Fig. 3 Proposal scheme

For a better understanding this section has been divided in two subsections.

#### 3.1 On-line identification stage. Recursive Least Square.

The Recursive Least Square (RLS) method is one of the most used methods in the field of on-line identification, due to its simplicity and promptness. The aim of this method is obtaining the value of transfer function parameters, defined in  $\theta$  vector, that minimize the prediction error and that best correlate the system input and output signals [1]. Also, this algorithm can be executed on a great variety of devices without large computational capabilities such as low-cost board (Arduino, BeagleBone...). Therefore, RLS method can be integrated into multiple applications and systems. In this research, the tank fill level plant is identified as a second order transfer function with time delay since it is the transfer function that best fits most systems, Equation 1. Therefore, the RLS algorithm obtains the parameters  $a_0$ ,  $a_1$  and  $b_0$  from the input and output signal of the system.



$$G_p(z^{-1}) = \frac{b_0 z^{-k}}{1 - a_0 z^{-1} - a_1 z^{-2}} \quad (1)$$

It is important to highlight that RLS is a recursive method so an exponential forgetting factor, defined as  $\lambda$ , frequently  $\lambda \in [0.8, 1]$  [34], is added to the algorithm. In order to achieve a good algorithm performance its tuning must be an indispensable key factor. To adjust this parameter, it is important to consider how its value affects the identification process. In this case, if  $\lambda$  corresponds to a low value, the identification algorithm have less memory, and its sensibility is higher, which may produce some errors in the identified parameters since RLS becomes sensitive to system noise. On contrast, higher values, close to 1, causes less sensitivity on RLS algorithm since its memory is greater. In addition, lower values cause that identification algorithm is more robust to system noise.

### 3.2 Fault detection stage

Once the RLS algorithm has identified the system, the transfer function parameters and the setpoint value are sent to the anomaly detection stage. This stage is divided into three blocks: the data cloud storage and centroid calculation block, the distance measurement module, and the decision module (comparing the distance with the threshold).

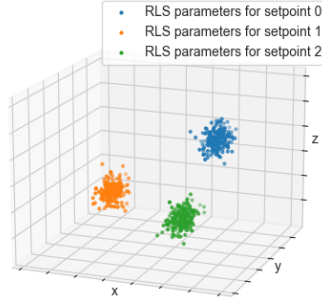
#### Data cloud storage and centroid calculation block

This stage stores the transfer parameters identified for normal operating points and calculates the centroid (mean value). As already mentioned, the control level plant is a non-linear system, so it is essential to separate the parameters of the transfer functions according to the selected setpoint, thus forming small data subsets for a correct anomaly detection system performance. Figure 4 shows an example of the subsets created for different setpoints.

On the other hand, since the setpoint value is continuous and the system dynamics do not present significant differences for close setpoints, seven subsets of data associated with a range of setpoint values from 20% to 90%, filled with 10% increments, will be differentiated. Each of these subsets has a centroid associated with it.

The number of samples in each data subset corresponds to the last  $n$  samples identified for a given setpoint value, where  $n$  is a value to be determined by the user.

To conclude, this module selects the data subset that corresponds to the entered setpoint value and calculates its centroid .

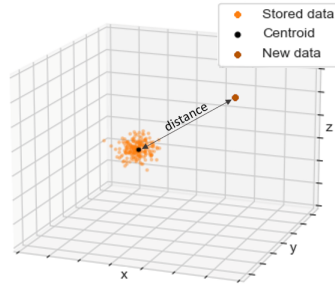


**Fig. 4** Identified parameter subsets in nonlinear systems.

### Distance measurement module.

After obtaining the centroid value, the euclidean distance,  $d$ , between the centroid and new data obtained by the RLS identification system, is measured by means of Equation 2. Also, in Figure 5 a distance between two three-dimensional points is shown.

$$\|d(a, b)\| = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \dots + (a_n - b_n)^2} \quad (2)$$



**Fig. 5** Euclidean distance between two three-dimensional points.

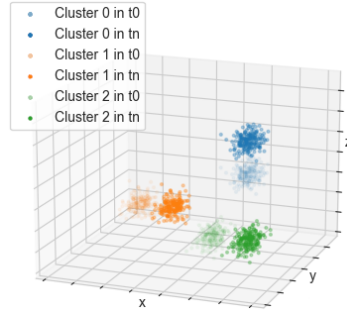
### Decision module.

Once the distance between the centroid and the identified datum is measured it is compared to the threshold value selected by the user, Equation 3.

$$\|d(\text{centroid}, \text{new data})\| > \text{threshold} \rightarrow \text{Anomaly detected} \quad (3)$$

In this way, if the distance measured is greater than a certain threshold, an anomaly is detected. On contrast, if a normal operating data is identified, its value is added to the cluster subset associated with the selected operating point. Therefore the system becomes adaptive by employing iterative centroids that adapt to the slow changes in the dynamics of the time-varying systems. An example of how these centroids are shifted is shown in Figure 6.

It is important to note that the threshold values selected, are not random but they correspond to the maximum distances measured between the centroids obtained and the furthest points of their corresponding normal operating dataset.



**Fig. 6** Example of system dynamics changes over time.

## 4 Experiments and results

Several tests were performed to test and validate the proposed anomaly detection method. First, it was necessary to obtain the best value for the centroid data window, the forgetting factor and the distance threshold. To determine the threshold value, the original dataset without the added anomalies was analyzed, and the system was executed to obtain the maximum measured distance between an identified data and its associated centroid. Through this process the maximum distance was found to be 1, 0.9 and 1.1 depends on the RLS forgetting factor.

The value of the forgetting factor and the distance threshold were then adjusted empirically. For this purpose, the system was run with the dataset generated with the anomalies for different values of the parameters mentioned above. Since it is an anomaly detection problem, the metric that was used to compare the results obtained was f1-score, which is the harmonic mean between the precision and the recall. If f1-score is 1, it means that all anomalies were detected and any normal data points were classified as an anomalies. In Table 1 the results obtained for different values are shown.

As shown in Table 1 the best value of distance threshold is 1.1, and the best forgetting factor value is 0.94. However, for these settings, with any value of the

**Table 1** Results obtained for different parameter values.

Threshold	Centroid data window (n)	Forgetting factor	Precision	Recall	F1-score
0.9	15	0.92	0.8108	1.0	0.8955
		0.94	0.9375	1.0	0.9677
		0.96	0.1971	0.8667	0.1032
	20	0.92	0.7692	1.0	0.8696
		0.94	0.9375	1.0	0.9677
		0.96	0.1971	0.8667	0.0998
	25	0.92	0.7317	1.0	0.8451
		0.94	0.9375	1.0	0.9677
		0.96	0.0545	0.8667	0.1032
1	15	0.92	0.9091	1.0	0.9524
		0.94	0.9677	1.0	0.9836
		0.96	0.0448	0.9667	0.1905
	20	0.92	0.8824	1.0	0.9375
		0.94	0.9677	1.0	0.9836
		0.96	0.0448	0.8667	0.1905
	25	0.92	0.8333	1.0	0.9091
		0.94	0.9677	1.0	0.9836
		0.96	0.0620	0.8333	0.1905
1.1	15	0.92	0.9677	1.0	0.9836
		0.94	1.0	1.0	1.0
		0.96	0.0287	0.8333	0.3273
	20	0.92	0.9375	1.0	0.9677
		0.94	1.0	1.0	1.0
		0.96	0.0362	0.8333	0.1905
	25	0.92	0.9375	1.0	0.9677
		0.94	1.0	1.0	1.0
		0.96	0.0512	0.8667	0.1962

centroid data window the same results are obtained and all anomalies are detected. Also any normal operation sample is classified as an anomaly.

Once the parameters were adjusted, the system was run in the real plant and it was verified that the system worked correctly, although in cases where the signals are affected by high noise, anomalies are detected in some cases when in fact they correspond to normal operating points.

## 5 Conclusions and Future work

This research presents a novel method based on on-line identification RLS algorithm to detect anomalies and abnormal behaviors in nonlinear and time varying systems. In contrast to other anomaly detection systems, this method presents an adaptive solution for time varying systems that can be updated in real time. The proposal has been tested on a real system, in this case, the laboratory filling level control plant.

The results obtained indicate that the system is capable of detecting many anomalies. However, tuning some parameters such as the RLS forgetting factor or distance

threshold is mandatory to achieve a good method performance and high accuracy. Previous knowledge of the system dynamics is necessary to tune these values. However, in high noise situations, the systems identified some standard data as an anomaly since its signal variations can be modified by the system dynamics.

On the other hand, in future works, the proposed method will be implemented in other laboratory plants, such as temperature control plants. Moreover, it would be essential to compare the proposed method with other fault detection techniques such as one-class methods or hybrid techniques [5, 7, 8, 20, 28, 30, 9, 4], among others.

## Acknowledgements

CITIC, as a Research Center of the University System of Galicia, is funded by Consellería de Educación, Universidade e Formación Profesional of the Xunta de Galicia through the European Regional Development Fund (ERDF) and the Secretaría Xeral de Universidades (Ref. ED431G 2019/01).

## References

1. Åström, K.J., Wittenmark, B.: Adaptive control. Courier Corporation (2013)
2. Basurto, N., Arroyo, Á., Cambra, C., Herrero, Á.: A hybrid machine learning system to impute and classify a component-based robot. *Logic Journal of the IGPL* (2022)
3. Bobál, V., Böhm, J., Fessl, J., Macháček, J.: Self-tuning PID Controllers. Springer (2005)
4. Casado-Vara, R., Sittón-Candanedo, I., la Prieta, F.D., Rodríguez, S., Calvo-Rolle, J.L., Venayagamoorthy, G.K., Vega, P., Prieto, J.: Edge computing and adaptive fault-tolerant tracking control algorithm for smart buildings: A case study. *Cybernetics and Systems* **51**(7), 685–697 (2020)
5. Crespo-Turrado, C., Casteleiro-Roca, J.L., Sánchez-Lasheras, F., López-Vázquez, J.A., De Cos Juez, F.J., Pérez Castelo, F.J., Calvo-Rolle, J.L., Corchado, E.: Comparative study of imputation algorithms applied to the prediction of student performance. *Logic Journal of the IGPL* **28**(1), 58–70 (2020)
6. Dutta, V., Pawlicki, M., Kozik, R., Choraś, M.: Unsupervised network traffic anomaly detection with deep autoencoders. *Logic Journal of the IGPL* (2022)
7. Fernandez-Serantes, L.A., Casteleiro-Roca, J.L., Calvo-Rolle, J.L.: Hybrid intelligent system for a half-bridge converter control and soft switching ensurement. *Revista Iberoamericana de Automática e Informática Industrial* **00**, 1–5 (3 2022)
8. Fernández-Serantes, L.A., Estrada Vázquez, R., Casteleiro-Roca, J.L., Calvo-Rolle, J.L., Corchado, E.: Hybrid intelligent model to predict the soc of a lfp power cell type. In: *International conference on hybrid artificial intelligence systems*. pp. 561–572. Springer (2014)
9. García-Ordás, M.T., Alaiz-Moretón, H., Casteleiro-Roca, J.L., Jove, E., Benítez-Andrades, J.A., García-Rodríguez, I., Quintián, H., Calvo-Rolle, J.L.: Clustering techniques selection for a hybrid regression model: A case study based on a solar thermal system. *Cybernetics and Systems* **0**(0), 1–20 (2022)
10. Go, G.M., Bu, S.J., Cho, S.B.: Insider attack detection in database with deep metric neural network with monte carlo sampling. *Logic Journal of the IGPL* (2022)
11. Gonzalez-Cava, J.M., Arnay, R., Mendez-Perez, J.A., León, A., Martín, M., Reboso, J.A., Jove-Perez, E., Calvo-Rolle, J.L.: Machine learning techniques for computer-based decision

- systems in the operating theatre: application to analgesia delivery. *Logic Journal of the IGPL* **29**(2), 236–250 (2021)
12. Guevara, C., Santos, M.: Intelligent models for movement detection and physical evolution of patients with hip surgery. *Logic Journal of the IGPL* **29**(6), 874–888 (2021)
  13. Jove, E., Casteleiro-Roca, J., Quintián, H., Méndez-Pérez, J., Calvo-Rolle, J.: Anomaly detection based on intelligent techniques over a bicomponent production plant used on wind generator blades manufacturing. *Revista Iberoamericana de Automática e Informática industrial* **17**(1), 84–93 (2020)
  14. Jove, E., Blanco-Rodríguez, P., Casteleiro-Roca, J.L., Quintián, H., Moreno Arboleda, F.J., López-Vázquez, J.A., Rodríguez-Gómez, B.A., Meizoso-López, M.D.C., Piñón-Pazos, A., De Cos Juez, F.J., et al.: Missing data imputation over academic records of electrical engineering students. *Logic Journal of the IGPL* **28**(4), 487–501 (2020)
  15. Jove, E., Casteleiro-Roca, J.L., Quintián, H., Méndez-Pérez, J.A., Calvo-Rolle, J.L.: A fault detection system based on unsupervised techniques for industrial control loops. *Expert Systems* **36**(4), e12395 (2019)
  16. Jove, E., Casteleiro-Roca, J.L., Quintián, H., Simić, D., Méndez-Pérez, J.A., Luis Calvo-Rolle, J.: Anomaly detection based on one-class intelligent techniques over a control level plant. *Logic Journal of the IGPL* **28**(4), 502–518 (2020)
  17. Jove, E., Casteleiro-Roca, J.L., Quintián, H., Zayas-Gato, F., Vercelli, G., Calvo-Rolle, J.L.: A one-class classifier based on a hybrid topology to detect faults in power cells. *Logic Journal of the IGPL* (2021)
  18. Jove, E., Casteleiro-Roca, J.L., Casado-Vara, R., Quintián, H., Pérez, J.A.M., Mohamad, M.S., Calvo-Rolle, J.L.: Comparative study of one-class based anomaly detection techniques for a bicomponent mixing machine monitoring. *Cybernetics and Systems* **51**(7), 649–667 (2020)
  19. Jove, E., Casteleiro-Roca, J.L., Quintián, H., Méndez-Pérez, J.A., Calvo-Rolle, J.L.: A new method for anomaly detection based on non-convex boundaries with random two-dimensional projections. *Information Fusion* **65**, 50–57 (2021)
  20. Jove, E., Gonzalez-Cava, J.M., Casteleiro-Roca, J.L., Quintián, H., Méndez Pérez, J.A., Vega Vega, R., Zayas-Gato, F., de Cos Juez, F.J., León, A., Martín, M., et al.: Hybrid intelligent model to predict the remifentanyl infusion rate in patients under general anesthesia. *Logic Journal of the IGPL* **29**(2), 193–206 (2021)
  21. Kozik, R., Pawlicki, M., Kula, S., Choraś, M.: Fake news detection platform—conceptual architecture and prototype. *Logic Journal of the IGPL* (2022)
  22. Leira, A., Jove, E., Gonzalez-Cava, J.M., Casteleiro-Roca, J.L., Quintián, H., Zayas-Gato, F., Álvarez, S.T., Simic, S., Méndez-Pérez, J.A., Luis Calvo-Rolle, J.: One-Class-Based Intelligent Classifier for Detecting Anomalous Situations During the Anesthetic Process. *Logic Journal of the IGPL* (11 2020)
  23. Luis Casteleiro-Roca, J., Quintián, H., Luis Calvo-Rolle, J., Méndez-Pérez, J.A., Javier Perez-Castelo, F., Corchado, E.: Lithium iron phosphate power cell fault detection system based on hybrid intelligent system. *Logic Journal of the IGPL* **28**(1), 71–82 (2020)
  24. Mokhtari, S., Abbaspour, A., Yen, K.K., Sargolzaei, A.: A machine learning approach for anomaly detection in industrial control systems based on measurement data. *Electronics* **10**(4), 407 (2021)
  25. Pachauri, G., Sharma, S.: Anomaly detection in medical wireless sensor networks using machine learning algorithms. *Procedia Computer Science* **70**, 325–333 (2015)
  26. Pawlowski, P., Urbaniak, R.: Logic of informal provability with truthvalues. *Logic Journal of the IGPL* (2022)
  27. Quintián, H., Jove, E., Casteleiro-Roca, J.L., Urda Muñoz, D., Arroyo Puente, Á., Calvo-Rolle, J.L., Herrero Cosío, Á., Corchado, E., et al.: Advanced visualization of intrusions in flows by means of beta-hebbian learning. *Logic Journal of the IGPL* (2022)
  28. Simić, S., Banković, Z., Villar, J.R., Simić, D., Simić, S.D.: A hybrid fuzzy clustering approach for diagnosing primary headache disorder. *Logic Journal of the IGPL* **29**(2), 220–235 (2021)
  29. Simić, S., Corchado, E., Simić, D., Đorđević, J., Simić, S.D.: A novel fuzzy metaheuristic approach in nurse rostering problem. *Logic Journal of the IGPL* **28**(4), 583–595 (2020)

30. Simić, S., Milutinović, D., Sekulić, S., Simić, D., Simić, S.D., Đorđević, J.: A hybrid case-based reasoning approach to detecting the optimal solution in nurse scheduling problem. *Logic Journal of the IGPL* **28**(2), 226–238 (2020)
31. Simić, S., Sakač, S., Banković, Z., Villar, J.R., Calvo-Rolle, J.L., Simić, S.D., Simić, D.: A three-stage hybrid clustering system for diagnosing children with primary headache disorder. *Logic Journal of the IGPL* (2022)
32. Zayas-Gato, F., Jove, E., Casteleiro-Roca, J.L., Quintián, H., Pérez-Castelo, F.J., Piñón-Pazos, A., Arce, E., Calvo-Rolle, J.L.: Intelligent model for active power prediction of a small wind turbine. *Logic Journal of the IGPL* (2022)
33. Zayas-Gato, F., Michelena, Á., Quintián, H., Jove, E., Casteleiro-Roca, J.L., Leitão, P., Luis Calvo-Rolle, J.: A novel method for anomaly detection using beta hebbian learning and principal component analysis. *Logic Journal of the IGPL* (2022)
34. Zhang, H., Gong, S.j., Dong, Z.z.: On-line parameter identification of induction motor based on rls algorithm. In: 2013 International Conference on Electrical Machines and Systems (ICEMS). pp. 2132–2137. IEEE (2013)

# Powerful Biogeography-Based Optimization algorithm with local search mechanism for Job Shop Scheduling Problem with additional constraints

Madiha Harrabi, Olfa Belkahla Driss and Khaled Ghedira

**Abstract** This paper proposes a Hybrid Biogeography-Based Optimization algorithm for solving the Job shop Scheduling Problem with additional constraints of Time Lags and transportation time using a Single Transport Robot to minimize the makespan (Completion time of the last operation executed). Biogeography-Based Optimization (BBO) algorithm is an evolutionary algorithm inspired by the migration of species between habitats. It has successfully solved optimization problems in many different domains and has demonstrated excellent performance. In order to improve the optimization efficiency of BBO algorithm, the Greedy constructive heuristic is used for population initialization to guarantee the diversity of solutions and the local search metaheuristic is used for the mutation step. The efficiency of the proposed algorithm is demonstrated by using new set of benchmarks for the problem. Numerical results show that the proposed Hybrid BBO algorithm not only significantly improves the performance of the standard BBO algorithm, but also finds competitive results compared with recently developed optimization methods.

## 1 Introduction

The Job shop Problem with Time Lags and Single Transport Robot (JSPTL-STR) is a special case of the classical job shop problem, it arises as a new sub-problem in a job

---

Madiha Harrabi

LARIA Laboratory, Ecole Nationale des Sciences de l'Informatique, Manouba Tunisia e-mail: madiha.harrabi@gmail.com

Olfa Belkahla Driss

LARIA Laboratory, Ecole Nationale des Sciences de l'Informatique, Manouba Tunisia e-mail: olfa.belkahla@isg.rnu.tn

Khaled Ghedira

LARIA Laboratory, Ecole Nationale des Sciences de l'Informatique, Manouba Tunisia e-mail: Khaled.Ghedir@isg.rnu.tn



shop environment where additional minimum and maximum Time Lags constraints between operations are considered and jobs have to be transported between machines by a single transport robot. The addition of time lag constraints between operations and transportation time of operations between machines makes difficult even the usually simple task of finding a feasible schedule. We study the Job Shop Scheduling Problem with Time Lags and Single Transport Robot which is a new extension of Job Shop Scheduling problem and consists of two sub-problems; the Job Shop Scheduling Problem with Time Lags and the Job Shop Scheduling Problem with Single Transport Robot.

Time lag means the waiting-time constraints between two consecutive operations in the same job or between two operations of different jobs. Minimum and maximum time lag constraints arise in many real-life scheduling applications. For example, in the steel industry, the time lag between the heating of a piece of steel and its molding should be small [31]. Similarly, when scheduling chemical reactions, the reactive usually cannot be stored for a long time between two stages of a process to avoid inter-actions with external elements. Many other industrial applications such as fabrication of printed circuits [20], hoist-scheduling problems [27], perishable product production [20] and in biotechnology and chemistry [28]. Different methods were proposed for solving the Job Shop Scheduling Problem with Time Lags in the literature. Caumont et al. [4], [5], and [6] introduced different metaheuristics such as tabu search algorithm, genetic algorithm and memetic algorithm. Deppner [8] proposed heuristics for a general scheduling problem which includes the job shop problem with minimal and maximal time lags between every pair of operations. Karoui et al. [21] investigated a Climbing Discrepancy Search method. Artigues et al. [2] proposed a job insertion heuristic and generalized resource constraint propagation mechanisms, González et al. [9] proposed a scatter search procedure combining the path relinking and the tabu search metaheuristic. Harrabi et al. [10] [11] [12] [13] [14] [15] [16] proposed a variety of metaheuristics, hybrid approaches, and distributed models using multi-agent system. Lacomme et al. [24] [25] proposed some dedicated constraint propagation and greedy randomized priority rules.

Different methods were proposed for solving the Job Shop Scheduling Problem with Single Transport Robot. Hurink and Knust [18] proposed a tabu search metaheuristic. Lacomme et al. [23] proposed a branch and bound procedure combined with a discrete events simulation model, Caumont et al. [7] proposed a mixed integer linear program then a heuristic branch and bound approach coupled with a discrete events simulation model. Nouri et al. [33] [34] proposed hybrid metaheuristics for solving Flexible Job shop Scheduling problem with transportation constraints. Afsar et al. [1] proposed a disjunctive graph modeling and a Greedy Randomized Adaptive Search Procedure with an Evolutionary Local Search procedure (GRASP  $\times$  ELSE) Harrabi et al. [17] proposed a metaheuristic hybridization for Job shop scheduling problem with time lags and transportation times. In this paper, we propose the resolution of the Job Shop Scheduling Problem with Time Lags and Single Transport Robot using a Hybrid Biogeography-Based Optimization algorithm (HBBO).

## 2 Job shop Scheduling Problem with Time Lags and Single Transport Robot: JSPTL-STR

The problem linear formulation of Job Shop Scheduling problem with Time Lags and Single Transport Robot is an extension of the Lacomme's formulation [23] extended by adding the transportation time constraints. The Job Shop Problem with minimum and maximum Time Lags and Single Transport Robot is a generalization of job shop problem, in which there are time constraints restricting the minimum and/or the maximum distance between two operations additionally jobs have to be transported between machines by a single transport robot. The JSPTL-STR involves a set of jobs that should be processed on a set of machines. Each job  $i$  consists of a sequence of operations;  $(i,j)$  denotes the  $j$ th operation of job  $i$ . Every operation must be assigned to a unique machine without interruption. For some pairs of operations  $(i,j)$  and  $(i',j')$  there are minimum and maximum time lag constraints respectively denoted by  $TL_{(i,j),(i',j')}^{min}$  and  $TL_{(i,j),(i',j')}^{max}$  restricting the distance between the end of  $(i,j)$  and the start of  $(i',j')$  to the interval  $[TL_{(i,j),(i',j')}^{min}, TL_{(i,j),(i',j')}^{max}]$ . Additionally, each job  $J_i$  ( $J_1, \dots, J_n$ ) is composed of  $n_{i-1}$  transport operations  $\{T_{i,1}, T_{i,2}, \dots, T_{i,n_{i-1}}\}$  to be made by a robot  $R$  from one machine to another. They occur if a job changes from one machine to another, i.e. if job  $J_j$  is processed on machine  $M_k$  and afterwards on machine  $M_l$ , a transportation time  $t_{jkl}$  arises. We assume that all transportations have to be done by a single transport robot  $R$  which can handle at most one job at a time. Solving the JSPTL-STR consists in sequencing all operations on the machines, such that the following constraints are satisfied: (i) Precedence constraints for operations of the same job; (ii) Minimum and Maximum Time Lag constraints; (iii) Each machine processes at most one operation at a time. (iiii) The robot transport at most one operation at a time. The objective is to find a schedule that minimizes the makespan which is the total completion time  $C_{max} = \max C_i$  where  $C_i$  is the finish time of job  $i$ .

## 3 Powerful Biogeography-Based Optimization with local search mechanism for Job shop Scheduling Problem with Time Lags and Single Transport Robot

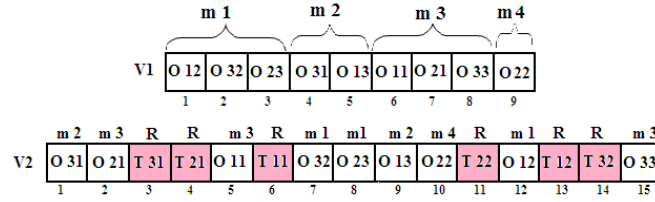
BBO algorithm, proposed by Simon in 2008 [30], is inspired by the mathematics of biogeography and mainly the work from MacArthur and Wilson [26]. Later, a large amount of theoretical, methodological, and practical studies on BBO have come into being. The two main concepts of BBO are habitat suitability index (HSI) and suitability index variables (SIVs). Considering the optimization algorithm, a population of candidate solutions can be represented as vectors. Each integer in the solution vector is considered to be a SIV. After assessing performance of the solutions, good solutions are considered to be habitats with a high HSI, and poor

ones are considered to be habitats with a low HSI. Therefore, HSI is analogous to fitness in other population-based optimization algorithms [32].

### 3.1 Habitat representation

The Job shop Scheduling Problem with Time Lags and Single Transport Robot is composed of two types of operations: the machine operations and the transport operations, that's why the representation is encoded in two vectors:

firstly, a vector V1 contain the machine operations sequence with length L1 equal to the total number of machine operations and where each index represents the selected operation to be processed on machine indicated at position p, see figure 1.



**Fig. 1** Habitat representation

For example,  $p = 4$ ,  $V1(4)$  is the selected operation O31 to be executed on machine m2. Secondly, a vector V2 contain the machine operations and transport operations sequence with length L2 equal to the total number of machine operations and transport operations and where each index represents the selected machine operation or transport operation indicated at position p. For example,  $p = 2$  and  $p = 6$ ,  $V2(2)$  is the selected operation O21 to be executed on machine m3 and  $V2(6)$  is the selected transport operation T11 to be transported by the robot R.

### 3.2 Initialization of population

The BBO algorithm starts with population habitats containing PS individuals, generated using Greedy constructive heuristic. This choice is based on high performance of the greedy algorithm which can produce solutions with good quality and its use often leads to better quality local optima. The greedy algorithm starts building the solution from one operation to another. After insertion the operation to a defined position of the current solution, the different constraints were checked. If all constraints are satisfied, we proceed to the next operation. Else, this operation was deleted in the current position and added to another position that respects the different constraints.

### 3.3 Migration operator

Migration is a probabilistic operator that is used for modifying each solution  $H_i$  by sharing features among different solutions. The idea of a migration operator is based on the migration in biogeography which shows the movement of species among different habitats. Solution  $H_i$  is selected as immigrating habitat with respect to its immigration rate  $\lambda_i$ , and solution  $H_j$  is selected as emigrating habitat with respect to its emigration rate  $\mu_j$ . It means that a solution is selected for immigrating or emigrating depends on its immigration rate  $\lambda_i$ , or emigration rate  $\mu_j$ ; the migration process can be shown as:

$$H_i(\text{SIV}) \leftarrow H_j(\text{SIV})$$

After calculating the HSI for each solution  $H_i$ , the immigration rate  $\lambda_i$  and the emigration rate  $\mu_j$  can be evaluated as follows:

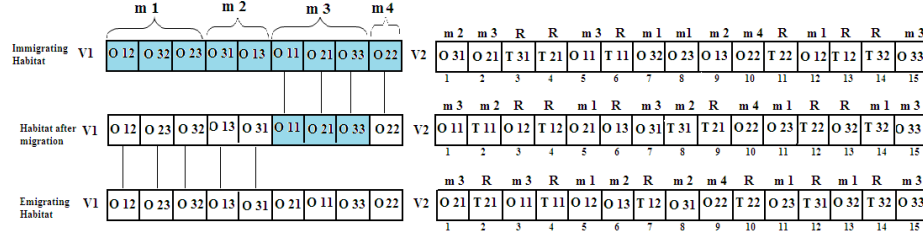
$$\lambda_i = I(1 - \frac{k_i}{n}) \quad (1)$$

$$\mu_j = E(\frac{k_j}{n}) \quad (2)$$

In (1) and (2),  $k_i$  represents the rank of the  $i$ th habitat after sorting all habitats according to their HSI. It is clear that since more HSI represents a better solution, more  $k_i$  represents the better solution. Therefore, the 1<sup>st</sup> solution is the worst and the  $n^{\text{th}}$  solution is the best.

$I$  is the maximum immigration rate and  $E$  the maximum emigration rate which are both usually set to 1,  $n$  is the number of habitats in the population. The two rates,  $\lambda_i$  and  $\mu_j$  are the functions of fitness or HSI of the solution. Since, according to the biogeography, the SIVs of a high-HSI solution tend to emigrate to low-HSI solutions, a high-HSI solution has a relatively high  $\mu_j$  and low  $\lambda_i$ , while in a poor solution; a relatively low  $\mu_j$  and a high  $\lambda_i$  are expected. Fig. 2 illustrates an example of migration operator of BBO for the Job Shop Scheduling Problem with Time Lags and Single Transport Robot.

As mentioned earlier, the SIVs from a good habitat tend to migrate into a poor habitat. This migration operator is performed probabilistically based on immigration and emigration rates. In this example, we will explain how the migration is implemented in our BBO algorithm. Consider dealing with an instance of Job Shop Scheduling Problem with Time Lags and Single Transport Robot presented in table 1. Suppose, based on immigration and emigration rates, that an immigrating habitat  $H_i = H_i =$



**Fig. 2** HMigration operator

$(O_{12}, O_{32}, O_{23}, O_{31}, O_{13}, O_{11}, O_{21}, O_{33}, O_{22})$  and an emigrating habitat  $H_e = (O_{12}, O_{23}, O_{32}, O_{13}, O_{31}, O_{21}, O_{11}, O_{33}, O_{22})$ . The migration process is:  
 $H_e(\text{SIV}) \leftarrow H_i(\text{SIV})$

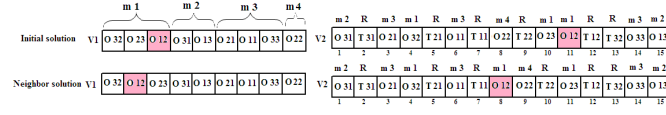
. Therefore, the migration process consists in:

- (1) SIVs of machine 3 and machine 4 from  $H_i$  ( $O_{11}, O_{21}, O_{33}, O_{22}$ ) migrate into  $H_e$  to replace SIVs of  $H_e$  ( $O_{21}, O_{11}, O_{33}, O_{22}$ ).
- (2) SIVs of  $H_i$  ( $O_{11}, O_{21}, O_{33}, O_{22}$ ) replace SIVs of  $H_e$  ( $O_{21}, O_{11}, O_{33}, O_{22}$ ).
- (3) SIVs ( $O_{12}, O_{23}, O_{32}, O_{13}, O_{31}$ ) of machine 1 and machine 2 from  $H_e$  remain at original places.
- (4) Therefore, the new habitat,  $H_n = (O_{12}, O_{23}, O_{32}, O_{13}, O_{31}, O_{11}, O_{21}, O_{33}, O_{22})$  is produced.

### 3.4 Mutation operator

Mutation is a probabilistic operator that randomly modifies a solution's SIV based on its priori probability of existence. Mutation is used to enhance the diversity of the population, which helps to decrease the chances of getting trapped in local optima. Solutions with very high HSI and very low HSI are both equally improbable, while medium HSI solutions are relatively probable to mutate. Namely, a randomly generated SIV replaces a selected SIV in the solution  $H_i$  according to a mutation probability. Note that an elitism approach is employed to save the features of the habitat that has the best solution in BBO process. The habitat with the best solution has a mutation rate of 0. In this step of the algorithm, we choose to introduce a local search algorithm in order to enhance the diversity of solutions. This step starts using the solution result of migration step as the initial solution then, applying the neighborhood structure to generate a modified solution. Figure 3 illustrates an example of mutation operator of BBO. We propose to use a local search procedure based on exchange moves neighborhood mechanism. This type of neighborhood is to swap the positions  $p_i$  and  $p_j$  of any two elements. This movement can generate good neighborhood solutions and more explore the solution space.

As mentioned earlier, the local search mutation mechanism is performed by replacing



**Fig. 3** Mutation operator

a selected SIV of a habitat with other generated SIV. (1) SIV  $O_{12}$  is chosen to mutate. (2) Assume that the new SIV which is randomly generated is  $O_{23}$ . SIV  $O_{12}$  is replaces with  $O_{23}$ . (3) SIV  $O_{23}$  takes the place of  $O_{12}$ . (4) The resulting mutated habitat is produced.

## 4 Experimental results

We study the Job shop Scheduling problem with Time Lags and Single Transport Robot and we propose a new data set of benchmarks. The studied problem is com-posed of two sub-problems; the job shop scheduling problem with time lags constraints and the job shop scheduling problem with single transport robot, for this rea-son the new benchmark data set is based on benchmark from the literature of these two sub-problems. In fact, we combine benchmark data set of Lawrence, 1984 [3] for job shop scheduling problem with generic time lags and data set of set of Hurink and Knust, 2005[19] for job shop scheduling problem with single transport robot. For instances of Lawrence (la01, la06, la11, la16 and la21), we add the full and empty moving transportation time of single robot between machines from Hurink and Knust data set [19]. In table 1 we give results of Hybrid Biogeography-Based Optimization algorithm used for solving Lawrence's instances of Job Shop Scheduling Problem with Time Lags and Single transport Robot. We give for each instances the name, size, results of CPLEX linear programming model and results of Biogeography-Based Optimization algorithm.

**Table 1** Makespan Results for Lawrence instances

Instances	Size	cplex	BBO	GBBO	IBBO
la01-D1-d1	10×15	923	1406	1364	1225
la01-D2-t0	10×15	962	1452	1328	1173
la01-D3-t1	10×15	937	1284	1017	948*
la01-D5-t2	10×15	954	1447	1381	1193
la01-T1-t1	10×15	977	1461	1291	1168
la01-T2-t1	10×15	825	1172	1053	834*
la01-T3-t2	10×15	813	1489	1267	1194
la01-T5-d1	10×15	822	1432	1294	1162
la06-D1-d1	15×15	1119	1903	1851	1725
la06-D2-t0	15×15	1183	1928	1834	1673
la06-D3-t1	15×15	1085	1893	1767	1615
la06-D5-t2	15×15	1251	1567	1369	1264*
la06-T1-t1	15×15	1283	1932	1814	1782
la06-T2-t1	15×15	1107	1917	1871	1743
la06-T3-t2	15×15	1358	1621	1439	1358*
la06-T5-d1	15×15	1191	1946	1772	1826
la11-D5-t2	20×5	1423	1693	1452	1431*
la11-T1-t1	20×5	1451	1954	1908	1896
la11-T2-t1	20×5	1381	1960	1937	1874
la11-T3-t2	20×5	1369	1647	1407	1396*
la11-T5-d1	20×5	1408	1961	1943	1827
la16-D5-d1	10×10	1325	1827	1706	1698
la16-D5-t1	10×10	1442	1574	1506	1442*
la16-D5-t2	10×10	1468	1843	1791	1648

We propose to study the Job Shop Scheduling Problem with Time Lags and Single Transport Robot. The CPLEX linear programming model of different instances was given in table 1 is considered as optimal solution. Compared with CPLEX optimal solutions, Hybrid Biogeography-Based Optimization algorithm gives results near to optimal solution in most of instances.

## 5 Conclusion

We propose a Hybrid Biogeography-Based Optimization algorithm for solving the Job Shop Scheduling Problem with Time Lags and Single Transport Robot. We propose a new set of benchmark instances inspired from benchmark instances of job shop scheduling problem with time lags and instances of job shop scheduling problem with single transport robot. According to an analysis and comparisons of the test results of HBBO algorithm through different instances of JSPTL-STR problem with optimal solutions of CPLEX, this algorithm gives good results near to the optimal ones. The proposed HBBO algorithm can be used to solve other extensions of our problem. We can develop the hybridization of BBO with other algorithms in

order to solve the same problem. We can also adopt the distributed BBO algorithm via the Multi-Agent System.

## References

1. H.M.Afsar, P.Lacomme, L.Ren, C.Prodhon, D.Vigo, "Resolution of a Job-Shop Problem with Transportation Constraints", A Master/slave Approach. IFAC Conference on Manufacturing Modelling, Management and Control, 2016.
2. C. Artigues, M. Huguet, and P. Lopez, "Generalized disjunctive constraint propagation for solving the job shop problem with time lags", Engineering Applications of Artificial Intelligence 24, 220–231, 2011.
3. S. Lawrence, S. Supplement to Resource Constrained Project Scheduling: An experimental investigation of Heuristic Scheduling Techniques. Graduate School of Industrial Administration, Carnegie Mellon University, 1984.
4. A. Caumont A, P. Lacomme, N. Tchernev, "Proposition d'un algorithme génétique pour le job-shop avec time-lags". ROADEF'05, 183-200, 2005.
5. A. Caumont, M. Gourgand, P. Lacomme and N. Tchernev, "Métaheuristiques pour le problème de job shop avec time lags", Jm|li,s j(i)|Cmax. 5ème conférence Francophone de Modélisation et SIMulation (MOSIM'04). Modélisation et simulation pour l'analyse et l'optimisation des systèmes industriels et logistiques, 939–946, Nantes, France, 2004.
6. A. Caumont, P. Lacomme, and N. Tchernev, "A memetic algorithm for the job-shop with time-lags", Computers Operations Research 35, 2331-2356, 2008.
7. A. Caumont, p. Lacomme, A. Moukrim, N. Tchernev, An MILP for scheduling problems in an FMS with one vehicle, European Journal of Operational Research, Vol. 199, No 3, 706-722, 2009.
8. F. Deppner, "Ordonnancement d'atelier avec contraintes temporelles entre opérations", PhD thesis, Institut National Polytechnique de Lorraine, 2004.
9. M.A. González, A. Oddi, R. Rasconi, R. Varela, "Scatter search with path relinking for the job shop with time lags and set up times", Computer Operation Research 60, 37-54, 2015.
10. M. Harrabi, O. Belkahla Driss, "MATS-JSTL: a Multi-Agent model based on Tabu Search for the Job Shop problem with Time Lags", International Computational Collective Intelligence Technologies and Applications ICCCI, 39-46, 2015.
11. M. Harrabi, O. Belkahla Driss and K. Ghedira, "Competitive Agents implementing parallel Tabu Searches for job shop scheduling problem with Time Lags", IASTED International conference on Modelling, Identification and Control: 848-052 MIC 2017.
12. M. Harrabi, O. Belkahla Driss and K. Ghedira, "Combining Genetic Algorithm and Tabu Search for job shop scheduling problem with Time Lags", IEEE International conference on engineering MIS 2017.
13. M. Harrabi, O. Belkahla Driss and K. Ghedira, "A Multi-Agent Model based on Hybrid Genetic Algorithm for Job Shop Scheduling problem with Generic Time Lags", ACS/IEEE International Conference on Computer Systems and Applications AICCSA 2017.
14. M. Harrabi, O. Belkahla Driss and K. Ghedira, "A Greedy Biogeography-Based Optimization Algorithm for Job Shop Scheduling Problem with Time Lags", Intelligence in Security for Information Systems CISIS 2018.
15. M. Harrabi, O. Belkahla Driss and K. Ghedira, "A Modified Biogeography-Based Optimization Algorithm with improved mutation operator for Job Shop Scheduling Problem with Time Lags", Logic Journal of the IGPL, (29)6: 951-962 <https://doi.org/10.1093/jigpal/jzaa037>, 2020.
16. M. Harrabi, O. Belkahla Driss and K. Ghedira, "A Hybrid Biogeography-Based Optimization for Job Shop Scheduling Problem with Generic Time Lags", Journal of Scheduling, (24): 329–346, <https://doi.org/10.1007/s10951-021-00683-w>, 2021.



17. M. Harrabi, O. Belkahla Driss and K. Ghedira, "Hybrid Biogeography-Based Optimization Algorithm for Job Shop Scheduling Problem with Time Lags and Single Transport Robot", *International Computational Collective Intelligence Technologies and Applications ICCCI* : 86-98, 2021.
18. Hurink, S. Knust, A tabu search algorithm for scheduling a single robot in a job-shop environment. *Discrete Applied Mathematics* 119(12), 181203, 2002.
19. J. Hurink, S. Knust, Tabu search algorithms for job-shop problems with a single transport robot. *European Journal of Operational Research*. 162(1), 99111, 2005
20. S.M. Johnson, Optimal two-and three-stage production schedules with setup times included. *Naval Research Logistics*, 1, 61-68, 1954.
21. W. Karoui, M.-J. Huguet, P. Lopez and M. Haouari, "Méthode de recherche à divergence limitée pour les problèmes d'ordonnancement avec contraintes de délais" [Limited discrepancy search for scheduling problems with time-lags]. 8ème ENIM IFAC Conférence Internationale de Modélisation et Simulation, Hammamet (Tunisie), 10-12, 2000.
22. Y.D. Kim, H.G. Lim and M.W. Park, Search heuristics for a flow shop scheduling problem in a printed circuit board assembly process. *European Journal of Operational Research*, 91, 124-143, 1996.
23. P. Lacomme, A. Moukrim, N. Tchernev, Simultaneous job input sequeing and vehicle dispatching in a single-vehicle automated guided vehicle system: a heuristic branch-and-bound approach coupled with a discrete events simulation model, *International Journal of Production Research*, 43:9, 1911-1942, 2005
24. P. Lacomme, MJ. Huguet and N. Tchernev, "Dedicated constraint propagation for Job-Shop problem with generic time-lags", 16th IEEE conference on Emerging Technologies and Factory Automation IEEE catalog number: CFP11ETF-USB, Toulouse, France, 2011.
25. P. Lacomme, N. Tchernev, "Job-Shop with generic Time Lags: a heuristic-based approach", 9th International Conference of Modeling, Optimization and Simulation – MOSIM, 2012.
26. R. MacArthur and E. Wilson, "The Theory of Biogeography", Princeton University Press, Princeton, NJ, USA, 1967.
27. M. A. Manier and C. Bloch, A classification of hoist scheduling problems. *International Journal of Flexible Manufacturing Systems*, 15(1), 37-55, 2003.
28. M. Nawaz, J.E.E. Enscore and I. Ham, A heuristic algorithm for the n-job, machine sequencing problem. *Management Science*, 16/B, 630-637, I. 1983.
29. B. Roy and B. Sussmann, Les problèmes d'ordonnancement avec contraintes disjonctives. Technical report, SEMA, 1964.
30. D. Simon, "Biogeography-based optimization", *IEEE Trans Evol Comput* 12:702–713, 2008.
31. A.D. Wismer, Solution of the Flow Shop scheduling Problem with no intermediate queues. *Operations Research* 20, 689-697, 1972.
32. Y. Yang, "A Modified Biogeography-Based Optimization for the Flexible Job Shop Scheduling Problem", *Mathematical Problems in Engineering*, 2015.
33. H-E. Nouri, O. Belkahla Driss and K. Ghedira, "Simultaneous scheduling of machines and transport robots in flexible job shop environment using hybrid metaheuristics based on clustered holonic multi-agent model", *Computers Industrial Engineering*, (102): 488-501, <https://doi.org/10.1016/j.cie.2016.02.024>, 2016.
34. H-E. Nouri, O. Belkahla Driss and K. Ghedira, "Controlling a Single Transport Robot in a Flexible Job Shop Environment by Hybrid Metaheuristics", *Transactions on Computational Collective Intelligence*, 18(1): 93-115, 2018.

# Dimensionality-Reduction Methods for the Analysis of Web Traffic

Nuño Basurto<sup>1</sup>, Álvaro Michelena<sup>2</sup>, Daniel Urda<sup>1</sup>, Hector Quintián<sup>2</sup>, José Luis Calvo-Rolle<sup>2</sup>, and Álvaro Herrero<sup>1</sup>

**Abstract** One of the usual targets for attackers are websites. Thus, protecting such assets is a key issue and consequently, a great effort has been devoted so far to address this problem. However, scant attention has been paid to investigate the contribution of unsupervised machine learning to the analysis of web traffic in order to detect attacks. To bridge this gap, the present paper proposes the novel application of dimensionality reduction methods to generate intuitive visualizations that can support the visual analysis of web traffic. More precisely, Laplacian Eigenmap, Isomap, t-Distributed Stochastic Neighbor Embedding, and Beta Hebbian Learning have been benchmarked. Promising results have been obtained on the standard CSIC2010 v2 dataset, encouraging further research on this topic.

**Keywords:** cybersecurity, web attacks, unsupervised learning, exploratory projection, visual analysis

## 1 Introduction and Previous Work

It is widely acknowledged that the Web is the main place to publicly deliver information at present time. As a result, such information and associated resources are

---

1. Nuño Basurto, Daniel Urda, Álvaro Herrero  
Grupo de Inteligencia Computacional Aplicada (GICAP), Departamento de Ingeniería Informática,  
Escuela Politécnica Superior, Universidad de Burgos, Av. Cantabria s/n, 09006, Burgos, Spain.  
e-mail: {nbasurto, durda, ahcosio}@ubu.es

2. Álvaro Michelena, Hector Quintián, José Luis Calvo-Rolle  
University of A Coruña, CTC, CITIC, Department of Industrial Engineering. Avda. 19 de febrero  
s/n, 15405, Ferrol, A Coruña, Spain.  
e-mail: {alvaro.michelena, hector.quintian, jlcalvo}@udc.es

highly exposed and became one of the main target for attackers. In this sense, many threatens and unknown vulnerability are exploited by them; by injecting malicious code and hence sending malformed requests with tainted payloads, as in the case of SQL [17], to damage websites or benefit from the access they may achieve to them [7]. Attackers started to focus on website long time ago [13] but this still is an open problem at present time [22]. According to the ENISA 2021 Threat Report [8], a plethora of threats and attacks are targeting availability and integrity. Among them, Distributed Denial of Service (DDoS) and Web Attacks stand out. Furthermore, considering data breaches in the information sector, basic web application attacks, errors and system intrusion are the main patterns, together accounting for 83% of all reported breaches.

Artificial Intelligence (AI) in general, and Machine Learning (ML) in particular area revealing in recent years as powerful tools to address difficult and unsolved problems in the cybersecurity domain, by analyzing traffic and/or access data generated by communications throughout time [16], such as intrusion detection [11]. Both supervised [19], [16] and unsupervised [6] methods of ML have been widely applied in the cybersecurity field up to now.

To validate ML models, there exists some well-known datasets, such as CSIC2010 [9] or CSE-CIC-IDS2018 [20], that have been employed to develop AI/ML-based solutions in order to detect a given outcome of interest. Particularly, the former dataset consists of thousands of web requests automatically generated at the “Information Security Institute” of the Spanish Research National Council for an e-commerce website application. The CSIC 2010 dataset has been selected in the present work as this is a standard benchmark dataset that many researchers have previously studied. On the one hand, supervised learning techniques have been used to try to accurately predict normal and anomalous traffic from this dataset. This includes recent works that validate innovative proposals. For instance, in [10] several well-known ML models were trained over features obtained from a unsupervised language representation model for embedding HTTP requests, and, more recently, a deep learning-based approach was presented in [2] showing how these kind of models can effectively predict the traffic of visiting websites. Furthermore, deep learning architectures (Convolutional Neural Networks, more precisely) have been also applied [19] for web attack detection based on this dataset.

On the other hand, unsupervised learning techniques have been previously proposed as useful tools to visualize cybersecurity datasets [12]. However, they have been very scarcely validated on the CSIC 2010 dataset. Up to the authors knowledge, the only previous work is [1], in which authors proposed generating an overview of HTTP traffic (original features of the dataset) to identify anomalous situations. In the present work, going one step further, the CSIC 2010 dataset is analyzed by more-advanced visualization models, namely: Laplacian Eigenmap, Isomap, t-Distributed Stochastic Neighbor Embedding, and Beta Hebbian Learning. Furthermore, the authors of this work consider that some the original features (such as the full URL request or the payload), contain rich information to improve the discrimination performed by ML models. As these features have not been intensively exploited before, authors additionally propose preprocessing the original features of this dataset to

extract new and relevant ones that may increase the visualization performance of ML models. Besides, the increase on the number of input variables is not a problem for the applied ML methods as they are conceived as dimensionality reduction techniques.

Therefore, this paper aims at benchmarking advanced ML models for dimensionality reduction by using new and richer CSIC2010 dataset instances. As a result, two new instances,  $D_1$  and  $D_2$ , are generated as a result of further preprocessing the payload feature within the original dataset ( $D_1$ ), and this one together with the full URL request field ( $D_2$ ). Both datasets are used to fed the unsupervised models in order to generate intuitive visualizations.

The rest of the paper is organized as follows. The applied ML methods are presented in Section 2. Next, the analyzed dataset (CSIC2010 dataset) as well as the extended features are presented in Section 3. Then, the results of the evaluation carried out in this work are presented and analyzed in Section 4. Finally, Section 5 contains some conclusions and future work.

## 2 Applied Methods

### 2.1 Laplacian Eigenmaps

Laplacian Eigenmaps [3] aim to preserve local properties when looking for a low dimensional representation of the data, being these the distances between nearest neighbors. After calculating the distance between a point and its  $k$  nearest neighbors, its low dimensionality is calculated for subsequent representation. The cost function works in such a way that the nearest neighbor has a greater weights than the second nearest neighbor and this one has a greater value than the next nearest neighbor and successively so forth.

For those points  $x_i$  and  $x_j$  found, being their representations  $y_i$  and  $y_j$ , their cost function is represented in equation 1. The distances between the original point  $x_i$  and  $x_j$  weights are represented by the weights  $\omega_{ij}$ . In this way, their representations in low dimensionality make a large contribution to the cost function, generating as consequence a great closeness in the depiction.

$$(\mathbf{Y}) = \sum_{ij} ||\mathbf{y}_i - \mathbf{y}_j||^2 \omega_{ij} \quad (1)$$

For the experimentation, the values used were 12 for the  $k$  nearest neighbors.

## 2.2 Isomap

Similar to Laplacian Eigenmaps, Isomap [21] also tries to construct a graph by establishing the distances between a point  $x_i$  and its  $k$  nearest neighbors, constructing a matrix with all the distances between all the points in the graph. In a second step, Isomap estimates the geodesic distances from the matrix values, taking into account only the closest path between points. Finally, in order to perform a dimensionality reduction on the data, a scaling of values is applied to the distance matrix, by embedding the data in a  $d$ -dimensional Euclidean space.

For a fair comparison, the value of 12 has been selected again for the  $k$  parameter.

## 2.3 t-Distributed Stochastic Neighbor Embedding

Unlike Laplacian Eigenmaps, which is a Sparse Spectral Technique and Isomap known as a main Weakness, the t-Distributed Stochastic Neighbor Embedding (t-SNE) [15] algorithm is a Non-convex technique, conceived a variation of the original Stochastic Neighbor Embedding [14]. The algorithm tries to reduce the number of points placed together on the map to show a better visualization of high dimensionality, helping in the search for possible clusters in the data. To do so, this algorithm first creates the probability distribution of a point  $x$  and its  $k$  nearest neighbors, then it generates a visualization of the distribution in low-dimensionality.

## 2.4 Beta Hebbian Learning

Among the range of applications of unsupervised artificial neural networks, data projection or visualization is the one that facilitates, human experts, the analysis of the internal structure of a dataset. This can be achieved by projecting data on a more informative axis or by generating maps that represent the inner structure of datasets. This kind of data visualization can usually be achieved with techniques such as Exploratory Projection Pursuit (EPP) [5, 4, 18] which project the data onto a low dimensional subspace, enabling the expert to search for structures through visual inspection.

In spite other previous EPP algorithm has obtained good results in this field, recently the novel Beta Hebbian Learning technique (BHL) [18] has reported a clear improvement in the results compared with the most used ones (PCA, MLHL, CMLHL, etc.) // The Beta Hebbian Learning technique (BHL) [18] is an Artificial Neural Network belonging to the family of unsupervised EPP, which uses Beta distribution as part of the weight update process, for the extraction of information from high dimensional datasets by projecting the data onto low dimensional (typically 2 dimensional) subspaces. This technique is better than other exploratory methods in that it provides a clear representation of the internal structure of data. // BHL

uses Beta distribution to update its learning rule to match the Probability Density Function (PDF) of the residual ( $e$ ) with the dataset distribution, where the residual is the difference between input and output feedback through the weights (5).

Thus, the optimal cost function can be obtained if the PDF of the residuals is known. Therefore, the residual ( $e$ ) can be expressed by 6 in terms of Beta distribution parameters ( $B(\alpha \text{ and } \beta)$ ):

$$p(e) = e^{\alpha-1}(1-e)^{\beta-1} = (x - Wy)^{\alpha-1}(1 - x + Wy)^{\beta-1} \quad (2)$$

where  $\alpha$  and  $\beta$  control the PDF shape of the Beta distribution,  $e$  is the residual,  $x$  are the inputs of the network,  $W$  is the weight matrix, and  $y$  is the output of the network. Finally, gradient descent can be used to maximize the likelihood of the weights (Eq. 3,):

$$\frac{\partial p_i}{\partial W_{ij}} = (e_j^{\alpha-2}(1-e_j)^{\beta-2}(-(\alpha-1)(1-e_j) + e_j(\beta-1))) = (e_j^{\alpha-2}(1-e_j)^{\beta-2}(1-\alpha+e_j(\alpha+\beta-2))) \quad (3)$$

Therefore, BHL architecture can be expressed by means the following equations:

$$\text{Feed-forward} : y_i = \sum_{j=1}^N W_{ij}x_j, \forall i \quad (4)$$

$$\text{Feedback} : e_j = x_j - \sum_{i=1}^M W_{ij}y_i \quad (5)$$

$$\text{Weightupdate} : \Delta W_{ij} = \eta(e_j^{\alpha-2}(1-e_j)^{\beta-2}(1-\alpha+e_j(\alpha+\beta-2)))y_i \quad (6)$$

where  $\eta$  is the learning rate

### 3 Dataset

A dataset developed by the Spanish Research National Council (CSIC) in 2010 was used in this study: the CSIC2010 v2 dataset. In concrete, it simulates attacks produced in HTTP queries of an e-Commerce application where users access and make purchases by using a shopping cart and providing different personal data. The complete and original dataset consists of  $N = 223,585$  samples, each one described by  $P = 18$  input variables and labeled as normal or anomalous traffic, which is considered the event of interest in this problem.

With the aim of employing AI/ML-based techniques which can help to understand the structure within this data, this original dataset was preprocessed first by grouping samples by individual sessions (i.e., forming a group of samples for those that have

the same value in the *cookie* feature). After this grouping procedure, the size of the dataset in terms of samples was reduced to  $N = 13,569$ . Nevertheless, some further preprocessing was carried out to the following variables to extract new possible relevant variables:

- **method:** it is a simple feature which describes three possible methods of the HTTP connections, which are “PUT”, “POST” and “GET”. This variables was simply processed by generated three dummy variables (i.e., binary variables that take a true or one value when that connection used the specified method by that variable, and false or zero in any other case).
- **payload:** this feature contains particular and useful data linked to the resource accessed by the user in the shape of “key=value” strings structure. All possible keys within the payload of all samples in the dataset were extracted by parsing the string structure and identifying all possible words before to “=” sign, resulting in a total of 19 keys (e.g., “ID”, “password” or “DNI”, among others). Therefore, 19 new dummy variables were generated indicating whether that session includes the key in the given payload or not. Besides, the values associated for each of the identified keys were also processed by computing the total length, in characters, of the given string and, consequently, adding a new extra feature which accounts for this information. Therefore, in total 38 new variables are added in this preprocessing step. Finally, two more variables were also included to the dataset: one that sums up the length of all the key’s values, and another one which counts the number of keys present in the given sample. For a better understanding of this payload variable, authors provide an example that could be found within samples of a given session:

modoA=insertar      precio=1764      B1=Pasar+por+caja

- **url:** this features contains information about the domain and resource accessed by the user. A first preprocessing step of this features analyzes and validates the full path of the *URL* (i.e., it should always start with “http://localhost:8080/” within this dataset), thus adding an extra logical feature that takes a true or one value when the full path does not start with that specific string. The remaining part of the full path is splitted by the different directories accessed to get the requested resource. For this purpose, the symbol “/” is employed as the split character in order to identify all possible directories that can be accessed in all samples of the dataset and, consequently, includes one extra feature which accounts for the number of directories accessed in a given session (it is a number between 0 and 4), and a second one that sums the total length of the corresponding directories’ name accessed. Finally, the file accessed within the last directory is processed by identifying the file extension in the given path (in total, 24 possible file extensions such as “.jsp”, “.gif” or “.jpg” were identified across the dataset). Consequently, 24 new dummy variables were added to the dataset to indicate the type of file accesed in the given session. Besides, one last feature was added in order to take into account the length of the filename accessed. For a better understanding of this url variable, authors provide an example that could be found within samples of a given session:

Table 1: Summary of the CSIC dataset version.

ID	Samples	Features	Normal Class	Anomaly Class
<b>Original</b>	223585	18	104000	119585
<b>Dataset 1 (<math>D_1</math>)</b>	13569	50	4303	9266
<b>Dataset 2 (<math>D_2</math>)</b>	13569	78	4303	9266

<http://localhost:8080/tienda1/publico/autenticar.jsp>

Overall, after applying the above-mentioned preprocessing steps two new dataset instances ( $D_1$  and  $D_2$ ) were generated from the original dataset, where their main characteristics are presented in Table 1. The difference between  $D_1$  and  $D_2$  is the set of input features that each of these datasets includes from the preprocessing step:  $D_1$  only includes features extracted from the *payload* variable, while  $D_2$  also adds on top the ones extracted from the *url* variable. With respect to the change of the class distribution in the original and generated dataset instances, the higher presence of anomaly samples could be explained due to the higher chance of finding anomalies in smaller and more isolated sessions in contrast to normal traffic.

## 4 Results

The results obtained by the methods described in section 2 are shown in the following subsections, grouped by the dataset. In each one of the visualizations (2D projection obtained by each one of the dimensionality reduction method) anomalous data instances (attacks) are depicted in red while normal instances are depicted in green.

### 4.1 Dataset 1

When analyzing the results of the  $D_1$  dataset, the first thing to take into account is that its dimensionality is lower than that of the second dataset. The representations are shown in Figure 1, in the first box **(a)**, is found the representation of Laplacian Eigenmaps, where the normal class is highly concentrated on the edge formed by the points of the anomalous class, despite the fact that it concentrates the points of both classes in excess. Regarding the Isomap representation **(b)**, a similar trend is observed, accumulating many points of both classes in the center, again isolating the normal class in the same area. It is in t-SNE**(c)** where a great difference is observed in the representation of the data with respect to what could be observed in the above figures, where an accumulation of the two classes is seen in several zones, highlighting only the upper middle zone where figures similar to circles are



formed where the anomalous class only manages to isolate itself. Finally, it is in BHL(d) where the greatest differentiation between classes is observed, isolating with a wide separation a large number of instances of the anomalous class. Near the cloud of instances of both classes found in the lower left zone there are again certain anomalous instances that manage to isolate themselves slightly.

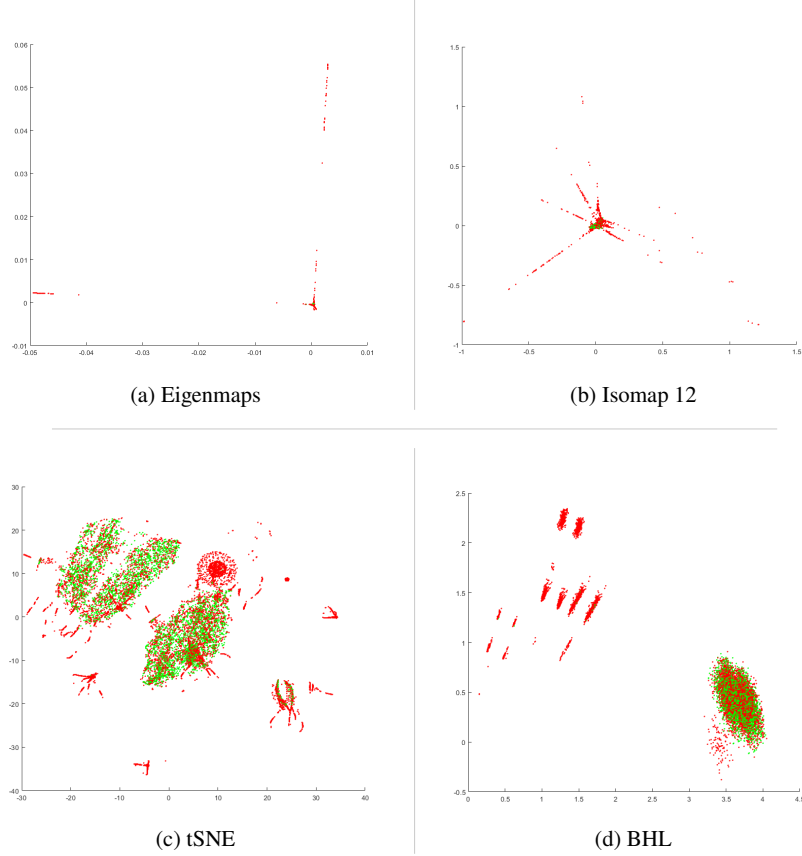


Fig. 1: 2D visualizations of the Dataset 1 generated by the different dimensionality-reduction methods. In red is represented the anomaly class, in green the normal class

## 4.2 Dataset 2

In the case of  $D_2$ , the results obtained by the different visualization techniques are shown in Figure 2. The Laplacian Eigenmap visualization (a) shows certain

differences when compared to that of the  $D_1$  dataset; a greater mixture of classes is observed along the line located on the  $y = 0$  coordinate. Hence, this visualization of the dataset has a smaller capability to differentiate among classes. Isomap (b) shows a similar distribution to the one previously seen, where the main difference is the normal class, which has been subdivided into two data groups. The distribution of class values remains very similar.

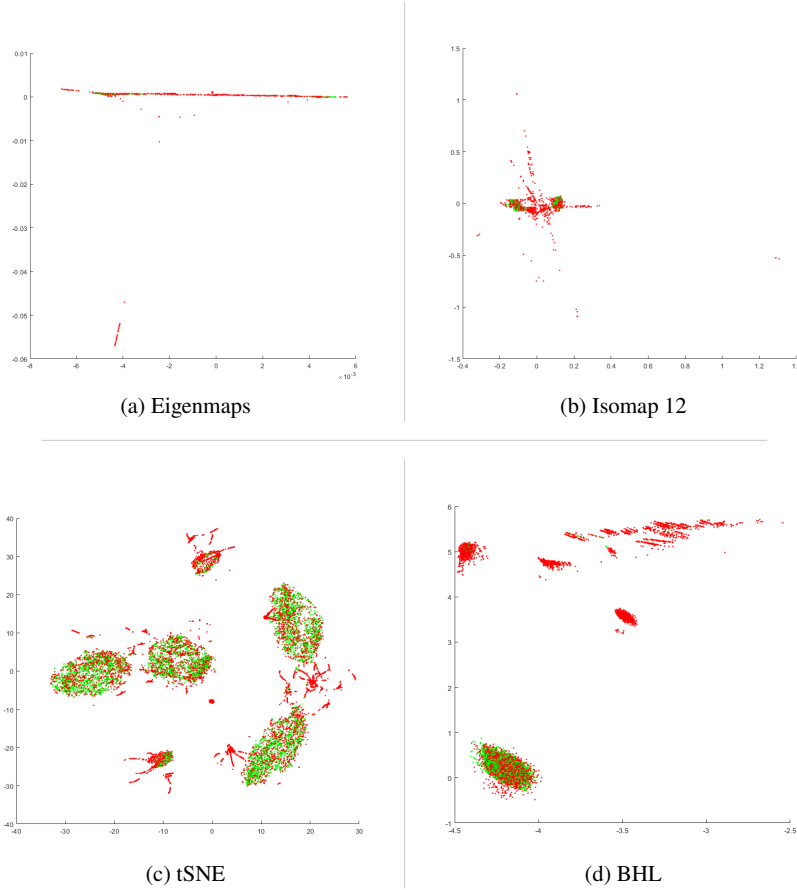


Fig. 2: 2D visualizations of the Dataset 2 generated by the different dimensionality-reduction methods..

As for t-SNE (c), a similar trend is observed, although at this time a new subgroup has been generated containing data from both classes. Despite finding some group of points of the anomalous class separated from the normal instances, it does not entail a significant improvement on the previous visualization of this same technique, grouping together many anomalous instances in a circular area. Finally, the BHL (d)

representation is again the most informative one, as a clear separation between the instances of the anomalous and normal classes can be seen. There is only one group, placed in the bottom-left corner of the figure, where the two classes overlap.

## 5 Conclusions and Future Work

Advanced ML models for dimensionality reduction are benchmarked on the well-known CSIC2010 dataset. Furthermore new and richer features extracted from the instances in this dataset have been also validated. Interesting visualizations, intuitively revealing the structure of the analyzed dataset have been obtained. More precisely, it is worth highlighting the visualizations obtained by BHL. These present the data in a clearer way that, although can not be associated with the normal/anomalous discrimination of instances, can greatly contribute to visually analyze web traffic. Regarding the extraction of information in order to generate new features of the data instances, it can be said that it does not contribute to better visualizations of the data.

The promising results that have been obtained encourage further research on the present topic; additional visualization methods as well as their combination with other unsupervised ML models will be studied for the analysis of web traffic.

**Acknowledgements** CITIC, as a Research Center of the University System of Galicia, is funded by Consellería de Educación, Universidade e Formación Profesional of the Xunta de Galicia through the European Regional Development Fund (ERDF) and the Secretaría Xeral de Universidades (Ref. ED431G 2019/01).

## References

1. Atienza, D., Herrero, Á., Corchado, E.: Neural analysis of http traffic for web attack detection. In: Á. Herrero, B. Barua, J. Sedano, H. Quintián, E. Corchado (eds.) International Joint Conference, pp. 201–212. Springer International Publishing, Cham (2015)
2. Bao, R., Zhang, K., Huang, J., Li, Y., Liu, W., Wang, L.: Research on website traffic prediction method based on deep learning. In: D. Jiang, H. Song (eds.) Simulation Tools and Techniques, pp. 432–440. Springer International Publishing, Cham (2022)
3. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* **15**(6), 1373–1396 (2003). DOI 10.1162/089976603321780317
4. Berro, A., Larabi Marie-Sainte, S., Ruiz-Gazen, A.: Genetic algorithms and particle swarm optimization for exploratory projection pursuit. *Ann. Math. Artif. Intell.* **60**, 153–178 (2010). DOI 10.1007/s10472-010-9211-0
5. Corchado, E., Fyfe, C.: Connectionist techniques for the identification and suppression of interfering underlying factors. *IJPRAI* **17**, 1447–1466 (2003). DOI 10.1142/S0218001403002915
6. Dutta, V., Pawlicki, M., Kozik, R., Choraś, M.: Unsupervised network traffic anomaly detection with deep autoencoders. *Logic Journal of the IGPL* (2022). DOI 10.1093/jigpal/jzac002. URL <https://doi.org/10.1093/jigpal/jzac002>. Jzac002
7. ENISA: ENISA Threat Landscape Report 2020. [Online; Accessed 9-June-2020] <https://bit.ly/3gdsB10>

8. ENISA: ENISA Threat Landscape Report 2021. [Online; Accessed 11-July-2022] <https://www.enisa.europa.eu/publications/enisa-threat-landscape-2021/@download/fullReport>
9. Giménez, C.T., Villegas, A.P., Álvarez Marañón, G.: HTTP DATASET CSIC 2010. [Online; Accessed 2-June-2022] <https://www.isi.csic.es/dataset/>
10. Gniewkowski, M., Maciejewski, H., Surmacz, T.R., Walentynowicz, W.: Http2vec: Embedding of HTTP requests for detection of anomalous traffic. *CoRR* **abs/2108.01763** (2021). URL <https://arxiv.org/abs/2108.01763>
11. Go, G.M., Bu, S.J., Cho, S.B.: Insider attack detection in database with deep metric neural network with Monte Carlo sampling. *Logic Journal of the IGPL* (2022). DOI 10.1093/jigpal/jzac007. URL <https://doi.org/10.1093/jigpal/jzac007>. Jzac007
12. Herrero, Á., Corchado, E., Sáiz, J.M.: Movocab-ids: Visual analysis of network traffic data streams for intrusion detection. In: E. Corchado, H. Yin, V. Botti, C. Fyfe (eds.) *Intelligent Data Engineering and Automated Learning – IDEAL 2006*, pp. 1424–1433. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
13. Kruegel, C., Vigna, G.: Anomaly detection of web-based attacks. In: *Proceedings of the 10th ACM Conference on Computer and Communications Security, CCS '03*, p. 251–261. Association for Computing Machinery, New York, NY, USA (2003). DOI 10.1145/948109.948144. URL <https://doi.org/10.1145/948109.948144>
14. van der Maaten, L., Hinton, G.: Stochastic neighbor embedding. *Advances in Neural Information Processing Systems* **15**, 833–840 (2002)
15. van der Maaten, L., Hinton, G.: Visualizing data using t-sne. *Journal of Machine Learning Research* **9**, 2579–2605 (2008)
16. Magán-Carrión, R., Urda, D., Diaz-Cano, I., Dorronsoro, B.: Towards a reliable comparison and evaluation of network intrusion detection systems based on machine learning approaches. *Applied Sciences* **10**(5) (2020). DOI 10.3390/app10051775
17. Pinzón, C., Herrero, Á., De Paz, J.F., Corchado, E., Bajo, J.: Cbrid4sql: A cbr intrusion detector for sql injection attacks. In: E. Corchado, M. Graña Romay, A. Manhaes Savio (eds.) *Hybrid Artificial Intelligence Systems*, pp. 510–519. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
18. Quintián, H., Corchado, E.: Beta hebbian learning as a new method for exploratory projection pursuit. *Int. J. Neural Syst.* **27**(6), 1–16 (2017). DOI 10.1142/S0129065717500241. URL <https://doi.org/10.1142/S0129065717500241>
19. Shahid, W.B., Aslam, B., Abbas, H., Khalid, S.B., Afzal, H.: An enhanced deep learning based framework for web attacks detection, mitigation and attacker profiling. *Journal of Network and Computer Applications* **198**, 103270 (2022). DOI <https://doi.org/10.1016/j.jnca.2021.103270>. URL <https://www.sciencedirect.com/science/article/pii/S1084804521002666>
20. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp* **1**, 108–116 (2018)
21. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000). DOI 10.1126/science.290.5500.2319. URL <https://www.science.org/doi/abs/10.1126/science.290.5500.2319>
22. de la Torre-Abaitua, G., Lago-Fernández, L.F., Arroyo, D.: On the application of compression-based metrics to identifying anomalous behaviour in web traffic. *Logic Journal of the IGPL* **28**(4), 546–557 (2020). DOI 10.1093/jigpal/jzz062. URL <https://doi.org/10.1093/jigpal/jzz062>

# **Special Session on Cybersecurity in Future Connected Societies**

# About the Fujisaki-Okamoto Transformation in the Code-based Algorithms of the NIST Post-Quantum Call

M.A. González de la Torre and L. Hernández Encinas

**Abstract** Post-quantum encryption schemes use variants of the Fujisaki-Okamoto transformation in order to construct a highly secure key encapsulation mechanism from a weakly secure public key encryption scheme. In the third round of the NIST post-quantum cryptography standardization call, all the candidates for the key encapsulation mechanism category use some of these transformations. This work studies how the mentioned transformations are applied in the code-based candidates of the NIST third round. These are Classic McEliece (finalist), BIKE (alternative) and HQC (alternative). Studying the differences between the transformations gives a better understanding of these candidates.

**Key words:** Post-quantum cryptography, Fujisaki-Okamoto transformation, Public key encryption, Key encapsulation mechanism, Code-based cryptography.

## 1 Introduction

The advance in quantum computing represents a threat for current encryption systems. Shor's algorithm [15] will allow quantum computers to break the public encryption schemes whose security is based on the integer factorization or discrete logarithm problems. This threat also affects to symmetric encryption for which the use of Grover's algorithm [9] implies that the key length should be twice the length that is used today in order to achieve the same security level.

Because of this landscape, the National Institute of Standards and Technology (NIST) started in 2016 a process [12] to determine new post-quantum cryptography (PQC) standards, i.e., sufficiently secure algorithms to resist attacks from quantum computers. This proposal has two different categories, one for public key encryption schemes and another for signature schemes. The need to meet a strong security

---

Institute of Physical and Information Technologies (ITEFI), Spanish National Research Council (CSIC), C/ Serrano 144, 28006-Madrid, Spain, e-mail: \{ma.gonzalez,luis.h.encinas\}@csic.es

definition requires a more complex structure than simply basing security on a mathematical problem; hence, to construct sufficient secure Public Key Encryption (PKE) algorithms hybrid encryption is commonly used. A hybrid scheme consist in the combination of a PKE and a Symmetric Key Encryption (SKE) scheme, if the algorithms considered are specifically designed to be part of a hybrid scheme, then are called Key Encapsulation Mechanism (KEM) and Data Encapsulation Mechanism (DEM) respectively. The scope of this work focuses on how sufficiently secure KEMs are designed, applying a transformation to an underlying PKE with a lower security requirement. The digital signature schemes do not need this kind of transformation, hence they are not considered in this study. Since the initial submission to the NIST process, there are now only nine candidates remaining, four as finalists and five as alternatives. Three of these algorithms base their security in error-correcting codes problems, those are the ones we will study in this contribution.

In 1999, the Fujisaki-Okamoto (FO) transformation was introduced [8]. This transformation consists on conforming a hybrid encryption scheme [6], using a less secure PKE to construct a much secure (hybrid) one. Nowadays, newer and improved versions of FO transformation are commonly used to reach strict and high levels of security. Dent [7] defined the KEM version of the FO transformation and Hofheinz et al. [10] gave a generalization of these transformations. In both works, some variants of the transformation have tight proofs of security in the Random Oracle Model (ROM). Moreover, Hofheinz et al. introduced a security proof in the Quantum Random Oracle Model (QROM). Later works [11, 14] studied the quantum security of all the transformations presented in [10]. Currently, all the candidates in the third round of NIST's call, in the PKE/KEM category, use some version of this transformation to give a proof of security.

The main goal of this work is to study and analyze the use of the different versions of the FO transformation in the NIST third round code-based finalist and alternative proposals. The rest of this paper is organized as follows. In Section 2 the theoretical background on code-based cryptography is introduced. Section 3 contains the definition of the different FO-like transformations and the security reduction of each one. In Section 4 we study how the different variants of the FO transformation are used in the finalist or alternative code-based algorithms in the PKE/KEM category of the NIST call. Finally, in section 5 we present our conclusions for this study.

## 2 Theoretical background

### 2.1 Security definitions

In this section we recall the main security notions that the studied algorithms take into consideration.

- *One-wayness* (OW): given an asymmetric encryption of a random plaintext and a public key, an attacker cannot find the corresponding plaintext. This is safe to use only if plaintexts are fully random. In the algorithms considered in this work

the precise security definition is *One Way Chosen Plaintext Attacks* (OW-CPA). We can extend this definition if the attackers have access to different types of oracles. If an attacker has access to a plaintext checking oracle, then the security is defined as *OW Plaintext Checking Attack* (OW-PCA). If there is a ciphertext validation oracle, then the security is called *OW Validation Attack* (OW-VA). If the attacker has access to both oracles described before, then the security is defined as *OW-Plaintext Checking Validation Attack* (OW-PCVA). When one of these security definitions is a model with access to quantum computations, then the notation OW-qCPA is used.

- *INDistinguishability under Chosen Plaintext Attack* (IND-CPA): given an asymmetric encryption of one of two attacker-chosen plaintexts and the public key, the attacker cannot determine which of the two messages was encrypted.
- *INDistinguishability under Adaptive Chosen Ciphertext Attack* (IND-CCA): the attacker fails under the conditions for IND-CPA even if additionally he can access to a decryption oracle. The oracle can be used on all ciphertexts but the challenge ciphertext. There are two definitions for IND-CCA security: IND-CCA1 and IND-CCA2 depending on when the attacker can make guesses to the decryption oracle. If he can do it only before receiving the asymmetric encryption of one of the messages the attacker is IND-CCA1; on the contrary, if the attacker can make queries also after receiving the ciphertext, then it is IND-CCA2. However, the more commonly used definition is IND-CCA2 and for the sake of simplicity the IND-CCA2 case is referred to as IND-CCA.
- *Disjoint Simulatability* (DS): Let  $D_M$  be a distribution over a message space  $M$ . We say that a Deterministic PKE (DPKE) scheme is  $D_M$  disjoint simulatable (for a probability distribution  $D_M$  over the message space) if the ciphertext of a message that is distributed according to  $D_M$  can be simulated by a simulator that does not know the message, and the simulated ciphertext is invalid (i.e., out of the range of an encryption algorithm) with overwhelming probability. This notion of security was introduced in [14].

When proving the security of a primitive,  $P$ , under the hardness of a problem,  $S$ , usually a reduction algorithm,  $R$ , is constructed which an adversary,  $A$ , uses as a subroutine against the security of  $P$  and solves the problem  $S$  (see. [14]).

Let  $(t, \epsilon)$  and  $(t', \epsilon')$  denote the running time and success rate of  $A$  and  $R$ , respectively. A reduction is tight if  $t \approx t'$  and  $\epsilon \approx \epsilon'$ . Tight security ensures that breaking the security of  $P$  implies breaking  $S$ . Conversely, if a security reduction is non-tight, it is not sure that  $P$  is hard to break even when  $S$  is (see an example in [11]). Also, a non-tight security proof affects the functionality of the schemes since the parameters need to be adjusted properly.

In this work, we consider that a Probabilistic PKE (PPKE) is a set  $\pi = \{\mathcal{G}', \mathcal{E}, \mathcal{D}, M, C\}$ , where  $\mathcal{G}'$ ,  $\mathcal{E}$ , and  $\mathcal{D}$  are the key generation, the encryption, and the decryption algorithms, respectively. The set  $M$  is the set of possible messages and  $C$  is an optional randomness set. If the PKE is DPKE, then  $C$  is not considered.  $M$  can be omitted, to reduce the notation, in case it is not necessary to specify it. Moreover, we denote by  $\kappa = \{\mathcal{G}, \mathcal{E}_c, \mathcal{D}_c\}$  a generic KEM, where  $\mathcal{G}$ ,  $\mathcal{E}_c$ , and  $\mathcal{D}_c$  are the key generation, the encapsulation, and the decapsulation algorithms, respectively.



The *correctness* of a PKE is related to the probability of generating invalid ciphertexts, i.e. ciphertexts generated by the encryption algorithm so that, if the decryption algorithm takes them as input, then it outputs error:  $\perp$ . A PKE  $\pi = \{\mathcal{G}', \mathcal{E}, \mathcal{D}, M\}$  is said to be *perfectly correct* if for any pair of public-private keys,  $(pk, sk)$ , generated by  $\mathcal{G}'$  and for any  $m \in M$ , is fulfilled  $\Pr[\mathcal{D}(sk, c) = m \text{ such that } c = \mathcal{E}(pk, m)] = 1$ .

A PKE is  $\gamma$ -spread if, for every  $(pk, sk) \leftarrow \mathcal{G}$  and every message  $m \leftarrow M$ ,  $\gamma(pk, m) = -\log \max_{c \in \text{Im}(\mathcal{E})} \Pr_{r \leftarrow \mathcal{R}C}[c = \mathcal{E}(pk, m, r)] \geq \gamma$ . This means that the image of  $\mathcal{E}$  is sufficiently random.

## 2.2 Code-based encryption

The theory of *error-correcting codes* is a mathematical speciality that allows to detect and to correct the possible errors produced when a message or any other information is transmitted. The errors can be due to noise, alteration, etc. This mathematical tool permits the receiver to recover the original message or information. That is, error-correcting codes take digital information and transform it so that the information can be recovered even if some of the bits have been erased or modified. This is done by adding redundant information, sometimes called *parity check*, by using matrix algebra over finite fields,  $\mathbb{F}_p$ .

The way the codification/decodification process works is that only certain bit strings (called codewords) are valid ciphertext. A distance defined between valid codewords in the code space allows the user to estimate if a received codeword is valid or might be corrected to a valid one. The most commonly used distance definition is the Hamming distance, i.e. the number of different bits when comparing two bit strings.

In this work, the plaintexts are presented as binary strings, the binary field is denoted as  $\mathbb{F}_2$ , and the cyclic polynomial ring defined for  $n \in \mathbb{N}$  is denoted as  $\mathcal{R} = \mathbb{F}_2[x]/(x^n - 1)$ .

## 3 Modern FO-like transformations

Since the publication of Fujisaki and Okamoto [8], new results have proposed optimizations and adaptations. Dent [7] was one of the first to propose the KEM constructions used today, with tight security proofs. However, the hypotheses that Dent considered in most of the transformations (the PKE should be deterministic and perfectly correct) are too strict for most quantum algorithms. These premises were also considered in the initial FO transformation [8] and in [5], where another version of the transformation called REAC/GEM is described.

In a latter work, Hofheinz et al. [10] studied modular PKE/KEM transformations that loosen some of the previous restrictions. They also presented the FO transformation as a composition of basic transformations, which allows to consider particular requirements in the applications as well as particular security reductions. Moreover, the authors also included a security proof in the QROM, giving the transformation relevancy in post-quantum cryptography.

The basic scheme of how to construct a KEM,  $\kappa = \{\mathcal{G}, \mathcal{E}_c, \mathcal{D}_c\}$ , from a DPKE,  $\pi = \{\mathcal{G}', \mathcal{E}, \mathcal{D}\}$ , has been known for a long time [7]: the key generation algorithm is the same for both, i.e.,  $\mathcal{G}' = \mathcal{G}$ . The encapsulation algorithm,  $\mathcal{E}_c$ , generates a random plaintext, encrypts it by using  $\mathcal{E}$ , and defines the shared secret (or session key), usually as the output of a hash function or a key derivation function of the chosen plaintext. The decapsulation algorithm,  $\mathcal{D}_c$ , receives the ciphertext, decrypts it by means of  $\mathcal{D}$  with the secret key, and generates the same session key. In this basic scheme, specific mid-steps are introduced to achieve strong and tight security reductions. This generic scheme is denoted as the U transformation.

### 3.1 Encrypt-with-Hash

As mentioned before, most of the initial KEM constructions are based on DPKE. The following transformation, denoted by T [7], transforms a PPKE into a deterministic one. Let  $G$  be a hash function and  $\pi$  a PPKE, then  $\pi^\tau = T[\pi, G] = \{\mathcal{G}^\tau, \mathcal{E}^\tau, \mathcal{D}^\tau\}$  is the following DPKE: The key generation algorithm is the same from  $\pi$ ,  $\mathcal{G}^\tau = \mathcal{G}$ . The encryption algorithm is  $\mathcal{E}^\tau(pk, m) = \mathcal{E}(pk, m, G(m)) = c$ , where  $G(m)$  is used as the random coins for  $\mathcal{E}$ . The decryption mechanism,  $\mathcal{D}^\tau$ , decrypts  $m' = \mathcal{D}(sk, c)$  and checks if  $m' = \mathcal{E}(pk, m', G(m'))$  —a re-encryption—; if the response is positive, it outputs  $m' = m$ ; otherwise, it rejects the message. Table 1 contains the security reductions of this transformation depending on the underlying security and which of them have a tight security reduction.

Security	QROM	Tightness	Requirements
T: OW-CPA $\Rightarrow$ OW-PCA	✓	–	none
T: IND-CPA $\Rightarrow$ OW-PCA	✓	✓	none
T: OW-CPA $\Rightarrow$ OW-PCVA	✓	–	$\gamma$ -spread
T: IND-CPA $\Rightarrow$ OW-PCVA	–	✓	$\gamma$ -spread

**Table 1** Security offered by the T transformation [10]

### 3.2 Implicit/Explicit Rejection

The schemes presented by Dent in [7] are constructed with “explicit rejection” (notation  $\perp$ ). This refers to how the KEM deals with either errors in the encryption/decryption process of the PKE or invalid ciphertexts. Let us consider the case in which the decapsulation algorithm receives a ciphertext  $c$  and either is invalid or after running the decryption algorithm it outputs  $\perp$ . The design with “explicit rejection” means that, in this situation, the decapsulation outputs  $\perp$ . The other design variant is called “implicit rejection” (notation  $\perp$ ), in which, in the same situation as before, the decapsulation process outputs  $H(s, c)$ . This output is a hash (depending on the design a pseudorandom function may be used) that takes as input a randomly chosen string  $s$  (normally chosen from the message space) and the ciphertext  $c$ . The string  $s$  is generated during the key generation algorithm and saved as part of the secret key. Transformations with “implicit rejection” are mostly used in the NIST algorithms,

since the security that they provide is usually stronger. However, it is important to be aware that not only the secret key is longer in these schemes, also the use of implicit rejection supposes introducing modifications in the key generation algorithm.

### 3.3 Definition of the shared secret

In [10] a classification of the U transformation is introduced based on the definition of the shared secret. As explained before, the shared secret is defined using a randomly chosen plaintext,  $m$ , and a hash function  $H$ . The transformations with shared secret,  $K = H(m)$ , use the subindex  $m$  in their notation. In these cases, it is required that the underlying PKE is deterministic. The other definition for the shared secret uses the encryption of the randomly chosen plaintext,  $\mathcal{E}(pk, m) = c$ , and then the shared secret is defined as  $K = H(m, c)$ . In this case no subindex  $m$  is used and there is no requirement on the PKE.

### 3.4 Additional hash

The use of additional hash is one of the first proposed ways to obtain quantum security, was initially defined in [7], although the security proof in the QROM was given latter in [10]. The use of an additional hash works as follows: during the encapsulation a value  $d = G(m)$ , that is also part of the output, is defined. The decapsulation takes as input  $(c, d)$ , where  $c$  is the ciphertext, and once  $m' = \mathcal{D}(sk, c)$  is obtained, it checks if  $c = \mathcal{E}(pk, m')$  and if  $d = G(m')$ . The use of an additional hash is denoted by adding the letter Q in the notation of the transformation.

Table 2 shows the different transformations depending on the techniques or design decisions defined before. The construction of the FO transformation, as in [10], requires to compose the T transformation with a version of the U KEM construction.

Transformation	$\pi$ Secur. ROM	$\pi$ Secur. QROM	Tight. ROM	Sec.proof QROM	DPKE	Perf. Cor.
$\text{KEM}^\perp = \text{U}^\perp[\pi, H]$	OW-PCA	OW-qPCA	[10]	[11]	N	N
$\text{KEM}^\perp = \text{U}^\perp[\pi, H]$	OW-PCVA	OW-qPVCA	[10]	[11]	N	N
$\text{KEM}_m^\perp = \text{U}_m^\perp[\pi, H]$	OW-CPA	OW-CPA, DS	[10]	[14] [11]	Y	N[11] Y[14]
$\text{KEM}_m^\perp = \text{U}_m^\perp[\pi, H]$	OW-VA	OW-VA	[10]	[11]	Y	N[11]
$\text{QKEM}_m^\perp = \text{QU}_m^\perp[\pi, H, H']$	OW-PCA, OW-CPA	OW-CPA	[10] [7]	[10]	N[10] Y[7]	N[10] Y[7]
$\text{QKEM}_m^\perp = \text{QU}_m^\perp[\pi, H, H']$	OW-PCA	OW-CPA	[10]	[10]	N	N

**Table 2** Transformations and the security proofs sources and requirements

## 4 FO transformation application in code-based algorithms

All the finalist and alternative post-quantum encryption schemes of the NIST call in the PKE/KEM chapter use the FO transformation. In what follows, we will study how this transformation and some modifications of it are used in the code-based proposals.

#### 4.1 Classic McEliece

Classic McEliece [2] is the only code-based algorithm among the finalists of the NIST PQC call, whereas both BIKE [3] and HQC [1] are part of the alternatives. Classic McEliece is the evolution of a well-known code-based encryption scheme originally introduced by McEliece, with proper adaptations to face quantum attacks.

Following the notation from §3, the transformation applied in Classic McEliece is  $QU^\perp$ , indicating that the transformation  $U$  is used with implicit rejection and additional hash; the shared secret is defined as  $H(1, m, c)$ . Since McEliece PKE [2, §2.2] is deterministic, there is no need to apply the transformation  $T$ . This PKE is also perfectly correct.

Classic McEliece KEM's security relies on the results by Dent [7, §6] and by Saito et al. [13, Th 5.2]. Both publications require the underlying PKE to be perfectly correct and deterministic. Dent's results present tight security proofs in the ROM, while Saito et al.'s work introduces close to tight security reductions to certain transformations in the QROM.

The IND-CCA security of Classic McEliece tightly reduces to the OW-PCA security of the underlying PKE in the ROM [7]. The transformation is modified with the intention of applying both Dent's and Saito et al.'s results, which, originally, take into consideration different KEM constructions. The NIST submission for Classic McEliece states that a security reduction in the QROM for exactly the transformation applied in Table 3 is not proven today.

Table 3 shows the way Classic McEliece KEM operates. The notation used for the hash inputs is based in [2] and the numbers refer to byte entries.

$\mathcal{G}$	$\mathcal{E}c(pk)$	$\mathcal{D}c(c, sk)$
$(pk, sk') \leftarrow \mathcal{G}'$ $s \leftarrow_R \mathbb{F}_2^n$ $sk = (sk', s)$ <b>return</b> $(pk, sk)$	$m \leftarrow_R \mathbb{F}_2^n$ $c_0 \leftarrow \mathcal{E}(pk, m)$ $c_1 \leftarrow H(2, m)$ $c = (c_0    c_1)$ $K \leftarrow H(1, m, c)$ <b>return</b> $(K, c)$	Parse $c = (c_0    c_1)$ $b \leftarrow 1$ $e' \leftarrow \mathcal{D}(sk', c_0)$ <b>if</b> $e' = \perp, m' \leftarrow s, b \leftarrow 0$ $c'_1 \leftarrow H(2, m')$ <b>if</b> $c'_1 \neq c_1, m' \leftarrow s, b \leftarrow 0$ $K \leftarrow H(b, m', c)$ <b>return</b> $K$

**Table 3** McEliece: key encapsulation mechanism

#### 4.2 BIKE

BIKE (*Bit flipping Key Encapsulation*) is a KEM based on Quasi-Cyclic Moderate Density Parity-Check (QC-MDPC) codes [3], which uses  $FO^\perp$  transformation to form an IND-CCA KEM from a  $\delta$ -correct IND-CPA secure PKE [10].

Based on the hardness of the underlying code problems, BIKE is IND-CPA secure. This can be achieved choosing a correct set of parameters [3]. The algorithm reaches IND-CCA security depending on the  $\delta$ -correctness of the decoder and a sufficiently secure decoder is presented.

Moreover, BIKE uses ephemeral keys, i.e., it considers a fresh public/private key pair for every key exchange session, for which an IND-CPA security is sufficient (models using the same key pair more than once require IND-CCA security).

Table 4 presents the key generation, the encapsulation and the decapsulation algorithms for the BIKE proposal. An interesting modification of the transformation introduced in this algorithm is that the decapsulation checks the randomness instead of the encryption. Also, since in BIKE's submission the encryption scheme is not defined, Table 4 also contains the encryption/decryption process. The set  $\mathcal{H}_\omega$  is defined as  $\mathcal{H}_\omega = \{(h_0, h_1) \in \mathcal{R} \times \mathcal{R} : |h_0| = |h_1| = w/2\}$ , where  $w$  is one of the parameters [3].

$\mathcal{G}$	$\mathcal{Ec}(pk)$	$\mathcal{Dc}(c, sk)$
$(h_0, h_1) \leftarrow_R \mathcal{H}_\omega$	$m \leftarrow_R \mathbb{F}_2^l$	Parse $c = (c_0 \  c_1)$
$h \leftarrow h_1 h_0^{-1}$	$(e_0, e_1) \leftarrow G(m)$	$e' \leftarrow \text{decoder}(c_0 h_0, h_0, h_1)$
$s \leftarrow_R \mathbb{F}_2^l$	$c \leftarrow (e_0 + e_1 h, m \oplus H_1(e_0, e_1))$	$m' \leftarrow c_1 \oplus H_1(e')$
$sk = (h_0, h_1, s)$	$K \leftarrow H_2(m, c)$	<b>if</b> $e' = G(m')$
$pk = h$	<b>return</b> $(c, K)$	<b>return</b> $K \leftarrow H_2(m', c)$
<b>return</b> $(pk, sk)$		<b>else return</b> $K \leftarrow H_2(s, c)$

**Table 4** BIKE: key encapsulation mechanism

### 4.3 HQC

HQC (*Hamming Quasi-Cyclic*) is an efficient encryption scheme also based on coding theory, whose PKE associated is IND-CPA secure [1, Th.5.1]. HQC applies the  $\text{QFO}^\perp$  transformation and it is stated that it reaches IND-CCA security in the ROM [10]. However, in [10], security proofs are presented only for the transformations  $\text{QFO}_m^\perp$  and  $\text{QFO}_m^L$ . HQC submission [1] also claims that based on new publications, like [4, 10, 11, 14] the transformation  $\text{FO}^\perp$  could be applied to make the proposal IND-CCA secure in the QROM. Because, when the HQC was presented, these transformation results were considerably novel, they are mentioned in [1] only as a possible best practice.

Table 5 shows the encapsulation and decapsulation algorithms of the HQC proposal, where it can be appreciated how the  $\text{QFO}^\perp$  is applied. Since the transformation  $\text{QFO}^\perp$  uses “explicit rejection”, the key generation algorithm is the same for HQC KEM and HQC PKE [1].

$\mathcal{Ec}(pk)$	$\mathcal{Dc}(sk, c, d)$
$m' \leftarrow \mathbb{F}_2^k$	$m' \leftarrow \mathcal{D}(sk, c)$
$r \leftarrow G(m')$	$r' \leftarrow G(m')$
$c \leftarrow (u, v) = \mathcal{E}(pk, m, r)$	$c' \leftarrow \mathcal{E}(pk, m', r')$
$K \leftarrow H_1(m, c)$	<b>if</b> $c \neq c'$ or $d \neq H_2(m')$
$d \leftarrow H_2(m)$	<b>return</b> $\perp$
<b>return</b> $(c, d, K)$	<b>else return</b> $K \leftarrow H_1(m, c)$

**Table 5** HQC: key encapsulation mechanism

## 5 Conclusions

In this work we have analyzed the variants of the FO transformation applied to the NIST PQC code-based finalists and alternatives. It is remarkable the fact that each algorithm applies a different transformation. The  $FO^\perp$  variant is the one that most NIST finalist or alternatives apply (considering all the proposals, not only code-based). Among the code-based proposals, the  $FO^\perp$  is used in BIKE and mentioned in the HQC submission as an alternative transformation. The reason of this prevalence might be because this is the only transformation that has a tight security proof in the ROM (without additional hash) in [10].

One of the most surprising design decisions about the algorithms studied is the transformation chosen for HQC,  $QFO^\perp$ , which was not studied in [10]. Additionally, this is the only algorithm that uses a transformation with “explicit rejection”. In [4] (and in other sources) it is stated that “implicit rejection” gives a better security reduction. As mentioned in §3, “implicit rejection” implies some design decisions that may affect the functionality, however it seems like the predominant practice.

The transformation applied in the Classic McEliece submission,  $QU^\perp$ , is a good example of how relevant the characteristics of the underlying PKE are. McEliece’s PKE is deterministic and perfectly correct, which allows to apply Dent’s results that provide a tight security proof in the ROM. In the QROM, Saito et al. provide a close to tight security proof for the transformation  $U_m^\perp$ , which is not the same transformation applied to McEliece’s algorithm, although both are similar, in some way. It is reasonable to ask ourselves why this transformation was not used instead. In [10] it is proven that  $U_m^\perp$  tightly reduces IND-CCA security to OW-CPA in the ROM, and in [14] Saito et al. provide the security reduction in the QROM (from DS to IND-CCA in the QROM the reduction is very close to tight).

A possible reason for not choosing the  $U_m^\perp$  transformation might be that Dent’s results have been studied for a long time, while the more modern results, like the one from Saito et al. use new security definitions (DS) that are less known. As explained before, in the McEliece submission it is stated that there is not a concrete proof of security in the QROM for the transformation applied, however due to the similarities, the performance and the active research in the area, the designers are confident that this will not be an issue.

**Acknowledgements** This work was supported in part by project P2QProMeTe (PID2020-112586RB-I00/AEI/10.13039/501100011033), ORACLE Project, with reference PCI2020-120691-2, funded by MCIN/AEI/10.13039/501100011033 and European Union “NextGenerationEU/PRTR”, in part by the Spanish State Research Agency (AEI) of the Ministry of Science and Innovation (MCIN), and in part by the EU Horizon 2020 research and innovation programme, project SPIRS (Grant Agreement No. 952622).

## References

1. Aguilar Melchor, C., Aragon, N., Bettaieb, S., Bidoux, L., Blazy, O., Deneuville, J.C., Gaborit, P., Persichetti, E., Zémor, G., Bos, J.: HQC (Hamming Quasi-Cyclic). Online publication

- (2021), <https://pqc-hqc.org/>
2. Albrecht, M.R., Bernstein, D.J., Chou, T., Cid, C., Gilcher, J., Lange, T., an Ingo von Maurich, V.M., Misoczki, R., Niederhagen, R., Paterson, K.G., Persichetti, E., Peters, C., Schwabe, P., Sendrier, N., Szefer, J., Tjhai, C.J., Tomlinson, M., Wang, W.: Classic McEliece: conservative code-based cryptography (2020), <https://classic.mceliece.org/nist.html>
  3. Aragon, N., Barreto, P., Bettaieb, S., Bidoux, L., Blazy, O., Deneuville, J.C., Gaborit, P., Gueron, S., Güneysu, T., Aguilar Melchor, C., Misoczki, R., Persichetti, E., Sendrier, N., Tillich, J.P., Vasseur, V., Zemor, G.: BIKE (Bit Flipping Key Encapsulation). Online publication (2021), <https://bikesuite.org>
  4. Bernstein, D.J., Persichetti, E.: Towards KEM unification. Cryptology ePrint Archive, Report 2018/526 (2018), <https://eprint.iacr.org/2018/526>
  5. Coron, J.S., Handschih, H., Joye, M., Pailier, P., Pointcheval, D., Tymen, C.: GEM: a generic chosen-ciphertext secure encryption method. In: Proc. Topics in Cryptology - CT-RSA 2002, Lecture Notes Comput. Sci. vol. 2271, pp. 263–276 (2002), [https://doi.org/10.1007/3-540-45760-7\\_18](https://doi.org/10.1007/3-540-45760-7_18)
  6. Cramer, R., Shoup, V.: Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. Cryptology ePrint Archive, Report 2001-108 (2001), <http://eprint.iacr.org/2001/108>
  7. Dent, A.W.: A designer's guide to KEMs. In: Proc. 9th IMA International Conference on Cryptography and Coding, Lecture Notes in Computer Science. vol. 2898 (2003), [https://doi.org/10.1007/978-3-540-40974-8\\_12](https://doi.org/10.1007/978-3-540-40974-8_12)
  8. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. In: Proc. 19th Annual International Cryptology Conference, Advances in Cryptology - CRYPTO'99, Lecture Notes Comput. Sci. vol. 1666, pp. 537–554 (1999), [https://doi.org/10.1007/3-540-48405-1\\_34](https://doi.org/10.1007/3-540-48405-1_34)
  9. Grover, L.K.: Quantum mechanics helps in searching for a needle in a haystack. Physical Review Letters **79**(2), 325–328 (1997), <https://doi.org/10.1103/PhysRevLett.79.325>
  10. Hofheinz, D., Hövelmanns, K., Kiltz, E.: A modular analysis of the Fujisaki-Okamoto transformation. In: Proc. 15th International Conference Theory of Cryptography TCC'2017, Lecture Notes Comput. Sci. vol. 10677, pp. 341–371 (2017), [https://doi.org/10.1007/978-3-319-70500-2\\_12](https://doi.org/10.1007/978-3-319-70500-2_12)
  11. Jiang, H., Zhang, Z., Chen, L., Wang, H., Ma, Z.: IND-CCA-secure key encapsulation mechanism in the quantum random oracle model, revisited. Cryptology ePrint Archive, Report 2017-1096 (2017), <http://eprint.iacr.org/2017/1096>
  12. NIST: Post-quantum cryptography. On-line publication (2016), <https://csrc.nist.gov/Projects/Post-Quantum-Cryptography>
  13. Saito, T., Xagawa, K., Yamakawa, T.: Tightly-secure key-encapsulation mechanism in the quantum random oracle model. Cryptology ePrint Archive, Report 2017-1005 (2017), <http://eprint.iacr.org/2017/1005>
  14. Saito, T., Xagawa, K., Yamakawa, T.: Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In: Proc. Annual International Conference on the Theory and Applications of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT 2000, Lecture Notes Comput. Sci. vol. 10822, pp. 520–551 (2000), [https://doi.org/10.1007/978-3-319-78372-7\\_17](https://doi.org/10.1007/978-3-319-78372-7_17)
  15. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Review **41**(2), 303–332 (1999), <https://doi.org/10.1137/S0036144598347011>

# Analysis of Secret Key Agreement Protocol for Massive MIMO Systems

Seiya Otsuka, Hiroki Iimori, Kengo Ando,  
Giuseppe Thadeu Freitas de Abreu, Koji Ishibashi and Naoki Ishikawa

**Abstract** In this paper, we propose and analyze a secret key agreement (SKA) protocol designed for multiple-input multiple-output (MIMO) scenarios, in which a legitimate transmitter estimates the amount of information that can be leaked to potentially hazardous legitimate receivers in the network, generates a secret key, and transmits the generated key to a specific legitimate receiver. For the latter key transmission step, a new beamforming method is proposed that regards other legitimate receivers in the network as untrustworthy nodes and show the attack algorithm they can perform. The proposed protocol requires only one-way transmission and no mutual interaction to share the secret key. Through numerical simulations, we evaluate two scenarios for malicious users, where a part of them are other legitimate receivers. Namely, we evaluate the information leakage risk of our model under two conditions: a disadvantage for them, where the malicious user is unaware of the existence of beamforming, and an advantage, where the user can jam directly to the beamformer. Finally, to demonstrate the practicality of the protocol, we demonstrate that the beamforming advantage dramatically reduces the computational complexity requirement at the legitimate receiver.

**Key words:** massive MIMO, physical layer security, secret key agreement, beamforming.

---

Seiya Otsuka, Naoki Ishikawa  
Graduate School of Engineering Science, Yokohama National University, 240-8501 Kanagawa, Japan, e-mail: [otsuka-seiya-zz@ynu.jp](mailto:otsuka-seiya-zz@ynu.jp), [ishikawa-naoki-fr@ynu.ac.jp](mailto:ishikawa-naoki-fr@ynu.ac.jp)

Hiroki Iimori, Giuseppe Thadeu Freitas de Abreu  
Department of Electrical and Computer Engineering, Jacobs University Bremen, 28759 Bremen, Germany

Kengo Ando, Koji Ishibashi  
Advanced Wireless and Communication Research Center, The University of Electro-Communications, 182-8585 Tokyo, Japan



## 1 Introduction

Wireless communications technology has been an indispensable infrastructure of the modern society over the last decades. Owing both to the increasing reliance onto wireless systems for the transmission of public and private information, and to the rise of data privacy concerns which is expected to attract more attention hereafter, secure wireless communications will be a strict requirement for the future wireless network systems.

In spite of this growing demand, the rapid advancement of quantum computing may threaten the security of Rivest-Shamir-Adleman (RSA) cryptography [1] relying on the complexity of factoring the product of two prime numbers, which is known to be vulnerable to quantum methods such as Shor's algorithm [2]. In view of the latter, obsolete security protocols including those currently employed in wireless standards need be amended so as to offer truly secure communications, and consequently reliable and resilient applications.

A key enabling technology to that end is physical layer security (PLS), which aims to enhance security features of communications systems by exploiting physical characteristics of wireless channels [3]. PLS can counter information leakage by adapting multiple-input multiple-output (MIMO) method, an important technology in wireless communications, to utilize a precoding design. In particular, if a transmitter can instantly obtain the channel state information (CSI) of unauthorized user(s) (eavesdropper), it can separate data exchange between desired users from eavesdroppers [4].

Apart from secure waveform design based on PLS, however, secret key agreement (SKA) protocols are also necessary to establish properly secure communications between the transmitter and the desired receiver. A pioneering work in that area is Maurer's protocol for noisy public channels in the presence of a passive eavesdropper [5], which comprises of two phases: information reconciliation and privacy amplification. During the information reconciliation phase, legitimate users match a random sequence via the public channel, whereas the goal of the privacy amplification phase is to increase confidentiality by extracting a key from the agreed-upon random sequence. Here, to generate a secure key, it is necessary to use a more random public source, but on the other hand, error correction due to information reconciliation increases, which is a factor that compromises confidentiality.

In order to overcome this, Sharifian et al. proposed a new SKA protocol, dubbed one-way secret key agreement (OW-SKA) [6], which can prevent leakage caused by frequent exchange of information. The OW-SKA protocol is a non-interactive protocol in which a secret key is transmitted unilaterally, avoiding frequent information exchanges over the public channel, which is proven to be more secure from an information-theoretic perspective if the bit error rate (BER) of the desired user, Bob, is better than that of the unauthorized user, Eve.

Some remaining issues of OW-SKA are, however, that the original article missed to discuss a realistic system model and scheme that can ensure Bob's

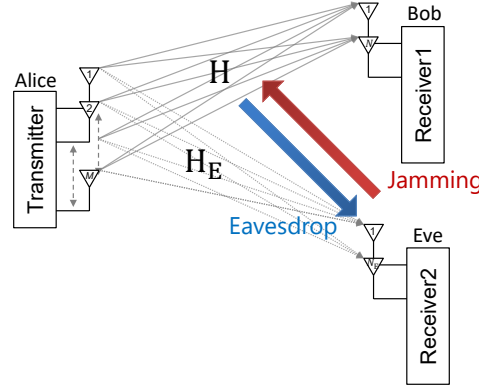


Fig. 1: System model where other legitimate receivers are untrustworthy.

BER superiority of Eve's, and the exponentially increasing complexity for generating an  $n$ -bit secret key at Bob,  $O(2^n)$ . To this end, we offer in this article the following contributions:

- We propose a PLS scenario in which legitimate users other than the desired receiver in the same wireless network are untrustworthy. We compare existing beamforming methods in passive PLS scenarios with new beamforming method in active jamming scenario, and analyze proposed attack algorithms and their security.
- We incorporate practical beamforming methods into the SKA protocol and provide information-theoretic security and a low-complexity detection scheme, which are shown to be beneficial for massive MIMO scenarios.

## 2 System Model

In this section we introduce the system model describing how information is transmitted through a wireless channel from an access point (AP) Alice to a desired receiver Bob, in the presence of a untrustworthy user Eve.

For the sake of generality, consider a multiple-antenna scenario where Alice, Bob, and Eve are respectively equipped with  $M$ ,  $N$ , and  $N_E$  antennas, with the number of antennas at Alice larger than those at Bob and Eve, *i.e.*,  $M > N$  and  $N = N_E$ , where  $N = N_E$  is to generate a correlated matrix. Given the existing trend towards massive MIMO, we consider in particular the scenario where  $M$  can reach up to thousands, while  $N$  and  $N_E$  are limited to small numbers.<sup>1</sup>

<sup>1</sup> In our future work, we consider the case where  $N_E$  is large.

In light of the above, let  $\mathbf{y}_B \in \mathbb{C}^{N \times 1}$  be the received signal at Bob, which can be modeled as

$$\mathbf{y}_B = \mathbf{H}\mathbf{W}\mathbf{s} + \sigma_{vB} \cdot \mathbf{v} \quad (1)$$

where  $\mathbf{H} \in \mathbb{C}^{N \times M}$  is the communication channel matrix between Alice and Bob, with each element given by independent and identically distributed (i.i.d.) complex Gaussian variables with zero mean and unit variance, denoted by  $h_{n,m} \sim \mathcal{CN}(0, 1)$ ;  $\mathbf{W} \in \mathbb{C}^{M \times N}$  is the linear precoding matrix employed by Alice;  $\mathbf{s} \in \mathbb{C}^{N \times 1}$  is a digitally-modulated symbol vector with elements drawn from a prescribed normalized constellation set  $\mathcal{S}$  such as phase shift keying (PSK) and quadrature amplitude modulation (QAM);  $\sigma_{vB}^2$  is the aggregate noise power at Bob; and  $\mathbf{v} \in \mathbb{C}^{N \times 1}$  denotes an additive white Gaussian noise (AWGN) vector such that  $v_n \sim \mathcal{CN}(0, 1)$ .

Similarly to the above, the received signal at Eve  $\mathbf{y}_E \in \mathbb{C}^{N_E \times 1}$  can be represented as

$$\mathbf{y}_E = \mathbf{H}_E \mathbf{W} \mathbf{s} + \sigma_{vE} \cdot \mathbf{v}_E, \quad (2)$$

where  $\mathbf{H}_E \in \mathbb{C}^{N_E \times 1}$  is the channel matrix from Alice to Eve,  $\mathbf{v}_E \in \mathbb{C}^{N_E \times 1}$  is an AWGN vector with each element following  $\mathcal{CN}(0, 1)$ , and  $\sigma_{vE}^2$  is the aggregate noise power at Eve.

For convenience and without loss of generality, both the average precoded and non-precoded transmit signal powers are normalized according to  $\mathbb{E}[\|\mathbf{W}\mathbf{s}\|_F^2] = 1$  and  $\mathbb{E}[\|\mathbf{s}\|_F^2] = 1$ , such that precoded or non-precoded methods can be fairly compared. Consequently, the average signal-to-noise ratios (SNRs) at Bob and Eve are respectively given by  $\text{SNR}_B \triangleq 10 \cdot \log_{10}(1/\sigma_{vB}^2)$  [dB], and  $\text{SNR}_E \triangleq 10 \cdot \log_{10}(1/\sigma_{vE}^2)$  [dB].

In order to model the channel similarity between Bob and Eve, we further consider the following model with a correlation coefficient  $0 \leq \rho \leq 1$ :

$$\mathbf{H}_E = \rho \mathbf{H} + \sqrt{1 - \rho^2} \mathbf{H}_{EU} \quad (3)$$

where  $\mathbf{H}_{EU} \in \mathbb{C}^{N_E \times M}$  denotes the uncorrelated quantity with each element following  $\mathcal{CN}(0, 1)$  and  $\rho$  controls the similarity.

**Attack Model:** We consider two attack models in which Eve remains passive or interferes actively.

In the passive case, Eve eavesdrops by solving the following optimization problem

$$\hat{\mathbf{s}} = \underset{\mathbf{s} \in \mathcal{S}}{\text{argmin}} \|\mathbf{y}_E - \mathbf{H}_E \mathbf{W} \mathbf{s}\|_F^2. \quad (4)$$

Here, Eve is unable to acquire the Alice-Bob channel  $\mathbf{H}$  and  $\mathbf{W}$ , which is the same assumption as [7]. Then, Eve performs powerful multivariable op-

timization for  $\mathbf{W}$  exploiting its gradient and estimates the original symbol vector.

In the active case, Eve interferes with the channel estimation process between Alice and Bob. Then, we assume that the Alice-Bob channel matrix  $\mathbf{H}$  results in

$$\mathbf{H}_{\text{BJ}} = \frac{\mathbf{H} + \sigma_J \cdot \mathbf{H}_J}{\sqrt{1 + \sigma_J^2}}, \quad (5)$$

where  $\mathbf{H}_J \in \mathbb{C}^{N \times M}$  is jamming matrix with each element following  $\mathcal{CN}(0, 1)$ , and  $\sigma_J^2$  is the power ratio. Since the channel matrix is normalized by  $\sqrt{1 + \sigma_J^2}$ , the variance of each element remains 1. In this case, Bob has to use the precoding matrix configured with the noisy channel matrix  $\mathbf{H}_{\text{BJ}}$ . We evaluate the security level under the above assumptions in Section 4.1.

### 3 Proposed SKA

In this section, we describe the proposed SKA mechanism composed of two main ingredients (*i.e.*, precoding and protocol design). Following the requirement of the OW-SKA protocol [6], it is assumed that the capacity of the desired channel (Alice to Bob) is larger than that of the untrustworthy channel (Alice to Eve). In order to reinforce this assumption, we first introduce the proposed precoding design to nullify the active attacker channel. In this case, all other legitimate users in the network are considered Eve, and Alice assumes that Eve's channel is available.

#### 3.1 Precoding Design

Taking advantage of high degrees of freedom at Alice, we propose a precoding design that actively nullifies Eve's channel under the assumption that Eve's channel is available at Alice. First, define  $\bar{\mathbf{H}} \in \mathbb{C}^{(N+N_E) \times M}$  as

$$\bar{\mathbf{H}} \triangleq \begin{bmatrix} \mathbf{H} \\ \mathbf{H}_E \end{bmatrix}. \quad (6)$$

Under the condition that  $N + N_E < M$ , the sub-spaces corresponding to Bob and Eve can be separated by pseudo-inversion, such that given the matrix

$$\mathbf{G} \triangleq \bar{\mathbf{H}}^H (\bar{\mathbf{H}} \bar{\mathbf{H}}^H)^{-1}. \quad (7)$$

The precoding matrix  $\mathbf{W}$  that nullifies Eve's channel, hereafter referred to as the nulling beamforming (N-BF) matrix, can be obtained as the first  $N$

columns of  $\mathbf{G}$ , *i.e.*,

$$\mathbf{W} \triangleq [g_{ij}]_{M \times N, (1 \leq i \leq M, 1 \leq j \leq N)}. \quad (8)$$

In N-BF, Alice can use  $\mathbf{H}_E$  due to the presence of active Eve and cancel the channels of Eve regardless of channel correlation.

### 3.2 Protocol Design

In this subsection, we describe the steps of our proposed protocol, composed of three phases (Generation, Reconciliation, and Privacy Amplification). The legitimate transmitter and receiver (Alice and Bob) observe the sequence of random variables  $\mathbf{a}$  and  $\mathbf{b}$ , respectively, and the other user Eve observes  $\mathbf{e}$ . Samples of observation at users  $\mathbf{a} \in \{0, 1\}^n$ ,  $\mathbf{b} \in \{0, 1\}^n$  and  $\mathbf{e} \in \{0, 1\}^n$  are binary vectors of length  $n$ .

#### 3.2.1 Generation Phase

Alice modulates binary message  $\mathbf{a}$  into a complex symbol vector  $\mathbf{s}$  with Gray-labeling, and transmits it using the precoding matrix  $\mathbf{W}$  to Bob as in (1). At the receiver side, Bob estimates symbol vector with maximum-likelihood decoding (MLD) as following optimization problem.

$$\hat{\mathbf{s}} = \underset{\hat{\mathbf{s}} \in \mathcal{S}}{\operatorname{argmin}} \|\mathbf{y}_B - \mathbf{H}\mathbf{W}\hat{\mathbf{s}}\|_{\mathbb{F}}^2. \quad (9)$$

With the aid of large BF gain, the symbol vector  $\hat{\mathbf{s}}$  can be accurately estimated. The estimated symbol  $\hat{\mathbf{s}}$  is demodulated into the binary vector  $\hat{\mathbf{b}}$ , which is used to generate a shared key. In this phase, Eve executes the attack algorithm given in (4) and obtains  $\mathbf{e}$ .

#### 3.2.2 Reconciliation Phase

In the original contribution of OW-SKA [6], it is verified that any universal hash function family can be used for the reconciliation phase. In this paper, we propose to use the low-density parity-check (LDPC) matrix [8] as a hash function, which satisfies the desirable properties for our scenario. The hash-based SKA protocol in wiretap channel is described in [9]. A hash vector  $\mathbf{v} \in \{0, 1\}^t$  of length  $t$  [bit] is generated as

$$\mathbf{v} \equiv \mathbf{G}_s \mathbf{a} \pmod{2}, \quad (10)$$

where  $\mathbf{G}_s \in \mathbb{Z}^{t \times n}$  is the LDPC matrix determined by a seed value  $s$ , the parameter  $t$ , and the length of input  $n$ . The parameter  $t$  is calculated in advance depending on the sample length  $n$  and the entropy between Alice and Bob. We assume that all of these parameters are known to Eve. The sparse matrix  $\mathbf{G}_s$  is randomly generated according to Gallager's algorithm [8] and the elements in the matrix are randomly distributed, depending on the value of the seed  $s$ . The matrix product calculation cannot be reversed, i.e., the irreversibility of the hash function makes it difficult to estimate the input  $\mathbf{a}$  from the information in the hash vector  $\mathbf{v}$ . Thus, Alice calculates the hash vector  $\mathbf{v}$  and then transmits it to Bob over a public channel. The goal of information reconciliation is to recover Alice's vector  $\mathbf{a}$  from  $\mathbf{b}$ . Specifically, the estimated vector at Bob,  $\hat{\mathbf{a}}$ , has to satisfy [6]

$$-\log P_{\mathbf{a}|\mathbf{b}}(\hat{\mathbf{a}}|\mathbf{b}) \leq \lambda, \quad (11)$$

where  $\lambda$  denotes an upper bound calculated by  $n$  and the entropy, which is also known to Eve. The computational complexity for obtaining  $\hat{\mathbf{a}}$  is lower bounded by  $O(2^n)$ .

**Proposed Low-Complexity Estimation:** Based on (12), we propose a low-complexity estimation method for  $\hat{\mathbf{a}}$  from  $\mathbf{b}$  at Bob, that mitigates exponentially increasing complexity with  $n$ . Let the Hamming distance  $d$  be  $d_H(\hat{\mathbf{a}}, \mathbf{b})$  and  $p$  be BER between Alice and Bob. Then, the conditional probability of (11) can be transformed into [6]

$$-\log\{p^d(1-p)^{n-d}\} \leq \lambda. \quad (12)$$

From (12), the search space for vectors  $\hat{\mathbf{a}}$  is significantly reduced when the BER  $p$  between Alice and Bob is low. Here, instead of exhaustive search, we calculate the maximum Hamming distance  $d_{\max}$  that satisfies (12) and start the search procedure from  $d = 0$  to  $d_{\max}$ . The total number of candidates  $c$  is calculated as

$$c = \sum_{r=0}^{d_{\max}} {}_n C_r, \quad (13)$$

which is much smaller than  $2^n$ . Thus, it is possible to reduce the computational complexity for obtaining  $\hat{\mathbf{a}}$ .

### 3.2.3 Privacy Amplification Phase

Finally, using a seed  $s'$  known to Eve, Alice and Bob generate hash vectors  $\mathbf{k}_A = \mathbf{G}_{s'}\mathbf{a}$  and  $\mathbf{k}_B = \mathbf{G}_{s'}\hat{\mathbf{a}}$ , which are the final shared secret key.

All in all, our propose protocol is summarized as Algorithm 1.

---

**Algorithm 1** SKA with PLS
 

---

**Input:**  $n$ -fold samples  $\mathbf{a} \in \{0, 1\}^n, \lambda, t, s \in \mathcal{S}, s' \in \mathcal{S}'$ .

**Output:** Key estimates  $\mathbf{k}_A$  and  $\mathbf{k}_B$ .

1: Alice transmits  $\mathbf{a}$  to Bob using the precoding matrix  $\mathbf{W}$ .

2: Bob receives  $\mathbf{b}$  and Eve estimates  $\mathbf{e}$  (Decoded by MLD) .

*Secret Key Agreement:*

3: Alice calculates  $\mathbf{v} \equiv \mathbf{G}_s \mathbf{a} \pmod{2}$  and transmits it to Bob through public channel.  
 $\{\mathbf{G}_s \in \mathbb{Z}^{t \times n} \text{ is LDPC matrix}\}$

4: Bob generates a list of candidates for  $\mathbf{a}$  that satisfies  $-\log\{p^d(1-p)^{n-d}\} \leq \lambda$ , where  
 $d = d_H(\hat{\mathbf{a}}, \mathbf{b})$  and  $p = \text{BER estimated by SNR}$ .

5: **if** no  $\hat{\mathbf{a}}$  is found or  $\hat{\mathbf{a}}$  is not unique **then**

6:   Abort the protocol

7: **end if** Bob finds a unique  $\hat{\mathbf{a}} \in \mathcal{T}(\mathbf{a}|\mathbf{b})$

8: **return** Alice:  $\mathbf{k}_A \equiv \mathbf{G}_{s'} \mathbf{a} \pmod{2}$ , Bob:  $\mathbf{k}_B \equiv \mathbf{G}_{s'} \hat{\mathbf{a}} \pmod{2}$

---

## 4 Simulation Results

Numerical evaluations are offered in this section so as to assess the performance of the proposed SKA protocol. In order to evaluate the performance from different perspectives, we first illustrate the secrecy capacity of the proposed SKA protocol in comparison with other precoding methods, followed by advantage of the proposed SKA protocol in terms of complexity.

### 4.1 Secrecy Capacity

First, we evaluate in Fig. 2(a) the secrecy capacity of the proposed SKA protocol compared with popular precoding methods such as Zero-Forcing (ZF), singular value decomposition (SVD), maximum ratio combining (MRC), and minimum mean square error (MMSE). At this time, we analyzed the security corresponding to the attack model shown in Section 2. We used the parameters  $M = 1024, N = 2, N_E = 2, \sigma_J = 1$  and 16QAM. Specifically, the security evaluation was performed by secrecy capacity, first when Eve is passive and decodes using only its own channel information, and then when Eve actively interferes with the beamformer. The secrecy capacity evaluated hereafter defined as the difference of average mutual information (AMI) between Bob and Eve in [10] as  $C_s = \max\{0, \text{AMI}_B - \text{AMI}_E\}$ . It can be seen from the Fig. 2(a) that N-BF can retain a high secrecy level even at high SNR regimes, in which other precoding approaches fail to convey information securely. This is due to the assumption that Eve's channel is available at Alice as shown in Section 3 (*i.e.*, active user Eve). As for the other beamformer methods, they can achieve high secrecy capacity that meets half the capacity, but security performance is not preserved at high SNRs where Eve's channel is affected. In contrast, the attack model with interference showed significant degradation

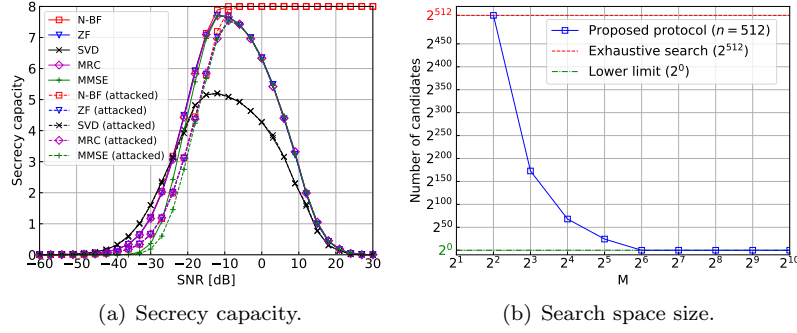


Fig. 2: Performance comparisons.

of secrecy capacity for all methods, including N-BF. Here, the power level of the jamming is  $\sigma_J = 1$ . This indicates that the wireless channel exposed to active attack cannot maintain confidentiality.

## 4.2 SKA Complexity

Having clarified that the active eavesdropper assumption is needed to maintain secure communications, we next illustrate the behavior that the proposed algorithm can significantly reduce its complexity as  $M$ , the number of transmit antennas, increases, where we have  $\text{SNR} = 5$  [dB]. The result is shown in Fig. 2(b), in which the number of candidates required to reach secret key agreement based on the proposed SKA protocol is demonstrated as a function of  $M$ , where the number of candidates is equal to  $c$  given by (13). It can be seen from Fig. 2(b) that as  $M$  increases, extremely high quality wireless communication is possible between the legitimate transmitter and receiver, and the range of vectors  $\hat{\mathbf{x}}$  to be estimated decreases because Bob's MLD is almost error free. In addition, the red line in Fig. 2(b) shows the upper boundary of the number of candidates in the exhaustive search, while the green line shows the lower boundary, a single candidate, indicating that the search space is significantly reduced.

## 5 Conclusion

In this study, we proposed a new protocol that introduces SKA as a method to realize PLS, and analyzed how well security is actually maintained when executing PLS scenarios. Since systems containing untrustworthy users have



information about each other's counterpart, we proposed a beamforming method that uses this information to eliminate the rogue user's channel and, conversely, analyzed in detail the security performance when subjected to interferes with the channel estimation. Furthermore, we show that the use of beamforming in a massive MIMO environment mitigates the computational complexity of the SKA and demonstrate the effectiveness of PLS security using secret keys.

## Acknowledgement

This work was partially supported by JST SICORP Grant Number JP-MJSC20C1, Japan.

## References

- [1] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.
- [2] P. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *35th Annual Symposium on Foundations of Computer Science*, Nov. 1994, pp. 124–134.
- [3] L. Mucchi, S. Jayousi, S. Caputo, E. Panayirci, S. Shahabuddin, J. Bechtold, I. Morales, R.-A. Stoica, G. Abreu, and H. Haas, "Physical-layer security in 6G networks," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1901–1914, 2021.
- [4] A. Khisti, G. Wornell, A. Wiesel, and Y. Eldar, "On the gaussian MIMO wiretap channel," in *IEEE International Symposium on Information Theory*, Jun. 2007, pp. 2471–2475.
- [5] U. M. Maurer, "Secret key agreement by public discussion from common information," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 733–742, 1993.
- [6] S. Sharifian, A. Poostindouz, and R. Safavi-Naini, "A capacity-achieving one-way key agreement with improved finite blocklength analysis," in *International Symposium on Information Theory and Its Applications*, Oct. 2020, pp. 407–411.
- [7] T. R. Dean and A. J. Goldsmith, "Physical-layer cryptography through massive MIMO," *IEEE Transactions on Information Theory*, vol. 63, no. 8, pp. 5419–5436, 2017.
- [8] R. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, 1962.
- [9] J. Muramatsu and S. Miyake, "Construction of codes for the wiretap channel and the secret key agreement from correlated source outputs based on the hash property," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 671–692, 2012.
- [10] L. Wang, S. Bashar, Y. Wei, and R. Li, "Secrecy enhancement analysis against unknown eavesdropping in spatial modulation," *IEEE Communications Letters*, vol. 19, no. 8, pp. 1351–1354, 2015.

# Efficient implementation of stream cipher SNOW 3G for resource-constrained devices

G. Cotrina, A. Peinado and A. Ortiz

**Abstract** SNOW 3G is one of the stream ciphers proposed by the standard to protect data in 3G and 4G mobile communications. In recent years, the possible adaptation of the algorithm to the requirements of 5G communications has been studied, resulting in more efficient implementations, although modifications of the cipher, such as SNOW-V and SNOW-Vi, have finally been proposed to achieve the throughput and security that 5G requires. In this article, an efficient implementation of SNOW 3G is presented. It takes advantage of the 32-bit architecture of the processors and employs  $n$ -grouped operations giving raised to similar performance than the SIMD instructions set employed in the most recent stream ciphers. The implementation is based on the use of the equivalent binary model of the LFSRs defined in  $GF(2^n)$ . Although SNOW 3G seems ruled out for 5G communications, it is still interesting to have efficient implementations in 4G that allow its integration in devices with limited processors.

## 1 Introduction

3G and 4G mobile communication systems define a series of algorithms to provide confidentiality and integrity in radio links. The SNOW 3G stream cipher [1] is the primitive used to carry out data encryption and integrity through the UEA2 and UIA2 algorithms (UEA/UIA stand for UMTS Encryption/Integrity Algorithm), in the case of UMTS (3G) and through the EEA1 and EIA1 algorithms (EEA/EIA stand for Evolved packed systems Encryption/Integrity Algorithm), in the case of LTE (4G), which is the adaptation of UEA2/UIA to the specific parameters of 4G.

---

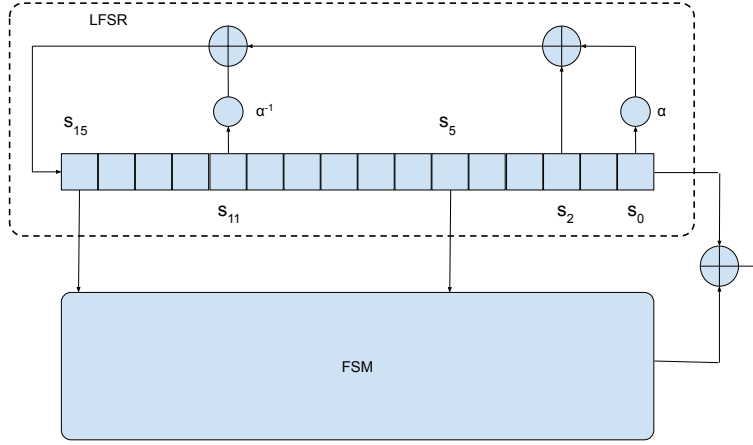
G. Cotrina, A. Peinado, A. Ortiz

Escuela Técnica Superior de Ingeniería de Telecomunicación, Universidad de Málaga, Campus de Teatinos 29071 Málaga - Spain, e-mail: gcotrinacuenca@uma.es, apeinado@ic.uma.es, aortiz@ic.uma.es

Recently, studies have been carried out in order to evaluate the possible use of SNOW 3G in 5G systems, which have more demanding throughput and security requirements. On the one hand, this standard determines the use of symmetric cryptographic systems with 256-bit keys [2], which is not a problem since, although the SNOW 3G algorithm is used with 128-bit keys in 3G/4G communications, it was originally defined to be able to use longer keys. On the other hand, the standard requires that the cryptographic algorithms can operate at 10 and 20 Gbps, in the uplink and downlink, respectively [3]. However, improvements made to SNOW 3G implementations have failed to go beyond 9 Gbps [4]. This fact, together with an effective key length of less than 256 bits, according to the analysis carried out in [5], suggests that SNOW 3G will be replaced by similar but improved versions, such as SNOW-V and SNOW-Vi [6].

However, attempts to improve the performance of SNOW 3G have generated new implementations based on parallelization and the use of Single Instruction Multiple Data (SIMD)-type instruction sets, on which the new SNOW-V and SNOW-Vi algorithms are based, to take advantage of the features of the most advanced CPUs that also include native instructions to work with the S-boxes defined in the SNOW family. Unlike these proposals, this work describes an improvement of the SNOW 3G implementation without using the resources of most advanced CPU, that is, SIMD instructions set o cryptographic native instructions. Hence, the improvement is focused on devices with several limitations, in the sense that they do not provide the mentioned advanced capabilities. This kind of devices represents those using 3G/4G networks that will continue to operate for a long time, taking into account the latest report on Telecommunications from the National Markets Commission of Spain [7], that reveals a slow migration towards the 5G systems.

The efficient implementation proposed uses an approach similar to that used by SIMD instruction sets, but without the need for a CPU with such features. In a certain way, it is about taking advantage of the ability of the basic instructions of the CPU, such as the XOR operation, to operate on the maximum number of bits simultaneously, that is, to apply the SIMD concept without the need for SIMD-specific instructions. To do this, it takes advantage of the fact that SNOW 3G uses a Linear Feedback Shift Register (LFSR) defined on the extended field  $GF(2^{32})$  to generate 32 bits in each iteration. This type of LFSR can be implemented by means of an equivalent binary model composed of multiple binary LFSRs, much simpler and all controlled by the same feedback polynomial [8]. The rest of the paper is structured as follows. In Section 2, the stream cipher SNOW 3G is described. Section 3 and 4 describe the equivalent binary model and its application to the particular case of SNOW 3G. Section 5 contains the performance evaluation and section 6 the conclusions.



**Fig. 1** Block diagram for SNOW 3G stream cipher

## 2 SNOW 3G description

SNOW 3G is a word-oriented stream cipher designed to generate a sequence of 32-bit words that will be used as the keystream to be xored with the plaintext. The cipher is controlled by a 128-bit key and a 128-bit initialisation variable (IV) and is composed by two main blocks, as it is shown in Fig. 1, an LFSR and a Finite State Machine (FSM). The LFSR is defined in  $GF(2^{32})$ . Hence, every cell contains 32 bits. As it has 16 cells,  $s_0, s_1, \dots, s_{15}$ , the LFSR has an internal state of 512 bits. The LFSR produces a pseudorandom sequence determined by the feedback polynomial  $g(x) \in GF(2^{32})[x]$

$$g(x) = \alpha x^{16} + x^{14} + \alpha^{-1} x^5 + 1, \quad (1)$$

where  $\alpha$  is a root of the polynomial  $x^4 + \beta^{23} x^3 + \beta^{245} x^2 + \beta^{48} x + \beta^{239} \in GF(2^8)[x]$  and  $\beta$  is a root of  $x^8 + x^7 + x^5 + x^3 + 1 \in GF(2)[x]$ . Hence, the feedback connection can be expressed as follows

$$s_{15}^{(t+1)} = \alpha^{-1} s_{11}^{(t)} + s_2^{(t)} + \alpha s_0^{(t)}, \quad (2)$$

where  $+$  denotes the bitwise XOR operation. The FSM block is responsible for nonlinear operations. It takes two 32-bit words from the LFSR as the inputs and outputs a 32-bit word to be xored with the LFSR output.

Two operation modes are defined. The first one performs an initialisation. The 128-bit key and IV are combined to initialise the LFSR state and then it is operated 32 times without producing output. Then the normal mode begins to output the 32-bit word sequence

### 3 Equivalent binary model of LFSR in $GF(2^n)$

The LFSR is one of the most widely used pseudorandom sequence generators in the design of stream ciphers due to its simplicity, ease of design and the good statistical properties of the sequences they generate. As it is well known [9], when the feedback polynomial used is primitive, the generated sequence is of maximum length,  $2^L - 1$ , where  $L$  is the number of cells in the LFSR. These sequences of maximum length are known as  $m$ -sequences. In most cases, LFSRs are defined over  $GF(2)$  and therefore only one bit is generated in each iteration. This architecture, initially oriented to a hardware implementation, allows its application in high-speed communications since only the XOR operation is used. However, it is very common to use software implementations, as is the case of mobile communication networks in which the virtualization of nodes is increasing [6]. In these and other cases, processor word lengths make software implementations generally inefficient [10]. To overcome this situation, recently, stream ciphers with LFSRs defined in extended fields,  $GF(2^n)$ , where  $n$  is the word length of the processor, have been proposed [11, 12]. Thus, in each iteration,  $n$  bits will be generated. The drawback of operating on extended fields is that the operations to obtain a new element from the sequence are much more computationally complex. For this reason, current implementations carefully select the feedback polynomials instead of using any primitive polynomial, as is the case in the binary field.

Recently, in [8], an equivalent model of the LFSRs defined in  $GF(2^n)$  has been proposed. The model allows LFSRs to be expressed by means of  $n$  binary LFSRs, that is, defined in  $GF(2)$ , with the particularity that all the binary LFSRs have the same feedback polynomial. As described in [8], the model is based on the relationship between the  $m$ -sequences generated by the extended field LFSR and the  $m$ -sequences generated by the binary LFSRs. Therefore, if we call  $\langle L, f(x), n \rangle$  the LFSR of  $L$  cells defined in  $GF(2^n)$  with a feedback polynomial  $f(x) \in GF(2^n)[x]$  of degree  $L$ , and similarly,  $\langle nL, p(x) \rangle$  to the LFSR of  $nL$  binary cells with feedback polynomial  $p(x) \in GF(2)[x]$  of degree  $nL$ , the model described in [8] states that the LFSR  $\langle L, f(x), n \rangle$  is equivalent to the set formed by  $n$  LFSR  $\langle nL, p(x) \rangle$  such that the sequences generated by each binary LFSR correspond to each of the decimated sequences of the LFSR  $\langle L, f(x), n \rangle$ . Equivalently, the sequence that generates the  $i$ -th binary LFSR matches the decimated sequence that is obtained by taking the  $i$ -th bit of each element of the sequence generated by the LFSR  $\langle L, f(x), n \rangle$ . On the other hand, although the polynomials  $f(x)$  and  $p(x)$  are related, obtaining  $p(x)$  from  $f(x)$  can be done much more efficiently by generating the first few elements of the original sequence and calculating the minimal polynomial  $p(x)$  of any of the decimated sequences, as shown in [8].

## 4 Efficient implementation

The model in [8], described in the previous section, can be applied to SNOW 3G, since the cipher is based on an LFSR that operates on  $GF(2^{32})$ . As shown in the figure 1, the LFSR block is independent of the FSM block, so modifications can be applied to the LFSR implementation without affecting the FSM, as long as the connections between both are maintained. Therefore, the implementation proposed in this work focuses exclusively on modifying only the LFSR block.

The equivalent model transforms the original 512-bit LFSR  $\langle 16, f(x), 32 \rangle$  into 32 binary LFSRs  $\langle 512, p(x) \rangle$  of 512 bits each. The first step is to obtain the polynomial  $p(x)$  that determines the feedback of the binary LFSRs. The most efficient procedure is to generate enough elements of the pseudorandom sequence, generate the 32 decimated sequences, and calculate the minimal polynomial using the Massey-Berlekamp algorithm [13]. Since all 32 sequences have the same feedback polynomial, it is only necessary to apply the Massey-Berlekamp algorithm to one of them. The obtained polynomial is

$$\begin{aligned}
p(x) = & 1 + x^5 + x^{16} + x^{19} + x^{20} + x^{21} + x^{25} + x^{26} + x^{30} + x^{33} + x^{36} \\
& + x^{37} + x^{39} + x^{41} + x^{44} + x^{45} + x^{46} + x^{47} + x^{49} + x^{50} + x^{51} + x^{53} \\
& + x^{55} + x^{56} + x^{57} + x^{58} + x^{59} + x^{63} + x^{65} + x^{66} + x^{67} + x^{68} + x^{69} \\
& + x^{70} + x^{72} + x^{73} + x^{75} + x^{76} + x^{78} + x^{81} + x^{87} + x^{88} + x^{89} + x^{93} \\
& + x^{94} + x^{97} + x^{98} + x^{102} + x^{103} + x^{104} + x^{105} + x^{109} + x^{111} + x^{112} \\
& + x^{113} + x^{115} + x^{116} + x^{117} + x^{118} + x^{119} + x^{120} + x^{121} + x^{123} + x^{124} \\
& + x^{125} + x^{126} + x^{128} + x^{129} + x^{131} + x^{132} + x^{136} + x^{138} + x^{141} + x^{143} \\
& + x^{146} + x^{147} + x^{149} + x^{151} + x^{152} + x^{153} + x^{155} + x^{156} + x^{158} + x^{163} \\
& + x^{164} + x^{165} + x^{169} + x^{171} + x^{172} + x^{173} + x^{174} + x^{175} + x^{177} + x^{178} \\
& + x^{179} + x^{184} + x^{190} + x^{193} + x^{199} + x^{201} + x^{203} + x^{206} + x^{208} + x^{210} \\
& + x^{213} + x^{214} + x^{215} + x^{216} + x^{217} + x^{222} + x^{226} + x^{227} + x^{228} + x^{229} \\
& + x^{230} + x^{232} + x^{233} + x^{235} + x^{236} + x^{238} + x^{239} + x^{240} + x^{243} + x^{248} \\
& + x^{249} + x^{251} + x^{253} + x^{255} + x^{257} + x^{258} + x^{259} + x^{262} + x^{264} + x^{269} \\
& + x^{271} + x^{274} + x^{276} + x^{278} + x^{281} + x^{282} + x^{283} + x^{285} + x^{286} + x^{289} \\
& + x^{291} + x^{292} + x^{293} + x^{294} + x^{295} + x^{296} + x^{297} + x^{298} + x^{299} + x^{301} \\
& + x^{303} + x^{304} + x^{305} + x^{306} + x^{307} + x^{310} + x^{319} + x^{322} + x^{324} + x^{326} \\
& + x^{329} + x^{330} + x^{333} + x^{336} + x^{337} + x^{338} + x^{339} + x^{341} + x^{343} + x^{344} \\
& + x^{345} + x^{347} + x^{350} + x^{351} + x^{352} + x^{353} + x^{354} + x^{355} + x^{356} + x^{357} \\
& + x^{359} + x^{362} + x^{363} + x^{365} + x^{366} + x^{371} + x^{372} + x^{374} + x^{375} + x^{385} \\
& + x^{387} + x^{388} + x^{390} + x^{392} + x^{393} + x^{395} + x^{396} + x^{398} + x^{399} + x^{400} \\
& + x^{402} + x^{405} + x^{409} + x^{411} + x^{412} + x^{414} + x^{419} + x^{420} + x^{423} + x^{424} \\
& + x^{427} + x^{429} + x^{431} + x^{432} + x^{433} + x^{434} + x^{437} + x^{438} + x^{441} + x^{444} \\
& + x^{445} + x^{446} + x^{449} + x^{452} + x^{458} + x^{459} + x^{461} + x^{462} + x^{466} + x^{469} \\
& + x^{470} + x^{471} + x^{477} + x^{478} + x^{480} + x^{481} + x^{483} + x^{485} + x^{486} + x^{490} \\
& + x^{493} + x^{494} + x^{501} + x^{502} + x^{504} + x^{510} + x^{512}
\end{aligned} \tag{3}$$

Once the polynomial  $p(x)$  has been calculated, the 32 binary LFSRs can be implemented as shown in figure 2, maintaining the connections with the FSM block. Thus, there are 32 LFSRs that must be executed in parallel. This scheme is, precisely, the one that allows to obtain the maximum performance of the processors, since the instruction that carries out the XOR operation operates with 32-bit numbers. These same processors that are inefficient when executing binary LFSR software implementations because each feedback only generates one bit, despite working with instructions that can handle 32-bit data, can now take advantage of their full potential because each XOR operation is being applied to 32-bit data, that is, to the 32 binary LFSRs simultaneously.

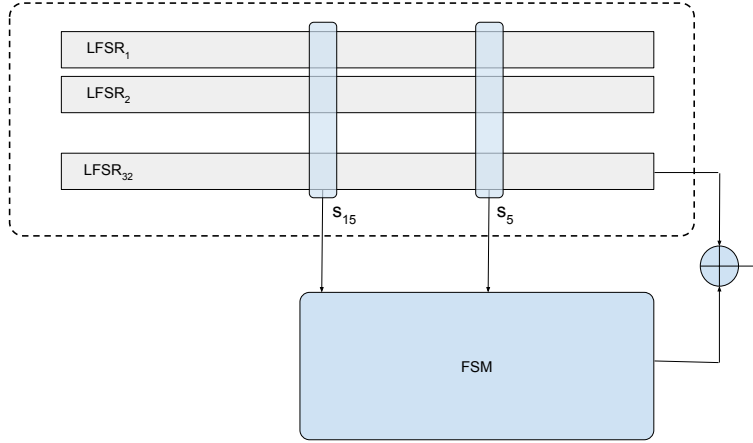
However, although the binary equivalent model allows the generation of the same sequences as the original LFSR, it is necessary to calculate the correct seeds for

binary LFSRs since the original seed has only 512 bits length. To do this, it must be taken into account that although the 32 binary LFSRs have the same feedback polynomial and, therefore, the  $m$ -sequence they generate is the same, there is a displacement or phase shift  $\theta_j$  between them that is determined by the seed of each one. These relative displacements between the different  $m$ -sequences can be calculated giving rise to the corresponding matrices that would allow obtaining the state of any  $m$ -sequence from the others. Considering the 0-th  $m$ -sequence as a reference,  $A^{(j)} = A^{\theta_j}$  is the matrix to obtain the state of the  $j$ -th  $m$ -sequence from 0-th  $m$ -sequence. In this way, if the state of any of the binary LFSR is known, the states of all of them can be computed. This approach could be applied to compute the binary seeds from only one binary seed. However, no binary seeds are completely known. The 512 bits of the initial seed are distributed among all binary LFSR giving rise to partial binary seeds.

For any given seed  $(s_0, s_1, \dots, s_{L-1})$ , the seeds of the binary LFSRs can be represented as  $(s_{0,j}, s_{1,j}, \dots, s_{nL-1,j})$ , where  $s_j = (s_{j,0}, s_{j,1}, \dots, s_{j,31})$ . Furthermore, we have,

$$(s_{0,j}, s_{1,j}, \dots, s_{nL-1,j}) = (s_{0,0}, s_{1,0}, \dots, s_{nL-1,0})A(j), \quad (4)$$

where  $s_{L,j}, \dots, s_{nL-1,j}$  are unknowns for every  $0 \leq j \leq n-1$ . The values  $s_{L,0}, \dots, s_{nL-1,0}$  can be obtained solving the linear system composed with the  $(nL-L)$  equations with the known values  $s_{0,j}, s_{1,j}, \dots, s_{L-1,j}$ , for  $0 \leq j \leq n-1$ .



**Fig. 2** Efficient implementation of SNOW 3G stream cipher



## 5 Performance evaluation

The proposed implementation maximizes the use of XOR operations, since the time that the processor takes to perform an XOR instruction with 1-bit data is the same with 32-bit data. Therefore, the increase in the bit generation rate, by a factor  $n = 32$ , produced by using LFSRs defined in  $GF(2^{32})$ , would generate, from a theoretical point of view, an improvement in the execution time of the same amount. However, it must be taken into account that the feedback determined by  $f(x)$  in the LFSR  $< 16, f(x), 32 >$  and by  $p(x)$  in the LFSR  $< 512, p(x) >$  are different. In the specific case of SNOW 3G, the polynomial  $p(x)$  has 250 coefficients, which considerably limits the improvement. In order to quantify the improvement, we have made an estimation on the processing time. We denote  $t_A$  the number of operations of the function  $A$ . The original version of SNOW 3G can be performed using a total number of operations

$$t_{TOTAL} = t_{SEED} + (32 + n)(t_F + t_{LFSR}), \quad (5)$$

where  $t_{SEED}$  refers to the initial state setting,  $t_F$  the FSM output computation and  $t_{LFSR}$  the computation of the LFSR next state. Eq 5 includes the effect of the 32 iterations of the Initialization phase and the  $n$  iterations of keystream phase responsible for the on-demand generation of  $n$  32-bit words, as it is described in section 2. Taking as reference the code published in [1], we estimate  $t_{SEED} = 29$ , since the initial state setting performs 10 XOR operations and 19 accesses to memory;  $t_F = 98$ , since 2 additions modulus  $2^{32}$ , 2 AND operations, 2 XOR operations and 2 S-box functions are performed in the FSM for the output computation; and  $t_{LFSR} = 1372$ , related to the feedback computation in the extended field  $GF(2^{32})$ . Note that these computations are performed using an iterative scheme determined by the exponents of  $\beta$  in section 2. Consequently, we have

$$t_{TOTAL} = 29 + (32 + n)1470, \quad (6)$$

The improved version proposed in this paper does not modify the FSM output computation. The improvement is focused on the LFSR that is performed using 250 XOR operations and 249 accesses to memory. The initial state setting is now performed by 10 XOR operations and 515 accesses to memory, due to the utilization of 32 binary LFSR. Hence, the overall performance of the improved version is

$$t_{TOTALeq} = t_{SEEDeq} + (32 + n)(t_F + t_{LFSReq}) = 525 + (32 + n)597 \quad (7)$$

In order to estimate the theoretical speed-up provided by the binary equivalent model of LFSR defined in  $GF(2^{32})$ , we first calculate the improvement  $x_{LFSR}$  of the LFSR implementation as

$$x_{LFSR} = \frac{t_{LFSR}}{t_{LFSReq}} = \frac{1372}{499} = 2.75 \quad (8)$$

The LFSR feedback computation takes most of computational effort. The portion of the overall time consumed by LFSR can be computed as

$$F_{LFSR} = \frac{(32 + n)t_{LFSR}}{29 + (32 + n)(t_F + t_{LFSReq})}, \quad (9)$$

which is bounded between 93.27%, when  $n \rightarrow 0$ , and 93.33%, when  $n \rightarrow \infty$ . Hence, the theoretical speed-up  $S$  produced by the proposed improvement is

$$S = \frac{x_{LFSR}}{F_{LFSR} + x_{LFSR}(1 - F_{LFSR})} = 2.46, \quad (10)$$

However, experimental tests indicate an improvement of 3.3 times in the execution time, instead of 2.75, without taking into account the execution time of the FSM block, and 3.1 times considering the overall time. This difference may be caused by some theoretical considerations, such as, the assumption that all operations take the same processing steps.

The practical application of the equivalent model requires the transformation of the initial 512-bit seed into a longer seed that can be used to initialize the 32 binary LFSRs. To do this, it is necessary to precalculate the displacement matrices  $A^{(j)}$  which, as it depends only on  $f(x)$ , are always the same. Therefore, the execution time used in the calculation of these matrices has not been taken into account for the general analysis. However, these matrices must be applied in the initialization phase of the cipher. Since this calculation is only applied in the initialization phase, it does not significantly affect the total execution time, which is maintained three times better than the original. The experimental tests have been performed on an Intel computer using Python 3.9 programming language to implement both versions, the original, adapted from the official C code published in [1], and the proposed improvement. In this way, no SIMD instructions have been employed and only 32-bit data.

## 6 Conclusions

In this work, we have presented an efficient implementation of SNOW 3G stream cipher without the need to use a SIMD instruction set, but using a similar strategy, grouping  $n = 32$  binary LFSRs to execute the 32 simultaneous feedbacks in the same time as a single feedback. The reduction in the execution time is bounded by the high amount of nonzero coefficients of the equivalent feedback polynomial  $p(x)$ . Therefore, better results can be expected in the application of this model in other LFSR-based ciphers with a smaller number of coefficients.

**Acknowledgements** This work was supported in part by the research group BIOSIP (TIC-251) and by the Spanish State Research Agency (AEI) of the Ministry of Science and Innovation (MICINN), project P2QProMeTe (PID2020-112586RB-I00/AEI/10.13039/501100011033)

## References

1. SAGE. Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2. Version 1.1, ETSI/SAGE, (2006).
2. 3GPP. TS 33.841 (V16.1.0): 3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects; Security aspects; Study on the support of 256-bit algorithms for 5G (Release 16). (March 2019).
3. 3GPP. TS 33.501: 3rd Generation Partnership Project. Technical Specification Group Services and System Aspects; Security architecture and procedures for 5G system. (December 2020).
4. Ekdahl, P., Johansson, T., Maximov, A., Yang, J.: A new SNOW stream cipher called SNOW-V. In: IACR Transactions on Symmetric Cryptology, 3, pp. 1–42. (2019)
5. Yang, J.: Contributions to Confidentiality and Integrity Algorithms for 5G. Department of Electrical and Information Technology, PhD thesis, Lund University (2021)
6. Ekdahl, P., Maximov, A., Johansson, T.: SNOW-Vi: an extreme performance variant of SNOW-V for lower grade CPUs. In: WiSec '21: Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks June, pp 261–272. (2021) <https://doi.org/10.1145/3448300.3467829>
7. Comisión Nacional de los Mercados y la Competencia. ESTAD/CNMC/002/21. Informe Económico Sectorial de las Telecomunicaciones y el Audiovisual, (2020).
8. Espinosa García, J., Cotrina, G., Peinado, A., Ortiz, A.: Security and Efficiency of Linear Feedback Shift Registers in  $GF(2^n)$  Using  $n$ -Bit Grouped Operations. In: Mathematics, 10, 996. (2022)
9. Golomb, S.W.: Shift Register Sequences, 3rd Revised ed. Aegean Park Press: Laguna Hills, CA, USA, (2017)
10. Delgado-Mohatar, O., Fúster-Sabater, A., Sierra, J.M.: Performance evaluation of highly efficient techniques for software implementation of LFSR. In: Comput. Electr. Eng., 37, pp. 1222–1231. (2011)
11. Kiyomoto, S., Tanaka, T., Sakurai, K.: K2. A stream cipher algorithm using dynamic feedback control. In: Proceedings of the International Conference on Security and Cryptography, SECRYPT, Barcelona, Spain, 28–13 July 2007; Hernando, J., Fernández-Medina, E., Malek, M., Eds.; INSTICC Press: Lisboa, Portugal, pp. 204–213. (2007)
12. George, K., Michaels, A.J.: Designing a Block Cipher in Galois Extension Fields for IoT Security. In: IoT, 2, pp. 669–687. (2021).
13. Massey, J.L.: Shift register synthesis and BCH decoding. In: IEEE Trans. Inf. Theory, 15, pp. 122–127 (1969)

# State of the art of cybersecurity in cooperative, connected and automated mobility

O. Castillo Campo, V. Gayoso Martínez, L. Hernández Encinas, A. Martín Muñoz and R. Álvarez Fernández

**Abstract** In recent years, mobility systems have been steadily improving their performance in terms of connectivity, allowing road users to communicate among them and with the road infrastructure either directly or through internet cloud-based services. The aim of this contribution is to provide an overview of inter-vehicle communications cybersecurity challenges in the cooperative, connected and automated mobility sector and make possible a further analysis of the cryptographic methods and tools that can leverage their security in a world where quantum computers will be a practical menace at some point in the future.

## 1 Introduction

Information technologies and intelligent systems take an important part in several aspects of our daily life. Advances in digitalization have made possible that many society activities are assisted or even driven by some type of intelligent systems. In addition, most of these technologies are supported by global access to the internet thanks, among other technologies, to mobile phone communications. However, this new situation has developed new risks and security threats.

Cybersecurity has become a key issue for all business sectors, particularly in critical ones such as energy or transport. In Spain, the transport sector is one of the strategic areas suffering more cyberattacks and, due to that, it is of increasing relevance in the scope of national security strategy [1]. Cybercriminals and malicious

---

Óscar Castillo Campo, Roberto Álvarez Fernández  
Department of Industrial Engineering, Universidad Nebrija, Madrid, Spain, e-mail: ocastillo@nebrija.es, ralvarez@nebrija.es

Víctor Gayoso Martínez  
Data Research & Computation Group (DRACO), Centro Universitario de Tecnología y Arte Digital (U-tad), Las Rozas, Spain, e-mail: victor.gayoso@u-tad.com

Luis Hernández Encinas, Agustín Martín Muñoz  
Institute of Physical and Information Technologies (ITEFI), Spanish National Research Council (CSIC), Madrid, Spain, e-mail: luis@iec.csic.es, agustin@iec.csic.es

insiders frequently use ransomware attacks, while Distributed Denial of Service (DDoS) attacks and other means for disrupting the operations and stealing vital data cause important economic losses.

In this context, mobility systems have been steadily improving their performance in terms of connectivity, allowing road users to communicate among them and with the road infrastructure either directly or through internet cloud-based services. These new mobility technologies can be included under the term Intelligent Transport Systems (ITS). ITS can be used for improving traffic efficiency and road safety, managing road infrastructures, and enhancing the driving experience. In this way, ITS technologies are the tool that will allow to enable Cooperative, Connected and Automated Mobility (CCAM) in the near future.

ITS applications are based on data collection, processing, and analysis. For example, connected vehicles can learn from each other in order to generate and maintain a reliable picture of the driving environment, such as the presence of pedestrians and cyclists, dangerous crossings or areas with adverse weather conditions.

ITS require a complex ecosystem and depend on the continuity of communications between the vehicle and its environment. However, the proliferation of connected and automated technologies for transport systems using short range or mobile connectivity makes possible new opportunities and vector attacks for cyber intrusion. Accordingly, cybersecurity is a key feature for trusted communication of road users with other road users, infrastructures, and cloud-based services. CCAM systems must be free of operational fails and cybersecure under the operating conditions in which the system is designed, and make possible reliable, safe, and secure operation of mobility systems and services.

The aim of this article is to provide a summary of the specific threats and requirements for CCAM cybersecurity assets and, in addition to that, to review some of the latest cryptographic advances that can be used to guarantee a good level of security even in the presence of quantum computers.

This contribution is organized as follows: Section 2 describes the most important elements of the CCAM ecosystem. Section 3 enumerates the list of challenges in CCAM solutions. Section 4 discusses specific CCAM cybersecurity issues. Section 5 presents the latest advances in cryptography associated to the emergence of quantum computers. Finally, conclusions are offered in Section 6.

## 2 CCAM ecosystem

There are many stakeholders, infrastructures, connected services, and products in CCAM systems. In this complex ecosystem, it is possible to differentiate between connected services and off-board systems on the one hand, and physical infrastructures, vehicles and mobility users, mobility products and associated equipment, services based on Operational Technology (OT), Information Technology (IT), electric/electronics architectures, and on-board systems on the other.

Smart and autonomous vehicles could be considered as the mayor component of CCAM systems due to the significant amount of personal vehicles moving on the

roads. An essential part of these infrastructures are the Vehicular ad-hoc networks (VANETs). A VANET is a communication network using wireless inter-vehicle communication technology. Besides inter-vehicle communication, the devices onboard the vehicle need to communicate with each other using in-vehicle networks with the goal to implement various distributed control functions.

Sensors and device microcontrollers are instances of CCAM components used for collecting and processing environmental data while in routing, in addition to managing and controlling tasks in smart traffic controllers. Besides, applications, devices, and services around the Internet of Things (IoT) use mobile technology, so in this sense smartphone performance and connectivity standards will play a special role. Therefore, smartphones are a key component of CCAM systems, enabling integration with smart vehicles and other components of the system. As an example, the EU funded the Autopilot project [2], promoting automated driving thanks to the IoT and mobile communications.

Besides, the technologies that will enable the uptake of CCAM in the near future are summarized below:

1. *Computation*: Response time constraints, high volume of data and cryptographic communication protocols in the dynamic CCAM scenario need microcontrollers or on-board computers with a great processing capability. To this end, there are two distinct trends: cloud and fog computing. Unlike cloud computing, where the data is stored and processed in remote servers hosted on the internet, in fog computing most of the information is processed close to the source.
2. *Analytics*: Taking advantage of cloud or fog computing capabilities, it is possible to create a hierarchical analytics network based on regional traffic intermediary servers and a central traffic authority.
3. *Communication networks*: The main types of CCAM communications are Vehicle-to-infrastructure (V2I), Vehicle-to-vehicle (V2V) and Vehicle-to-network (V2N). V2I communication consists in the information exchange between vehicles and road infrastructure, such as the communication between the vehicle On-Board Units (OBUs) and the infrastructure Road-Side Units (RSUs). In comparison, V2V enables to share traffic information with other vehicles. Besides, V2N represents the communication from V2I devices to the central control servers. In general, CCAM communications used for broadcasting and relaying information fall under the Vehicle-to-Everything (V2X) umbrella.

All these CCAM communications are supported by VANETs, which are based on wireless network technologies and face several security challenges such as Denial-of-Service (DoS), sybil and replay attacks, and user privacy issues, among others, that potentially can be spread to other components jeopardizing the security of the entire system.

### 3 Challenges in CCAM solutions

The complex CCAM ecosystem faces a great variety of threats<sup>1</sup>, vulnerabilities<sup>2</sup>, and risks<sup>3</sup>. CCAM systems are subject to the following threats:

- Physical attacks. The ITS elements display a hybrid nature, as they are composed of hardware and digital components. Thus, they can be exposed to physical damage caused not only by an attacker but also by meteorological disasters.
- Accidental hardware or software mistakes.
- Service interruptions (e.g., disruptions in the electricity supply).
- Malicious activities like viruses, ransomware, data modifications, etc.

Besides, CCAM systems face several vulnerabilities:

- Size and topology of transport systems.
- Interdependence between systems.
- Availability of data access and online services in real time.
- Volume of ITS users and operators.

The threats and vulnerabilities previously mentioned point to the next risks:

- System components risks:
  - Loss of operability.
  - Hardware and software damages.
  - Confidence in the information received.
- Business risks:
  - Loss of service.
  - Damages in the corporate image.
  - Non-compliance with regulations.
- Social risks:
  - Service unavailability.
  - Personal damages.
  - Environmental impact.
  - Private and confidential data exposed.

CCAM security issues generate a great number of challenges that need to be addressed. Among them, we can point out the following ones:

- Hardware and digital security integration.
- Suitable investments in cybersecurity.
- Appropriate methodology of security effectiveness measurements.

---

<sup>1</sup> Threat: Potential cause of an incident, causing damage to the systems.

<sup>2</sup> Vulnerability: Weakness of a system that can be exploited by internal or external threats.

<sup>3</sup> Risk: Probability that a threat will materialize by exploiting the vulnerability.

- Construction of data loggers for security attacks and sharing of the gathered information.
- Coexistence of old and new systems in the CCAM infrastructure.
- Operator data exchange methodology and security responsibility.
- New and holistic design approach that consolidates vehicle and transportation safety from the ground up.

## 4 CCAM cybersecurity

The consequences of cyberattacks are not limited to financial losses or damage to the professional image, but can also extend to personal damage. In this sense, it is necessary to ensure the provision of critical measures against cyberattacks. Thus, resilience is one of the cornerstones in the security strategies for protecting critical infrastructures. In this sense, the Spanish National Center for the Protection of Critical Infrastructures (CNPIC) [3] and Ineco [4], an engineering and consultancy Spanish company focused on sustainable mobility and digital transformation, are working on the creation of a certification scheme for the protection of essential infrastructures and services through which the level of security maturity would be observed in all areas of the organization [5]. In this way, the operators could prove compliance with the Protection of Critical Infrastructures (PIC) regulations.

### 4.1 Threats and attack analysis

From a simplified security point of view, there are two potential attack targets: vehicle and road infrastructure elements. In this way, both the exposed interfaces and all the assets associated to the end-points of these interfaces are exposed to security risks. Under these premises, physical attacks, such as sabotage and vandalism attacks, or legal threats (e.g. non-compliance with regulations) are out of the scope of this analysis.

Exposed interfaces could be exploited by another valid user, such as vehicles or RSUs, leading to undesired actions or events due to, for example, design flaws or entities posing as valid users causing malicious actions or providing false or misleading information to other users.

CCAM security threats are classified according to the following security objectives: Confidentiality, integrity, availability, authenticity, and non-repudiation [6, 7].

1. *Confidentiality*: Eavesdropping problems related to secure and private information sharing within the CCAM system fall into this threat category. Confidentiality is an imperative feature in V2X technologies to prevent information disclosing attacks in CCAM communications. The complex CCAM ecosystem makes it challenging to achieve confidentiality in communications between such disparate devices with different computing capabilities.



2. *Integrity*: All the CCAM interfaces are exposed to integrity attacks. A malicious user can intercept messages between two other users and manipulate or distort their content, which should be avoided.
3. *Availability*: Vulnerabilities related to DoS attacks are categorized as availability threats.
4. *Authenticity*: This threat category includes identification issues.
5. *Non-repudiation*: Accountability and non-repudiation system capabilities are a cornerstone in CCAM systems providing protection against malicious actions by users (legitimate or not) within the system.

## 4.2 Cyberattack target examples

There are many elements that can become the target of cyberattacks. A non-exhaustive list divided into three categories is provided below [8]:

- Attacks against applications and systems that directly affect public safety and critical operations: vehicle detection systems, reversible lanes, railway crossing barriers, dynamic message signs, pedestrian detectors, street light controls, etc.
- Attacks against applications and systems that affect daily operations and revenue generation: bus lane cameras, red-light cameras, speed cameras, automated toll collection systems, ticket and parking payments, congestion zone charges, freight truck trolls, etc.
- Attacks against applications and systems that support the previous systems and the organization itself: emissions and air quality sensors, cooperative traffic and position sharing systems, artificial intelligence and machine learning applications, etc.

## 4.3 Cybersecurity methodologies in the CCAM development process

CCAM systems need to use a comprehensive methodology approach and the appropriate models to manage cybersecurity risks, which would allow to implement a unified cybersecurity approach of CAM products and services. In the following paragraphs, a brief summary of the methodologies used so far during the development process of cybersecurity CCAM products is shown.

In 2019, the National Institute of Standards and Technology (NIST) updated the version of the Cyber Security Framework (CSF) [9]. The updated version provides information and tools for the institutions to assess their current cybersecurity state, making it possible to identify security risks and build security improvements tailored to the characteristics and requirements of each network infrastructure. The main functions of the NIST framework are summarized below:

- *Identify assets and resources*. The first function of the NIST framework is to identify assets and resources needed to fulfill the institutional risk management plan.

- *Protect*. Security solutions (authentication tools, policies, protocols, and practices) guarantee the secure management of the identities and credentials for trusted users, devices, and processes.
- *Detect*. Information shortcomings and anomalies are easily identified because the system data is continuously monitored.
- *Respond*. An adequate response to cybersecurity issues must be based on complete analysis to determine the response and restore actions to be taken.
- *Recover*. To ensure the damage recovery after a cybersecurity incident in a short period of time, a recovery process plan including technical and management procedures and tools should be carried out.

Typically, cybersecurity has been included at the end of the development phase instead of taken into account in the design. Therefore, incorporating cybersecurity layers during the design enables strong defense capabilities based on the principle of least privilege cybersecurity [9]. The main cybersecurity layers that should be considered are:

- *Perimeter*. It is the first layer and therefore it secures the interfaces.
- *Network*. Firewall services protect the network layer from external access.
- *End-point security*. The end-point security delivers secure connections between electronic components.
- *Application*. The objective of this layer is to ensure that the applications running in the processor are authorized and trustworthy.
- *Data*. The last layer provides data security. Integrity, confidentiality, and reliability of user data must be guaranteed.

In order to implement a cybersecurity-by-design methodology, there are several tools, such as the Threat Modeling Method (TMM), the Hazard Analysis and Risk Assessment (HARA), and the Threat Analysis and Risk Assessment (TARA) [9].

The TMM analyzes the potential threats or attacks from unknown sources. There are several types of TMM (STRIDE, PASTA, LINDDUN, Attack tree, VAST, SecuriCAD, etc.). It is possible to combine several TMN methods to build a stronger tool to repel a potential risk.

TARA, based on the ISO/IEC 27001 standard, discriminates and identifies security threats. Both HARA and TARA analyses should be combined in the early phases of the design. The safety specifications are usually classified into functional safety requirements and safety integrity requirements, usually through a multi-step method:

1. Safety function identification.
2. Safety function assessment. Based on the concept of Safety Integrity Level (SIL), IEC 61508 shows a classification from most (SIL 4) to least reliable (SIL 1). ISO 26262 is a modification of the SIL classification that contains an Automotive Safety Integrity Level (ASIL) that considers special safety performances.
3. Safety function verification.
4. Failure Mode and Effects Analysis (FMEA).
5. Functional safety audits.

## 5 Cryptography

There are several types of cryptographic algorithms that can be used to build security protocols for vehicular communications. The main type of algorithms are described below [9, 10, 11]:

- Symmetric key encryption (also known as secret key encryption) algorithms: They are based on the usage of one key shared by both ends of the communication channel. These types of algorithms are generally very fast, but the key distribution may be a problem for a large number of users.
- Asymmetric key encryption (also known as public key encryption) algorithms: They are generally slower than symmetric key algorithms, but in comparison the logistics of key distribution are significantly reduced.
- Digital signature algorithms: This type of algorithms are used in order to ensure message authentication and non-repudiation. In practice, they rely on public key algorithms.

### 5.1 Post-quantum cryptography

As it is well known, when using large-scale quantum computers, Shor's algorithms are able to break the security of all public-key algorithms that rely on integer factorization and discrete logarithmic problems, including Diffie-Hellman, ElGamal, RSA, and Elliptic Curve Cryptography (ECC) [12]. This is due to the fact that, if a quantum computer with sufficient computing capacity is developed, the mathematical problems on which the security of such cryptosystems are based (basically, integer factorization and discrete logarithms) could be solved in just a few hours or days at much.

Another algorithm that will affect cryptographic applications is Grover's algorithm, which offers a quadratic speed up of unstructured search problems [13]. Although Grover's algorithm does not threaten to render current cryptographic technologies obsolete, it requires to double the key size for some symmetric key algorithms such as the Advanced Encryption Standard (AES) and hash functions such as the ones belonging to the SHA-2 and SHA-3 families.

In 2016, NIST initiated a process to develop and standardize one or more quantum-resistant public-key algorithms [14]. The process is now at its final stage, where a set of selected algorithms for Public Key Encryption (PKE)/Key Encapsulation Mechanism (PEM) and Digital Signature (DS) functionalities will be selected out of a reduced number of candidates. Those candidates are the following ones:

- PKE/KEM (first tier): Classic McEliece, CRYSTALS-KYBER, NTRU, and SABER.
- PKE/KEM (second tier): BIKE, FrodoKEM, HQC, NTRU Prime, and SIKE.
- DS (first tier): CRYSTALS-DILITHIUM, FALCON, and Rainbow.
- DS (second tier): GeMSS, Picnic, and SPHINCS+.

The candidates that are being evaluated in the still ongoing NIST initiative can be categorized into five areas: lattice-based cryptography, code-based cryptography, hash-based cryptography, multivariate-based cryptography, and elliptic curve isogeny-based cryptography. These techniques are believed to be resistant to attacks performed either by regular or quantum computers. The final selection is expected to be announced at some point in 2022.

## 5.2 Lightweight cryptography

Similarly to the process for choosing a post-quantum algorithm, in 2018 NIST initiated a process to evaluate, and standardize lightweight cryptographic algorithms that are suitable for use in constrained environments where the performance of current NIST cryptographic standards is not acceptable [15].

The reason for initiating a different selection process is that post-quantum algorithms are not expected to fulfill all the requirements needed for constraint devices. If a certain device is able to support a post-quantum algorithm, it should use that algorithm due to its improved capabilities against attacks made by quantum computers. In comparison, those devices that cannot implement post-quantum cryptography should adhere to the new lightweight cryptographic standard.

In 2021, NIST published the list of finalists: ASCON, Elephant, GIFT-COFB, Grain128-AEAD, ISAP, Photon-Beetle, Romulus, Sparkle, TinyJambu, and Xoodyak. It is expected that one or several candidates will be selected as standard solutions along 2022.

## 6 Conclusions

It can be stated that ITS play an essential role in CCAM infrastructures. As any other information system, ITS are composed of many different elements performing tasks such as data collection, processing, and analysis. These elements will allow to implement autonomous mobility in a satisfactory way.

However, in order to achieve that goal, security must be strengthened to the limit, as the CCAM environment faces many threats and risks, both by malicious users and non-malicious errors and bugs.

Cryptography is a vital component of any security algorithm, as it allows to guarantee confidentiality, authentication, and non-repudiation through a mirage of both symmetric and asymmetric algorithms. Two types of cryptography are of special interests for CCAM: post-quantum algorithms, that will provide systems with the much-needed resistance against future quantum computers, and lightweight cryptography, that must be used in devices with constraint capabilities where it is not possible to use post-quantum algorithms.

NIST has initiated two independent processes for selecting sound and reliable algorithms in those two fields. The use of the selected algorithms will be essential for making CCAM a secure reality.

## Acknowledgements

This work was supported in part by the Spanish State Research Agency (AEI) of the Ministry of Science and Innovation (MICINN), project P2QProMeTe (PID2020-112586RB-I00/AEI/10.13039/501100011033), and in part by ORACLE Project, with reference PCI2020-120691-2, funded by MCIN/AEI/10.13039/501100011033, and European Union "NextGenerationEU/PRTR".

## References

1. Ministerio del Interior: Estudio sobre la cibercriminalidad en España. (2020) <http://www.interior.gob.es/web/archivos-y-documentacion/informe-sobre-la-cibercriminalidad-en-espana>. Cited 21 May 2022.
2. ERTICO-ITS Europe: Project AUTOPILOT. (2020) <https://autopilot-project.eu>. Cited 21 May 2022.
3. CNPIC: Centro Nacional de Protección de Infraestructuras Críticas. (2022) <https://cnpic.interior.gob.es/opencms/en/home/>. Cited 21 May 2022.
4. Ineco: About Ineco. (2022) <https://www.ineco.com/webineco/en/about-us>. Cited 21 May 2022.
5. Dauksis, E., García, A., Lozano, L., Pérez, C., Rodríguez, A.: Cybersecurity and transport: protected systems. (2020) <https://www.revistaittransporte.com/cybersecurity-and-transport-protected-systems/>. Cited 21 May 2022.
6. European Telecommunications Standards Institute (ETSI): Security; Threat, Vulnerability and Risk Analysis (TVRA). ETSI TR 102 893. (2017) [https://www.etsi.org/deliver/etsi\\_tr/102800\\_102899/102893/01.02.01\\_60/tr\\_102893v010201p.pdf](https://www.etsi.org/deliver/etsi_tr/102800_102899/102893/01.02.01_60/tr_102893v010201p.pdf). Cited 21 May 2022.
7. Hahn, D., Munir, A., Behzadan, V.: Security and privacy issues in intelligent transportation systems: Classification and challenges. *IEEE Intelligent Transportation Systems Magazine* **13**(1) (2021) 181–196
8. Huq, N., Vosseler, R., Swimmer, M.: Cyberattacks Against Intelligent Transportation Systems. (2017) [https://documents.trendmicro.com/assets/white\\_papers/wp-cyberattacks-against-intelligent-transportation-systems.pdf](https://documents.trendmicro.com/assets/white_papers/wp-cyberattacks-against-intelligent-transportation-systems.pdf). Cited 21 May 2022.
9. Kim, S., Shrestha, R.: *Automotive Cyber Security. Introduction, Challenges and Standardization*. Springer Singapore (2020)
10. Menezes, A.J., Vanstone, S.A., Oorschot, P.C.V.: *Handbook of Applied Cryptography*. CRC Press, Inc. (1996)
11. Institute of Electrical and Electronics Engineers (IEEE): IEEE Trial-Use Standard for Wireless Access in Vehicular Environments – Security Services for Applications and Management Messages. (2006) <https://ieeexplore.ieee.org/servlet/opac?punumber=11000>. Cited 21 May 2022.
12. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing* **26**(5) (1997) 1484–1509
13. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: *Proceedings of the 28<sup>th</sup> annual ACM symposium on Theory of Computing (STOC'96)*. (1996) 212–219
14. National Institute of Standards and Technology (NIST): Post-quantum cryptography. (2022) <https://csrc.nist.gov/Projects/post-quantum-cryptography>. Cited 21 May 2022.
15. National Institute of Standards and Technology (NIST): Lightweight Cryptography. (2022) <https://csrc.nist.gov/Projects/lightweight-cryptography>. Cited 21 May 2022.

# Cryptographic protocols in advanced metering infrastructures in smart grids

Luis Hernández-Álvarez, Juan José Bullón Pérez and Araceli Queiruga-Dios

**Abstract** Smart grids arose from the necessity to define a smart power supply network that provides uninterrupted energy supply to homes and businesses. Renewable energy sources, distributed generation, and distribution are essential features of these networks. Metering systems in smart grids are intelligent electronic devices that measure consumers' energy usage, and send and receive data via two-way communication. With the advent of the Internet of things and industry 4.0 security is vitally important in the exchange of sensible and confidential information. This paper analyzes the security threats in advanced metering infrastructures and presents some cryptographic protocols that guarantee system security.

## 1 Introduction

The global economy is increasingly competitive, with growing demands that cannot be sustainable by conventional energy systems [1]. In the 21st century, there was a considerable increase in the electricity demand mainly due to the dependency of new consumers [2].

---

Luis Hernández-Álvarez

Institute of Physical and Information Technologies (ITEFI), Spanish National Research Council (CSIC), Calle Serrano, 144, Madrid 28006, and Computer Security Lab (COSEC), Universidad Carlos III de Madrid, Avda. de la Universidad 30, 28911 Leganés, Madrid, Spain.  
e-mail: luis.hdez.alvarez@iec.csic.es

Juan José Bullón Pérez

Higher Technical School of Industrial Engineering, Department of Chemical Engineering and Textile, Universidad de Salamanca, Avda. Fernando Ballesteros, 37700 Béjar, Salamanca, Spain.  
e-mail: perbu@usal.es

Araceli Queiruga-Dios

Higher Technical School of Industrial Engineering, Department of Applied Mathematics, Universidad de Salamanca, Avda. Fernando Ballesteros, 37700 Béjar, Salamanca, Spain.  
e-mail: queirugadios@usal.es

The Directive 2012/27/EU of the European Parliament and of the Council of October 25, 2012, about energy efficiency, amending Directives 2009/125/EC and 2010/30/EU, and repealing Directives 2004/8/EC and 2006/32/EC are related to the power supply. According to these Directives, the European Union is facing unprecedented challenges resulting from increased dependence on energy imports and scarce energy resources, and the need to limit climate change and overcome the economic crisis. Energy efficiency is a valuable means to address these challenges. It improves the Union's security of supply by reducing primary energy consumption and decreasing energy imports. Moreover, it helps to reduce greenhouse gas emissions in a cost-effective way and thereby to mitigate climate change.

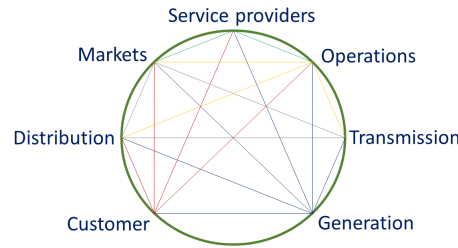
When designing energy efficiency improvement measures, efficiency gains and savings obtained through the widespread application of cost-effective technological innovations such as smart grid (SG) should be taken into account.

The smart grid is a cyber-physical system composed of a network of devices and power infrastructure which aims to monitor, control, and manage users' energy consumption [3]. SG collects and manages big data from several data sources through wireline and wireless communication infrastructures and converts homes into smart environments [4]. They have been improved with computation and communication capabilities to turn them into "smart" and "connected" spaces.

The smart grid connects two communities that "speak" different languages: on the one hand, information and communication technologies (ICT), which is related to the language of computers and networks of managing processes for companies and public services; and on the other hand, operation technology (OT), which is related to electronic devices that have their own operating systems, which serve to support energy supply and operating networks. And despite these differences, ICT companies are rapidly adapting to OT support, so that the operating systems, computing platforms, and communication networks commonly used in ICT are being currently used in some OT architectures [5].

On the other hand, the smart grid integrates electrical technologies (flows of electricity) with ICT (flows of information) in a distributed advanced energy network. Moreover, it supports the two-way flow of electricity and information as the energy and information delivery is not unidirectional anymore. Traditionally, the customers managed their power consumption by controlling their energy demand. With smart grids, consumers have better monitoring of the amount of energy that is produced and delivered to their homes and can manage their own consumption, thanks to the smart equipment, electronic meters, and automated management systems that communicate with each other. Within the smart grid concept, consumers can generate their own energy, which can be returned to the grid [6, 7].

Fig. 1 summarizes the structure and organization of an SG network, which is made of different domains such as bulk generation, energy transmission, and distribution, market, customers, operation, and service provider. The first domains (generation, transmission, distribution, and customer) are connected by a two-way energy flow. Operation, market, and service provider manage them through a two-way information flow [8].



**Fig. 1** Actors involved in smart grids domains through secure communications [9]

The distribution network of electric power starts in power plants and finishes in consumers. The electrical power, generated in power generation plants, is initially distributed as high voltage power and transformed to a medium voltage for feeders at buildings. To be used by consumers in smart home appliances and devices it is transformed to a lower caliber voltage. This network should have some features such as reliability, flexibility, demand adjustment, advanced services, security, performance, etc. [10].

The smart grid two-way distributed delivery network can be categorized into various hierarchical networks [8]:

- The home area network (HAN) is a short-range communications network that connects appliances and other devices in the environment of a house. Apart from lighting systems, these appliances also include entertainment systems and electric vehicles. Moreover, the HAN is equipped with a smart meter (home gateway) that collects energy consumption data.
- The neighborhood area network (NAN) or meter local area network provides the smart meters information exchange between consumers and electricity companies through the Internet.
- The wide area network (WAN) is the highest level network, which covers larger areas, usually integrates several smaller networks, and connects multiple data concentrators and the utility control center. The WAN collects data and information from the NAN and is responsible for transmitting it over long distances to power companies, power distribution stations, substations, distribution networks, and control centers.

The SG operates with smart measurement devices: sensors, networks, phasor measurement units, advanced metering infrastructure (AMI), automated meter reading, and automatic meter management [6].

Advanced metering infrastructures or systems with advanced measurement infrastructure are systems with the ability to measure, record, collect and transfer remotely the information associated with consumption, demand, electrical parameters, and how to use the electrical energy, to present, analyze, manage, and follow a decision-making process. A general AMI system is made up of three main components: smart



meters, communication networks, and the measurement data management (MDM) system [11].

Several security vulnerabilities exist in AMI in smart grids. Customer data confidentiality has not been studied much. However, this is not the case of AMI authentications, where more publications can be found. The AMI aims to collect, measure and analyze energy usage. These devices are composed of software (including MDM software), hardware and communication networks, and customer-associated systems [12].

This work aims to describe and analyze the security characteristics of advanced metering infrastructures and, more specifically, the cryptographic protocols used in AMI that guarantee system security.

The rest of the paper is organized as follows: Smart grids and advanced metering infrastructures are presented in Section 2. Section 3 details several aspects of security in advanced metering infrastructures. In Section 4, the cryptographic protocols to secure AMI are included. Finally, some conclusions are presented in Section 5.

## 2 Smart Grids and Advanced Metering Infrastructures

The European Parliament Directives 2009/72/EC and 2009/73/EC require Member States to ensure the implementation of intelligent metering systems to assist the active participation of consumers in the electricity and gas supply markets. Smart metering systems or intelligent metering systems are an electronic systems that can measure energy consumption, provide more information than a conventional meter, and transmit and receive data using electronic communications.

Member States must implement intelligent metering systems and roll out smart meters for natural gas and/or electricity in accordance with Directives 2009/72/EC and 2009/73/EC, which ensure the security of the smart meters and data communication, and the privacy of final customers, in compliance with relevant Union data protection and privacy legislation.

Distributed energy systems, including local renewable sources and energy storage, will continue to develop by leveraging technological advances [5].

The underlying SG technology consists of bidirectional communication, automatic metering systems, renewable energy sources, distribution, monitoring, and control of the entire power grid [6].

One of the key elements of advanced metering infrastructures is the smart meter, which has integrated two-way communication capabilities to remotely transfer information to data processing systems with the aim of remote monitoring and billing purposes. Consumer portals are responsible for processing the AMI data in order to provide more intelligent energy consumption. Moreover, they can provide interactive services like prepayment [13]. Fig. 2 shows the scheme of an advanced metering infrastructure.

In addition, AMI systems act as data collectors that perform data gathering from smart meters, and transfer them through HAN, NAN, and WAN, using different

communication tools usually implemented by wireless technologies such as radio frequency, microwave, Ethernet, Wi-Fi, ZigBee, Bluetooth, etc. This communication is established between the AMI and the MDMS, located in the offices of the company that provides the electrical distribution service, and is responsible for information processing for billing and quality monitoring [11, 14]. To support two-way communication, the AMI networks use the Internet protocol in such a way that an IP address is assigned to gateways, meters, and smart appliances [15].

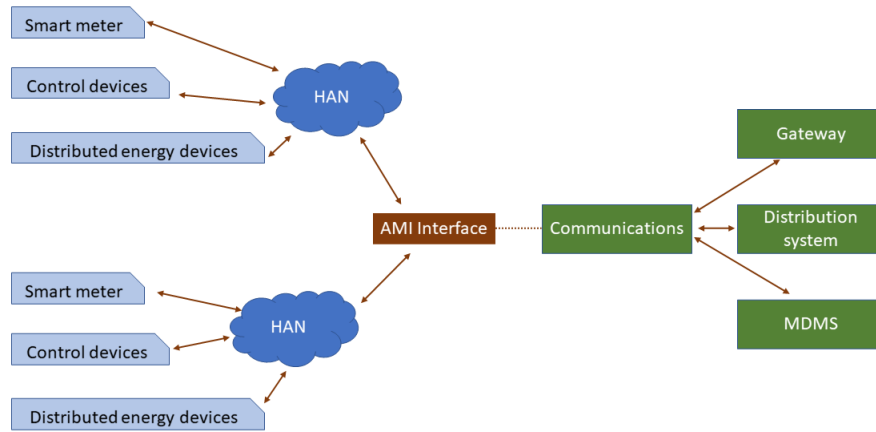
### 3 Security in Advanced Metering Infrastructures

Currently, the world is strongly interconnected, so cybersecurity is required to help protect devices from cyber-attacks or possible unexpected damage, as well as to maintain proper operation of the power supply to electricity during these events.

Cybersecurity must be designed not to interfere with the supply of electrical energy. It must be able to adapt to obsolete equipment with computational and communications bandwidth limitations, as well as to control and protect equipment that has been widely distributed in remote unmanned substations, both on top of poles and in accessible areas. To achieve the vision of the smart grid, real-time administration, management, monitoring, protection, and control of the energy supply are required, in addition to the security and protection of the data collected.

Particular considerations on security and privacy issues in AMI were proposed in [16]. Strong device authentication is needed before a smart meter joins the AMI network.

To deal with potential security threats in AMI systems, the following security requirements are considered [2, 17]:



**Fig. 2** AMI diagram with the different actors [13]

1. Data confidentiality: All data packets exchanged in the network including meter readings and control messages must be kept confidential such that only authorized entities with corresponding credentials are allowed to access specific sets of data. This requirement includes privacy protection of customers' information about power consumption.
2. Data integrity and authenticity: When information arrives at its destination, the recipient should be able to verify whether the message remains unaltered and indeed comes from the sender it claims. Data integrity applies to the transmitted information (including control commands) from the meter to the different systems of the network as well as from these to the meter.
3. Device authentication: Any smart meter's identity must be securely authenticated before it can join the AMI network and exchange data with other devices.
4. Data availability: Although some data is not critical, some others must be collected every minute. Component failure, software problems, human tampering with the meter or communication failure can cause unavailability of data.
5. Non-repudiation: It is known in AMI as accountability or non-denial, i.e., the devices that receive data will not deny receiving it and vice versa. This is a concern because in power distribution several vendors are in charge of manufacturing processes.

Security is fundamental in smart grid communications. The authors of [12] developed an in-network collaborative communication scheme for secure and reliable AMI communications but privacy issues are not mentioned in the context. In [18], the authors assigned two identifications (ID) to each smart meter and made the ID for high-frequency sensitive data anonymous through a third-party escrow service. In [19], the authors tried to make the appliance load signature undetectable using a rechargeable battery to mitigate any significant change in real-time power consumption.

A secure information aggregation method using homomorphic encryption is proposed in [20]. They suggested a secure and privacy-preserving AMI communication scheme including the idea of aggregation but with enhanced security considerations. The first step was to build an initial device registration procedure with strong authentication requirements to make sure unauthorized devices cannot join the network. Homomorphic encryption is adopted on sensitive metering data to ensure data confidentiality and meanwhile preserve customer privacy. A digital signature to each message was also attached for the integrity and authenticity of the metering data.

Several authors have developed studies about key establishment protocols but their implementation in AMI was not possible due to their high complexity. In the case of [21], the authentication of a smart meter in a HAN was made through an identity-based protocol, which included several encryption operations (both symmetric and asymmetric) and this increases the computational load considerably, i.e., the computation overhead is high. The computation overhead is defined as the time required to encrypt the data in the origin in the correct format to be transmitted or the time required to decrypt the received encrypted data in the destiny to obtain the corresponding plain data [15].

Intrusion detection systems, firewalls with access control list, network and system management, or public key infrastructure are some security technologies that protect and secure an AMI. The insecure location of smart meters, low bandwidth, and public communication protocols, or the need for other systems for a proper operation are some drawbacks in AMI functioning [2].

As with other networks, AMI can be used as the entry point of malware and computer viruses to damage the functionality of the complete smart grid. Smart meters can act as an attack vector and cause a denial of service (DoS) attack. Moreover, every node in the AMI network can be attacked and converted into a puppet node and thus be controlled by the attacker. This node acts as an infected node, i.e., it receives data and sends infected packets to the network, and thus, it could damage the AMI network [8].

#### **4 Cryptographic Protocols to Secure Advanced Metering Infrastructures**

Several public and private key cryptographic algorithms were used to secure smart grids. Authentication and key agreement protocols are usually focused on one-way communications. However, communications in advanced metering infrastructures need two-way authentication. To guarantee a secure scenario in a smart metering infrastructure, the following goals must be satisfied [22]:

1. Mutual authentication and key establishment between the NAN gateway and the smart meter. This mutual authentication will secure consumption data.
2. Confidentiality in the information that is transmitted to legal entities.
3. Protocol's message integrity during the communication process.
4. Information availability despite the computation that is required for authentication.
5. Forward security, which leads to independent sessions, without mixing data from the same user.
6. Anonymity in the data transmitted to the utility for billing purposes. This will avoid an adversary spoofing the identities of smart meters that send information from the HAN to the NAN.

Abood et al. [23] compared Advanced Encryption Standard (AES), Data Encryption Standard (DES), Triple DES, Educational DES, RSA, and BLOWFISH to find the best option with regard to key size, algorithm effectiveness, computational complexity, and execution time. They found that the AES protocol was the most efficient. In fact, some studies about time consumption showed public key encryption and decryption processes are secure, but time consuming, which results in non-efficient cryptography [22].

In [24], an identity-based key establishment protocol, based on elliptic curve cryptography (ECC), was proposed. ECC provides the same security level as other secure asymmetric protocols, such as RSA, DSA, and DH, but with the advantage

of using much fewer key sizes. A different ECC-based lightweight scheme was proposed in [25]. This authentication protocol provides security against all known attacks. Moreover, it is characterized by low computation and communication costs.

AMI networks include three main components: smart meters, gateways, and the utility computer. As resource-constrained systems, smart meters and gateways, have limited storage capabilities, so they will not store large amounts of information. In the case of including cryptography in these devices, keys and some other parameters must be stored in smart meters. In the case of RSA values of  $p$ ,  $q$ ,  $n = p \cdot q$  and Euler indicator,  $\phi$ , must be stored and keys and encryption and decryption parameters in the case of ECC-based schemes.

A hybrid protocol with AES cryptosystem and elliptic curve integrated encryption scheme (ECIES) was proposed in [15]. This system provided data integrity, confidentiality, and authenticity, and thus it was resistant to false data injection and message reply attacks [26]. However, the computation overhead is high in the AMI network because of the multiplications required by the ECIES and the large number of smart meters in the NAN.

Due to the resource-constrained nature of the SG, a lightweight authentication scheme was proposed in [10]. This protocol uses symmetric (AES) and asymmetric (RSA) cryptography for session key generation and provides authentication, guarantees message integrity and the computational efficiency is high. This proposal aims to secure HAN and NAN gateways avoiding attacks such as the interception of messages between the BAN and the HAN by an adversary (reply attack), the sending of false messages to the BAN gateway (false message injection), eavesdropping of the HAN messages (message analysis attack), and interception and modification of HAN messages (message modification attack) [15].

ECC-based schemes were used for anonymous authentication and key agreement. The problem with such algorithm is that it includes time-consuming operations such as bilinear pairings. The smart grid is an energy-limited system, so it is a challenge to achieve privacy protection with low computational cost [27].

Kumar et al. proposed an ECC-based lightweight authentication and key agreement scheme called LAKA. This protocol supports bilateral authentication, providing integrity and anonymity in the AMI communication. These authors proved the performance and semantic security of LAKA [22].

A different lightweight authentication protocol was proposed for smart metering infrastructure in [28]. This scheme used the fully hashed Menezes-Qu-Vanstone key agreement protocol, with ECC and one-way hash functions. In this proposal, the smart meters validate the authenticity of the NAN gateway and this gateway also validates the customer. When mutual authentication is successfully provided, both entities shared a session key for subsequent contacts. Moreover, this protocol resists several attacks such as DoS, replay, impersonation attacks, etc.

The NIST launched a call for a standard lightweight cryptographic protocol to select Authenticated Encryption with Associated Data and hashing schemes [29]. This challenge has not finished yet. There are currently 10 finalists that proposed their own algorithm for constraint devices [30, 31]. Several authors performed different

analyses of the NIST candidates including efficiency, performance, computational cost, and security [32].

However, a different consideration must be taken into account to achieve secure devices. Another challenge is open to provide a quantum-resistant protocol to secure against both quantum and classical computers [33]. Currently, this process is not closed and there is no final decision about the final post-quantum standard. The third round finalist are *Classic McEliece*, *CRYSTALS–KYBER*, *NTRU*, and *SABER*, although alternate candidates like *FrodoKEM* could also be considered. Nevertheless, these protocols might be too complex for constrained systems like SG. In any case, if the selected standard is not suitable for SG networks, then a lightweight cryptographic algorithm should be used.

## 5 Conclusions

Advanced metering infrastructures in smart grids integrate a two-way communication network which enables customers and utilities to actively monitor and manage their energy use. A secure AMI system can use an aggregate protocol, as suggested in [34] for smart grids. A set of AMI  $A = \{A_1, A_2, \dots, A_n\}$  or network nodes collect data and send it to a sink node that can act as a control center that provides keys to each AMI.

The smart grid is a constrained system, so it is a challenge to achieve security with low computational cost. This paper includes a review of cryptographic protocols proposed for secure AMI. Due to its energy-limited features, lightweight cryptography seems to be the best solution. However, after achieving a consensus of the NIST processes (lightweight and post-quantum cryptography), the results must be analyzed to assure smart grid devices' security with a compatible protocol.

**Acknowledgements** This work was supported in part by the Spanish State Research Agency (AEI) of the Ministry of Science and Innovation (MICINN), project P2QProMeTe (PID2020-112586RB-I00/AEI/10.13039/501100011033) and by the Fundación Samuel Solórzano Barruso under grant no. FS/26-2020. Luis H-Á thanks CSIC Project 202050E304 (CASDiM) for its support.

## References

1. F. Caputo, B. Buhnova, L. Walletzký, *Sustainability Science* **13**(5), 1299 (2018)
2. R.R. Mohassel, A. Fung, F. Mohammadi, K. Raahemifar, *International Journal of Electrical Power & Energy Systems* **63**, 473 (2014)
3. P. McDaniel, S. McLaughlin, *IEEE Security & Privacy* **7**(3), 75 (2009)
4. Y. Kabalci, *Renewable and Sustainable Energy Reviews* **57**, 302 (2016)
5. T. Ackermann, G. Andersson, L. Söder, *Electric power systems research* **57**(3), 195 (2001)
6. L. Chhaya, P. Sharma, G. Bhagwatikar, A. Kumar, *Electronics* **6**(1), 5 (2017)
7. X. Fang, S. Misra, G. Xue, D. Yang, *IEEE communications surveys & tutorials* **14**(4), 944 (2011)

8. P. Kumar, Y. Lin, G. Bai, A. Paverd, J.S. Dong, A. Martin, *IEEE Communications Surveys & Tutorials* **21**(3), 2886 (2019)
9. C. Greer, D.A. Wollman, D.E. Prochaska, P.A. Boynton, J.A. Mazer, C.T. Nguyen, G.J. Fitz-Patrick, T.L. Nelson, G.H. Koepke, A.R. Hefner Jr, et al., NIST framework and roadmap for smart grid interoperability standards, release 3.0. Tech. rep. (2014)
10. K. Mahmood, S.A. Chaudhry, H. Naqvi, T. Shon, H.F. Ahmad, *Computers & Electrical Engineering* **52**, 114 (2016)
11. J. Gómez, R. Castán, J. Montero, J. Meneses, J. García, *Boletín IIE* **39**(4), 180 (2015)
12. Y. Yan, Y. Qian, H. Sharif, in *2011 IEEE wireless communications and networking conference* (IEEE, 2011), pp. 909–914
13. N.M.G. Strategy, US Department of Energy Office of Electricity and Energy Reliability (2008)
14. J. Garcia-Hernandez, in *2015 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)* (IEEE, 2015), pp. 251–256
15. S. Khasawneh, M. Kadoch, *Mobile Networks and Applications* **23**(4), 982 (2018)
16. P. Deng, L. Yang, in *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)* (IEEE, 2012), pp. 1–5
17. M. Salpekar, in *2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2018 2nd International Conference on* (IEEE, 2018), pp. 22–26
18. C. Efthymiou, G. Kalogridis, in *2010 first IEEE international conference on smart grid communications* (IEEE, 2010), pp. 238–243
19. G. Kalogridis, C. Efthymiou, S.Z. Denic, T.A. Lewis, R. Cepeda, in *2010 First IEEE International Conference on Smart Grid Communications* (IEEE, 2010), pp. 232–237
20. F. Li, B. Luo, P. Liu, in *2010 first IEEE international conference on smart grid communications* (IEEE, 2010), pp. 327–332
21. H. Nicanfar, P. Jokar, K. Beznosov, V.C. Leung, *IEEE systems journal* **8**(2), 629 (2013)
22. P. Kumar, A. Gurtov, M. Sain, A. Martin, P.H. Ha, *IEEE Transactions on Smart Grid* **10**(4), 4349 (2019)
23. O.G. Abood, M.A. Elsadd, S.K. Guirguis, in *2017 Nineteenth International Middle East Power Systems Conference (MEPCON)* (2017), pp. 644–649. DOI 10.1109/MEPCON.2017.8301249
24. A. Mohammadali, M.S. Haghighi, M.H. Tadayon, A. Mohammadi-Nodooshan, *IEEE Transactions on Smart Grid* **9**(4), 2834 (2018)
25. K. Mahmood, S.A. Chaudhry, H. Naqvi, S. Kumari, X. Li, A.K. Sangaiah, *Future Generation Computer Systems* **81**, 557 (2018)
26. V. Gayoso Martínez, F. Hernández Álvarez, L. Hernández Encinas, C. Sánchez Ávila, (2011)
27. L. Zhang, L. Zhao, S. Yin, C.H. Chi, R. Liu, Y. Zhang, *Future Generation Computer Systems* **100**, 770 (2019)
28. S. Garg, K. Kaur, G. Kaddoum, J.J. Rodrigues, M. Guizani, *IEEE Transactions on Industrial Informatics* **16**(5), 3548 (2020)
29. NIST. Lightweight cryptography. On-line publication (2015). <https://csrc.nist.gov/projects/lightweight-cryptography>
30. NIST. Lightweight cryptography, finalists. On-line publication (2021). <https://csrc.nist.gov/Projects/lightweight-cryptography/finalists>
31. M.S. Turan, K. McKay, D. Chang, C. Calik, L. Bassham, J. Kang, J. Kelsey, et al., Status report on the second round of the nist lightweight cryptography standardization process. Tech. rep., Tech. rep. <https://doi.org/10.6028/NIST.IR.8369>. Gaithersburg, MD, USA (2021)
32. V.A. Thakor, M.A. Razzaque, M.R. Khandaker, *IEEE Access* **9**, 28177 (2021)
33. L. Chen, L. Chen, S. Jordan, Y.K. Liu, D. Moody, R. Peralta, R. Perlner, D. Smith-Tone, *Report on post-quantum cryptography*, vol. 12 (US Department of Commerce, National Institute of Standards and Technology . . . , 2016)
34. R. Lu, X. Lin, Z. Shi, X. Shen, in *2013 IEEE Wireless Communications and Networking Conference (WCNC)* (IEEE, 2013), pp. 1819–1824

## **Special Session on Cybersecurity and Trusted Supply Chains of ICT**



# Orchestrator Architecture and Communication Methodology for Flexible Event Driven Message Based Communication

Rudolf Erdei<sup>1</sup>, Daniela Delinschi<sup>1</sup>, Emil Paşca<sup>1</sup>, and Oliviu Matei<sup>1</sup>

**Abstract** This article presents the architecture, design and validation of an orchestration approach, that improves the flexibility of Service based platforms. Improving user experience and interaction, for time-critical applications are aspects that were primary objectives for the design of the architecture. Each Service can provide its own embedded User Interface component, also decentralizing the User Interface and, in consequence, improving the loosely coupled approach to the architecture. Obtained results are promising, with a 97% behavior score. Further research is proposed for improving the results and raising the final Technology Readiness Level of the system. These results make the approach a viable alternative to classical service composers.

## 1 Introduction

Microservices have come a long way within the last years, making them a viable alternative to large monolithic systems, or platforms that require flexibility and development speed. Microservices are a specific and improved instance of *Service Oriented Architectures* (SOA). In contrast with monolithic applications, they have several advantages, like optimization of resources, flexibility, the ability to integrate *Serverless Components* [14, 12, 13], or even mix different communication approaches, like REST API's and JMS based Services. Of course, when using microservices, several aspects need to be defined and standardized for the communication to be effective and robust. Integrating these services is traditionally done via service composers or service consumers, components that can have a large codebase and can result in large UI components.

When using Microservices, *Service Composition*, *Service Discovery* and even *Service Interoperability* have to be considered, in order to develop a working flexible

---

Technical University of Cluj Napoca, North University Centre of Baia Mare, Romania; e-mail: rudolf.erdei@cunbm.utcluj.ro

system. Thus, framework components have critical importance. *Service Composers* (SC) also tend to develop into large monolithic components, as they need to integrate several functionalities, as well as accommodate *User Interface and Interaction* (UII). *System State* is also an important part, as SC's need to understand workflows, workflow position and state while also accommodate several simple functionalities that do not justify the existence of another Service (like sending mail or logging). In the end, the SC's can potentially end up being themselves large monolithic components that introduce limitations in the system and are hard to modify, extend or update.

This article discusses a standardization proposal for *asynchronous messaging* inside a security assessment platform [11], implemented within the **Horizon 2020 BIECO Project**<sup>1</sup> that needs almost real-time communication and high throughput, with the use of Orchestration and light UI components instead of a single SC.

The remainder of the article is structured as follows: the next section (Section 2) addresses the latest advances in the field of Microservices Orchestration; Section 3 addresses the principles used for communication; Section 4 addresses the states of the microservices used in the architecture; Section 5 presents the proposed architecture; Section 6 discusses the advantages/disadvantages of the proposed approach; Section 7 presents conclusions and the next steps in the development of the proposed approach.

## 2 Related Work

Software Orchestrators are an important part of distributed system designs. Centralized orchestration, while having some disadvantages, does introduce a level of abstraction and security that the Internet lacks. Zaalouk et al. [20] present an orchestrator architecture for Software Defined Networking (SDN) that can respond to software attacks, raising the security of the resulting network. Another similar approach is presented by Jaeger [8] in a configurable Security Orchestrator approach that can be dynamically configured.

*Distributed Microservices Communication* can also be addressed in the context of *Fog Computing* [19]. Orchestrators also play a key role in this kind of system, enabling *Service Composition*, that provide a holistic approach to delivering functionality to the end-users. Brito et al. [4], Davoli et al. [3] and Borsatti et al. [1] have proposed architectures that address some key aspects like awareness, security, compositing, either defining a service component that implements all functionalities, or defining a service layer with different components for each functionality.

Orchestrating IoT devices and services is a challenge on its own, as the IoT ecosystem has several hard limitations that cannot be pushed. Wen et al. [18] discuss about these limitations and issues, proposing a Fog Orchestrator that aims to address them, while also proposing future research trends for this specific area.

---

<sup>1</sup> <https://www.bieco.org>

*Scalability*, *Interoperability*, and *Reliability* are terms frequently discussed when it comes to orchestrating microservices. The need for an efficient way to handle these aspects, especially in the context of IoT and edge computing applications is discussed by Shi et al. [17] in their paper on the challenges of Edge Computing. To address these problems Jamborsalamati et al. [9] proposed an Autonomous Resource Allocation system with a hierarchical architecture, and also local and global communication layers based on MQTT and HTTP TCP/IP protocols for cloud interactions. Another Publish-Subscribe architecture is also proposed by Saif et al. [16] who proposed a HTTP/3 (H3) solution that exploits the wide-ranging improvements made over H2 and takes better advantage of *QUIC* transport than an *MQTT* mapping would.

To reduce development costs and also provide a centralized solution for security systems based on Service Oriented Architecture (SOA), Goutam et al. [7] proposed an architectural framework called *Orchestrator Model for System Security* (OSS) which is based on three components: Services, Security Services and an Orchestrator.

### 3 Communication principles in Microservices-based Architectures

When designing a microservice communication system, some important aspects need to be considered [10]: *Types of services and communication modes*, *Events and Data types that will flow through the system*, *Platform performance, scalability, flexibility and extensibility*, *Medium of choice and standards used for communication*, *Mitigation measures in case of component failures*, *Data Integrity*, *Safety and Security*.

#### 3.1 Communication between microservices

The proposed orchestration model should know how to handle these the most common standard microservices features such as: **Logging** - in order to provide traceability; **Health checks** - continuous assessment of the system state; **Configuration** - easy and flexible setup for the Services; **User and Service Authentication and Authorization**; **Job Execution Scripts** - known internally as *Methodologies*;

From an architectural point of view, the orchestrator should have the following characteristics: **Fault tolerance / Fault resilience** - the ability of the system to survive and treat errors introduced by Services; **Modularity** - the ability to load only the required functionality; **Performance** - minimizing the execution time and maximizing throughput; **Reliability** - the ability of the service to provide good results consistently;

## 3.2 Security

Handling sensitive information is a serious aspect. Enabling encryption in transit and in REST is done using standard algorithms, TLS and SSL certificates, that encrypt all of the communication that is done via the Internet. Message validators make sure that only allowed messages are permitted inside the system, while Services that try to emit illegal messages will be banned. An hourly backup strategy will be defined and put in place after the final deployment of the system.

### 3.2.1 Calling microservices from a Security perspective.

In a Loosely Coupled Orchestrator based architecture, Services are not able to communicate directly. The Orchestrator acts as a facilitator of communication, understanding the source of the message and knowing the destination at each point of the Methodology Step. Dynamic Communication Mapping is used, defining sources and destinations that can be modified on demand. Also, a Message Mapping is able to determine who can emit what message types and what happens when this limitation is broken.

Securing the network is limited in scope, as some of the communication is done over the Internet. This implies that adding multiple layers of defense within the Orchestrator is in this case the only available choice, limiting as much as possible the possibility of malicious access. Attack detection is an important aspect and must be done in several layers, including the DDoS filtering and actively blocking malicious connections.

As most of the connected services will be asynchronous and execution time may vary, an application level **Event Based Reaction Loop** [2, 6] needs to be defined and implemented, in order to properly respond to any requests and events within the system.

### 3.2.2 Actions and Reactions.

Inside the system, **Reactions** are defined as specific operations that the platform can perform, extensible and pluggable. Potential **Actions** can be defined, depending on what requirement the platform has, and one or more **Reactions** can be assigned to each potential **Action**.

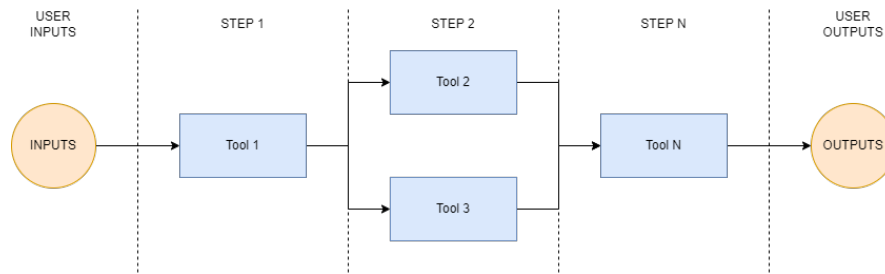
The **BIECO Platform**[15] is a distributed system, with a queue based message communication performed through `HTTP POST` requests. Each software component can be regarded as a *Service* (internally called a `Tool`) that is integrated inside the platform via a Central Coordination Component, called the *BIECO Orchestrator*.

The Orchestrator has several roles inside the platform, roles that are crucial in the correct functioning of the system in its entirety. From these roles, we can mention:

- Abstracting the communication between the Services, inside a Loosely Coupled Architecture;
- Defining Message types and behavior of the System for each type of Message;
- Liveliness Assessment (through a Heartbeat mechanism) for all the connected components;
- Assessment of the system's ability to execute a specific task (internally called a Job);
- Flexibility in integrations, interactions as well as definition of execution methodologies;
- System State Assessment, for understanding the current state of the system.

### 3.2.3 Execution Methodologies

In the execution phase, a series of Steps were defined that express the exact interaction between the Services, each of their inputs and outputs. The Orchestrator will have to know at each moment what Services are executing and their current status, in order to be able to understand what actions are to be executed next.



**Fig. 1** Example of a series of Steps

In this regard, each abstraction of the execution methodology (defined in the Methodology class) will be provided with a series of steps that are to be executed sequentially. Each Step defines all the information necessary for the Orchestrator to be able to understand what is to be done. This way, the Orchestrator will simply be a State Machine, interpreting and executing a previously defined Methodology.

As defined in Figure 1, each Step defines a series of inputs, that will be sent to the used Services inside the Step. After the Services have finished execution, their outputs will be saved as inputs for the next step. Also, Steps define the used Services as well as the current status for them.

This Methodology, regarded as a whole, defines all the Services that will be used, as well as their interactions. Some Message classes have been defined, messages that are treated differently depending on the context and the Services involved. This behaviour is controlled and defined in an optimized ontology, developed by Delinschi et al. [5].

### 3.3 Communication stages

In the *BIECO Communication Paradigm*, interpreting messages is regarded as a computing intensive task, as the Orchestrator has to continually assess the state of the System and load message related information from the DBMS. This is the main reason why the *Event Loop* was broken into three stages, each with its own important role:

#### 3.3.1 Stage 1. Quick Response:

The first stage of the Event Loop is the *Quick Response*. This is part of the mitigation strategy in case of component and/or communication failure. The Quick Response defines the system's behavior in the case of Exceptions, like communication errors or Services that fail. The Quick Response action is the first to run after each message is received. Also, Methodologies can define at each execution step a series of data transfer flows that will also be handled by the Quick Response, in order to improve throughput and lower the response time. Thus, this stage defines the *Extreme Importance Messages*, that need real-time response from the system.

#### 3.3.2 Methodology Execution

The Methodology Execution Plan defines the next steps that will be executed at each specific moment, when running a specific Job. Specific messages, like status messages and returning results have been assigned to Services, messages that update the Service's status inside the Methodology Step, as well as transfer results that need to be forwarded in the next Step to other Services. There are however types of messages, namely Data and Event, that are forwarded without updating the Service's status or changing the Step. This stage defines the *High Importance Messages*, that need priority interpretation and execution, without the need for real-time responses.

#### 3.3.3 Routing Engine

The Routing Engine will be the last communication stage. An extensible Plugin System will be designed for the Orchestrator, that will enable further development without the complex CI/CD pipeline, specific to high-risk components, and without the need to recompile the codebase of the Orchestrator. This behavior is defined in the *Reaction Interface* that allows the creation of extensions for the Orchestrator, that will be loaded and executed on demand. The mapping for Actions and their corresponding Reactions (one or more) will also be defined via an easily

modifiable medium, like a JSON String that can either be saved in the database, or simply in a configuration file.

## 4 Asynchronous Service State Handling

Using *Async Service Access* has the benefit of being non-blocking for the entire BIECO Platform. Services will be able to run their jobs in parallel, for an unlimited amount of time and the communication will be limited to only the required message exchange, improving the bandwidth requirements and the throughput.

In order for this type of communication to perform in a resilient manner, the *State* of each Service must be known at all times. For this, several communication and assessment methodologies can be deployed:

1. A constant, time based query for all the Services, where each Service has to send statuses for all the running jobs;
2. Deploying a time-based query for each running job, that assesses the state of the *in-use* Services;
3. A reactive query, that requests status updates from Services only when a Service has sent it's own status update.

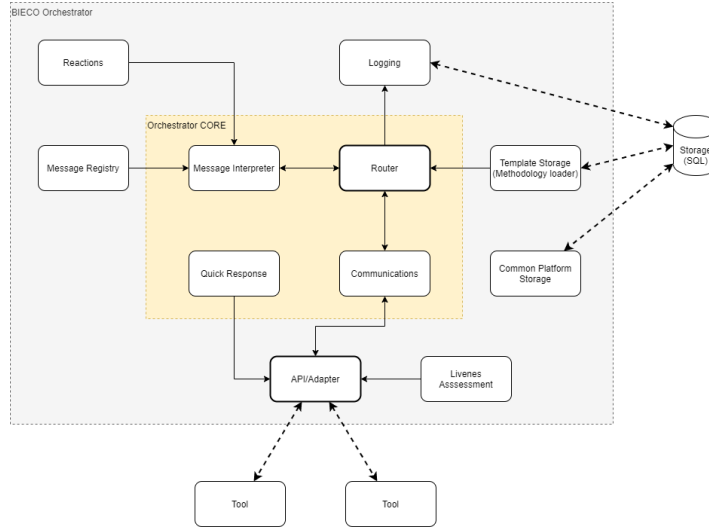
Each variant has several advantages and disadvantages. The main disadvantage of 1 and 2 variants is that the time base for the query can be difficult to estimate in order for the platform to behave correctly. If the time base is low, it may not detect status changes in time. If it is too high, the required bandwidth increases and Platform throughput decreases, introducing instability and unreliability.

Having taken into consideration all involved factors, including the need for some Services to run for an extended amount of time, the third option was considered as the best method to use, despite the fact that it will not be able to detect system problems in real-time.

## 5 Orchestrator Architecture

The architecture for the BIECO Orchestrator is presented in Figure 2. The main module is the Core, that will interpret and execute the needed actions for each received Message, depending on the context that this Message is received. The main components are:

- **Communications Module:** will receive and send Messages from/to any available/required communication channel. This module will receive a valid Message from a valid Service and send it up the processing chain;
- **Router Module:** will load and execute the current running Methodology, if it is loaded. Depending on the received Message, this could mean forwarding it,



**Fig. 2** The Architecture for the BIECO Orchestrator

saving it in the results or performing other actions. The Router will also use the Message Interpreter in order to perform Reactions;

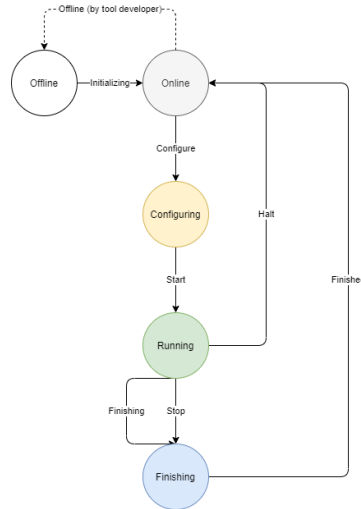
- **Message Interpreter Module:** will be called for specific Message Classes that have an added Action linked to them. An Action  $\Rightarrow$  Reaction map is defined inside the Platform, dynamically linking requests with parts of executable code.
- **Quick Response Module:** will respond to certain Messages (like Exceptions in the System) in the least time possible. This module is the most critical one in the System, as it has to protect the integrity of the running job, the running Services and in some cases even the integrity of the connected simulated or real environment. This is the reason why an inflexible hard-coded approach was used at the design and implementation of this module.

Implementation approaches were selected for each module in particular, accounting for the degree of importance that the specific module has. As stated, the Quick Response Module was implemented using hard coded logic, without complex structures and decisions, in order to keep the code as quick as possible. The methodology execution is guided by a single Methodology object, that is cached in memory for quicker access. The slowest part is the Message Interpreter, a part that is not critical, so the implementation could be made using a plugin-based approach.



## 5.1 Microservices Lifecycle in BIECO

For the Services that are integrated in the Platform, a lifecycle was proposed and implemented, that has two main working states and the transitions between them, as depicted in figure 3. The *Offline* state, while important, will not be discussed here as it is self-explanatory. While in the *Online* state, the services receive regular Heartbeats, with a frequency of 30 seconds.



**Fig. 3** The Generalized Lifecycle of a BIECO Tool

The other two important states are as follows:

1. **Configuring State:** is when the Service is doing its setup phase, in order to gather the required input for working. Inputs can come from user interaction, user input fields or outputs from other Services. In this state, the Service can check the provided information for correctness, internal validation and so on, without any time limitation. If everything is correct for the Service, it will confirm this tot the Orchestrator with a Ready Message.
2. **Running State:** after the Configuring is done, the Orchestrator can send the Start Message to all the Services at once. In the Running state, the Service can perform its task as normal, either sending back Data and Events, or sending the Results after finishing.

The Configuring state has a special importance in the context of some Services that interact with eachother or have certain time constraints that prevent them from performing if they are un-synchronized. In this context, the configuration of the Service and the normal execution have to be separated, to enable the correct synchronization. To achieve this, Services will announce the finalization of their Configuring state with a Ready Message. After all the Services are Ready, the Orchestrator

can send a Start Message in order to transit each one in the normal execution state. This approach enables near-ideal synchronization (depending on the network state and other external factors).

Of course, these states can be broken down internally by the Services. They do not have to implement all of them, depending on the type of Service and what its provided functionality is. Some of them can and will be used in parallel, meaning that the running phase of the Service has to be synchronized with all of the others that are executing in the same step of the Methodology.

## 6 Advantages and Disadvantages of this approach

The presented approach has some advantages that make it suitable for the use-case:

- Light Orchestrator and UI components, making the Platform easy to extend and modify;
- The possibility to use third party API's as Services, if the other Services that interact with it can understand the API;
- The possibility to accommodate both Methodology Workflows (chaining of Services) as well as full-duplex communication between defined Services at each step of the Methodology;
- Platform Safety and Security assessment, to ensure integrity of software and hardware components;

Of course, there are also some disadvantages:

- If complex user interaction is necessary, services will have to implement their own UI's and ask the Platform to display it to the user;
- Services will have to implement the communication methodology and the generic lifecycle;
- Services will have to know and trust the Orchestrator;
- Custom login methods for Services are not possible.

When comparing our platform with others, like the approaches of Zaalouk et al. [20], Jaeger [8], and others (discussed in Section 2), some important key differences can be noticed:

- Enhanced Loose-Coupling, as services can be used either stand-alone or integrated withing the platform (if they provide a UI);
- Ability to design UI components for all services, that can be easily integrated;
- Integration of service states, for more complex interactions between services themselves but also with the platform.

Using the REST HTTP standard for requests, has its benefit as being easily adaptable and also used in all major programming languages. Standard errors, caching, proxies and payloads can have formats as JSON, XML or other. HTTP also supports *Asynchronous Communication*, an important characteristic that is needed by some of the Services used in the BIECO Project.

## 7 Conclusions

The article proposes a novel orchestration methodology and architecture for Asynchronous Communication in a scalable and secure distributed platform.

The presented Communication Methodology and Orchestrator Architecture were tested in a laboratory environment that included 10 microservices. The interaction involved a 5 Steps Methodology with 1 to 5 Services running in parallel at each Step. Correct functionality was obtained, making it a candidate for demonstration and validation in a more challenging environment, closer to the final deployment environment.

Validating the proposed system was done using more complex Services, with variable access modes (ranging from local network to TCP based internet access) and more complex behavior (including UI components developed within Services and also Services that output large amounts of data), where the system obtained a behavior score of 97%, successfully addressing the vast majority of situations, resulting in a TRL of 5. Shortcomings were observed when Services would emit data at wrong moments, so the Methodology Driven Orchestration Service would not be able to understand what the Emitter meant to happen. Other observed shortcoming was addressing a surge of data, when more Services would emit many messages with large payloads in a very small amount of time, overloading the network and the available memory on the virtual machine.

Further research and development will be done in the area of optimizing the responses that the system provides, the behavioral modifications that involve invalid messages from the Services, in order to increase resilience of the system, improving the range of solutions it can address, so that it will result, at the end of the project, in a TRL of 7. Further improvement will also require better integration with the User Interface component of the Platform, that will enable platform users to better interact and understand the current state.

**Acknowledgment.** This work was supported by the project BIECO ([www.bieco.org](http://www.bieco.org)) that received funding from the European Union's Horizon 2020 Research and Innovation programme under grant agreement No. 952702.

## References

1. Davide Borsatti, Mario Valieri, Daniele Tarchi, and Carla Raffaelli. A fog computing orchestrator architecture with service model awareness. 2021.
2. Binildas Christudas. *Practical Microservices Architectural Patterns: Event-Based Java Microservices with Spring Boot and Spring Cloud*. Apress, 2019.
3. Gianluca Davoli, Walter Cerroni, Davide Borsatti, Mario Valieri, Daniele Tarchi, and Carla Raffaelli. A fog computing orchestrator architecture with service model awareness. *IEEE Transactions on Network and Service Management*, 2021.
4. Mathias Santos De Brito, Saiful Hoque, Thomas Magedanz, Ronald Steinke, Alexander Willner, Daniel Nehls, Oliver Keils, and Florian Schreiner. A service orchestration architecture for fog-enabled infrastructures. In *2017 Second International Conference on Fog and Mobile Edge Computing (FMEC)*, pages 127–132. IEEE, 2017.

5. Daniela Delinschi, Rudolf Erdei, and Oliviu Matei. Ontology driven high performance message system for distributed software platforms. *AQTR*, 2022.
6. Christian Esposito, Aniello Castiglione, Francesco Palmieri, Massimo Ficco, Ciprian Dobre, George V Iordache, and Florin Pop. Event-based sensor data exchange and fusion in the internet of things environments. *Journal of Parallel and Distributed Computing*, 118:328–343, 2018.
7. Aradhana Goutam, Rajkamal, and Maya Ingle. Orchestrator model for system security. In Srijia Unnikrishnan, Sunil Surve, and Deepak Bhoir, editors, *Advances in Computing, Communication and Control*, pages 195–199, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
8. Bernd Jaeger. Security orchestrator: Introducing a security orchestrator in the context of the etsi nfv reference architecture. In *2015 IEEE Trustcom/BigDataSE/ISPA*, volume 1, pages 1255–1260. IEEE, 2015.
9. Pouya Jamborsalamati, Edstan Fernandez, Mojtaba Moghimi, M. Jahangir Hossain, Alireza Heidari, and Junwei Lu. Mqtt-based resource allocation of smart buildings for grid demand reduction considering unreliable communication links. *IEEE Systems Journal*, 13(3):3304–3315, 2019.
10. Pooyan Jamshidi, Claus Pahl, Nabor C Mendonça, James Lewis, and Stefan Tilkov. Microservices: The journey so far and challenges ahead. *IEEE Software*, 35(3):24–35, 2018.
11. Oliviu Matei, Rudolf Erdei, Daniela Delinschi, and Laura Andreica. Data based message validation as a security cornerstone in loose coupling software architecture. In *Computational Intelligence in Security for Information Systems Conference*, pages 214–223. Springer, 2021.
12. Oliviu Matei, Rudolf Erdei, Alexandru Moga, and Robert Heb. A serverless architecture for a wearable face recognition application. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10-15, 2021, Proceedings, Part VII*, pages 642–655. Springer, 2021.
13. Oliviu Matei, Katarzyna Materka, Paweł Skyscraper, and Rudolf Erdei. Functionizer-a cloud agnostic platform for serverless computing. In *International Conference on Advanced Information Networking and Applications*, pages 541–550. Springer, 2021.
14. Oliviu Matei, Paweł Skrzypek, Robert Heb, and Alexandru Moga. Transition from serverfull to serverless architecture in cloud-based software applications. In *Proceedings of the Computational Methods in Systems and Software*, pages 304–314. Springer, 2020.
15. Ricardo Silva Peres, Lilian Adkinson, Emilia Cioroai, Eda Marchetti, Enrico Schiavone, Sara Matheu, Ovidiu Cosma, Radosław Piliszek, and José Barata. The bieco conceptual framework towards security and trust in ict ecosystems. 2021.
16. Darius Saif and Ashraf Matrawy. A pure http/3 alternative to mqtt-over-quic in resource-constrained iot. In *2021 IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 36–39, 2021.
17. Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3(5):637–646, 2016.
18. Zhenyu Wen, Renyu Yang, Peter Garraghan, Tao Lin, Tao Lin, Tao Lin, Jie Xu, Jie Xu, and Michael Rovatsos. Fog orchestration for internet of things services. *IEEE Internet Computing*, 2017.
19. Shanhe Yi, Zijiang Hao, Zhengrui Qin, and Qun Li. Fog computing: Platform and applications. In *2015 Third IEEE workshop on hot topics in web systems and technologies (HotWeb)*, pages 73–78. IEEE, 2015.
20. Adel Zaalouk, Rahamatullah Khondoker, Ronald Marx, and Kpatcha Bayarou. Orchsec: An orchestrator-based architecture for enhancing network-security using network monitoring and sdn control functions. In *2014 IEEE Network Operations and Management Symposium (NOMS)*, pages 1–9. IEEE, 2014.

# A comparative study of Machine Learning algorithms for the detection of vulnerable Python libraries

Laura Pérez-Vilarelle<sup>1</sup>, Eva Sotos Martínez<sup>1</sup>, and Javier Yépez Martínez<sup>1</sup>

**Abstract** Detecting the existence of vulnerabilities within source code is an important step in improving the overall security of an organisation and reducing the possibility of an attacker breaching the IT system. This has led to the creation of different vulnerability detection tools and, therefore, to devoting efforts to the study of detection techniques to provide the best results. One of the techniques used for this purpose is those that use Machine Learning and Data Mining models, this being a booming field. Under this premise, this paper presents a comparison of the results obtained with Machine Learning models capable of classifying the vulnerability or non-vulnerability of a real-world source code in Python language.

## 1 Introduction

Software vulnerabilities make companies, governments, society or individuals an easy target for cyber-attacks, causing security breaches that can become vital. Common measures such as model-based secure software design, up-to-date collections of well-known vulnerabilities, and security guidelines are critical steps in developing secure software. These guidelines can be complemented with analytical quality assurance techniques or tools to detect vulnerabilities in the source code.

Many have been the procedures and techniques used in the development of vulnerability detection tools. In this case, the study is focused on those tools that perform a static code analysis by the use of a combination of techniques that address Machine Learning (ML) and Data Mining (DM). In a previous study on the state of the art about techniques used in the detection area [1], three groups of approaches have been differentiated: anomaly detection, code pattern recognition and Vulnerability Prediction Models (VPM). This paper focuses on the use of VPM.

---

Gradiant, Carretera Vilar, nº 56-58, 36214 Vigo, Spain,  
e-mail: {lperez,esotos,jyopez}@gradient.org,  
<https://www.gradient.org/en/>

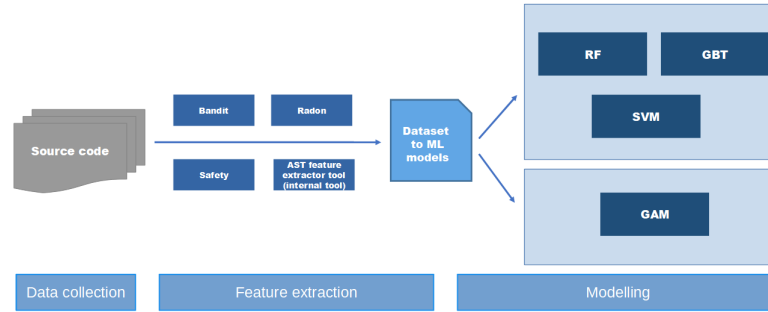
Several are the studies based on software properties to predict vulnerable code. Neuhaus et al [2] is one of the first on this field. This approach is based on the likelihood of vulnerable files to share similar sets of imports and functions calls. Authors proposed to use this imports and calls as independent variables along with Support Vector Machine (SVM) models. A similar idea is proposed by Schoter et al. [3], which focuses on the relationships between components by using models such as Linear Regression (LR), Ridge Regression (RR), SVM or Decision Tree (DT). There is extensive literature that use source code metrics to build their models. An example is Shin et al. [4] by the use of LR. Years later, in addition to complexity metrics, code churn and develop activity metrics are evaluated through Naive Bayes (NB), Random Forest (RF), and LR [5]. Other studies use classification and regression trees (CART) to predict attack-prone components, as Gegick et al. [6]. Additionally, Morrison et al. [7] studied typical code metrics and dependency metrics and evaluated their correlation with post-release vulnerabilities using models like SVM, NB, RF or LR. Newly, Bilgin et al. [8] uses text mining techniques and the Abstract Syntax Tree (AST) of the source code to analyse and investigate if ML algorithms like multilayer perceptron (MLP) or convolutional neural network (CNN) can distinguish vulnerable code fragments.

The aforementioned works use techniques based on VPM to detect vulnerabilities in the source code. Nevertheless, the difference between the used approaches and resources as well as the evaluation methodology, makes difficult to establish a comparison between all of them. However, the inputs made in the state of the art have contributed to the selection of variables that define the vulnerable character of a code resource, letting us to implement various VPM that leads to a comparative study between different ML algorithms to detect libraries vulnerabilities in Python code. This allows analyzing which one offers the best fit for different stages.

To capture this study, this paper is structured as follows: Section 2 introduces the proposed approach to the problem describing the stages of data collection, feature extraction and modelling, with a brief review about predictive models. Section 3 describes the implementation and compares the results obtained by the ML models. Finally, Section 4 provides conclusions and the main challenges for future works.

## 2 Approach

For the classification of Python code resources into vulnerable or non-vulnerable using ML models, an approach consisting of three stages has been carried out (Figure 1): In a first stage, the information available for analysis is extracted from the corresponding source (Subsection 2.1). A second phase generates knowledge through different static code tools (Subsection 2.2). Finally, the modelling of the data is performed by analysing different models oriented to different approaches: those that are suitable for managing all the knowledge obtained, and another that allows explaining the contribution of information from the data used in the classification, at expenses of a greater treatment and preprocessing of the data (Subsection 2.3).



**Fig. 1** Approach diagram: phases of the process, the static analysis tools used and ML models implemented.

## 2.1 Data collection

Obtaining the public dataset used in the training of the ML algorithms has been done through an internally developed library download tool. This tool takes public libraries from Pypi database<sup>1</sup> and divides them into vulnerable and non-vulnerable. To have this classification, a repository<sup>2</sup> has been consulted, which contains the names and versions of libraries vulnerable to a given CVE (Common Vulnerabilities and Exposures) or PVE (in case that the CVE is still provisional). Thus, a classification has been made by tagging vulnerable libraries as those that have at least one known vulnerability, and secure libraries as the ones that have no known vulnerabilities.

## 2.2 Feature extraction

Once the data is obtained, it is required to define the variables that will provide the information to predict the existence or not of vulnerabilities. To acquire them, different existing tools have been used, together with an internally developed tool which obtains extra information not provided by the existing ones. These tools perform a static analysis of the source code and extract different features to be used in the ML models. These tools are described below:

- **Bandit<sup>3</sup> (static code analysis):** Provides common security issues in Python code. It generates a report with different issues together with the number of evaluated lines in the code and the number of issues classified by confidentiality and severity.

<sup>1</sup> <https://pypi.org/>

<sup>2</sup> <https://pyup.io/safety/>

<sup>3</sup> <https://bandit.readthedocs.io/en/latest/index.html>

- **Radon<sup>4</sup>** : Computes various code metrics to obtain information about code complexity. These supported metrics are: Raw metrics, Cyclomatic Complexity, Halstead metrics and Maintainability Index.
- **Safety<sup>5</sup> (dependency check)**: Checks the local virtual environment, required files or any input from *stdin* for dependencies with security issues. The tool provides a dependency report indicating those vulnerable libraries on which they are dependent on.
- **AST<sup>6</sup> feature extractor tool**: Internally developed tool based on the AST (Abstract Syntax Tree) code representation which describes its static characteristics.

In order to use the different tools selected to obtain the features in a standard way, it is necessary to adapt the downloaded files since they present heterogeneity. For this, the source code files are extracted through the configuration file of each library in such a way that test or documentation files are discarded.

## 2.3 Modelling

Once the different features are obtained, the data is preprocessed and an Exploratory Data Analysis (EDA) is performed. When selecting the most suitable ML model and after having previewed previous works [9], Tree-based models have been chosen as a starting point. Taking into account the complex nature of the managed data and the difficulty to find a single global model to reflect the relationship between features, a comparison of the results from Tree-based methods, and other methodologies such as Support Vector Machine (SVM) and Generalised Additive Models (GAM), has been carried out. This selection covers two different approaches. On the one hand, Tree-based methods and SVM, which allow considering an extensive number of features and require low preprocessing of the data obtained. On the other hand, the GAM provides a different approach that allows to explain the information contribution of the features in the prediction of the model. In the following subsections, a brief review of the used models is presented.

### 2.3.1 Tree-based methods

Tree-based methods have become one of the benchmarks within the predictive fields since they provide good results in different areas, either regression or classification problems [10]. These are based on the creation of several decision trees whose nodes are used to make the prediction. The trees are achieved by taking an initial node, formed by the entire training sample, and subdivided it conditioned by a certain characteristic into two new subsets, giving rise to two new nodes. Subsequently, the

<sup>4</sup> <https://radon.readthedocs.io/en/latest/>

<sup>5</sup> <https://pyup.io/safety/>

<sup>6</sup> <http://icps.u-strasbg.fr/pop/gcc-ast.html>



process is recursively performed a predetermined finite number of times, obtaining a final decision tree. The ensemble of individual decision trees that combine multiple models into a new one allows to achieve a balance between bias and variance, a common issue present in statistical learning and ML models.

### Random Forest

One of the most used ensemble models to solve a possible lack of compensation between bias and variance is *bagging*. One example of an algorithm is Random Forest (RF) [10]. These adjust multiple models, each one trained with different training data, and provide the mean of the different obtained predictions or the most frequent class.

The algorithm can be described as followed:

1. Being  $B$  the number of trees, for  $b = 1$  to  $B$ 
  - a. Draw a bootstrap sample  $Z^*$  of size  $N$  from the training data.
  - b. Grow a random-forest tree  $T_b$  to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size  $n_{min}$  is reached.
    - i. Select  $m$  variables at random from the  $p$  variables.
    - ii. Pick the best variable/split-point among the  $m$ .
    - iii. Split the node into two daughter nodes.

2. Output the ensemble of trees  $\{T_b\}_1^B$ .

To make a prediction for classification problem at a new point  $x$ :

Let  $C_b(x)$  be the class prediction of the  $b$ th random-forest tree. Then

$$C'_{rf}(x) = \text{average all predictions } \{C_b(x)\}_1^B$$

### Gradient Boosting Tree

Another way to acquire the balance between bias and variance is the ensemble models based on *boosting*. Unlike bagging, these adjust sequentially multiple simple models, taking into account the information of the previous tree to correct prediction errors to improve each iteration. A model that uses this kind of metric is the well-known Gradient Boosting Tree (GBT).

The algorithm for its implementation, can be described as follows:

1. Initialize  $f_0(x) = \arg \min_{\gamma} \sum_{i=1}^N L(y_i, \gamma)$ , where  $L$  is the loss function.
2. For  $m = 1$  to  $M$ :
  - a. For  $I = 1, 2, \dots, N$  compute  $r_{im} = -[\frac{\delta L(y_i, f(x_i))}{\delta f(x_i)}]f = f_{m-1}$
  - b. Fit a regression tree to the targets  $r_{im}$  giving terminal regions  $R_{jm}$ ,  $j = 1, 2, \dots, J_m$ .
  - c. For  $j = 1, 2, \dots, J_m$  compute  $\gamma_{jm} = \arg \min_{\gamma} \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma)$ .
  - d. Update  $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm})$ .
3. Output  $f'(x) = f_M(x)$ .

### 2.3.2 Support Vector Machine

A different method to consider is the SVM [11]. This is a supervised and linear ML algorithm created to solve binary classification problems, dividing the data through hyperplanes, and maintaining all the main features that characterise the algorithm. The objective is to build a tolerance margin ( $\epsilon$ ) with which there will be observations (support vectors) inside and outside the margins. Using only the observations outside the margins, the errors  $\xi$  are calculated and with them the objective function ( $FO$ ) to be minimised is constructed, i.e.:

$$\text{Minimize: } FO = \frac{1}{2}\omega^2 + C \sum_{i=1}^N \xi_i, \text{ subject to } y_i(\omega, x_i) \geq 1, \forall i$$

Where  $\omega$  is the vector with the slopes associated with each of the variables and  $C$  is the penalty value for the errors.

### 2.3.3 Generalised Additive Models

An alternative methodology in terms of predictive models is the use of General Linear Models (GLM) [12], and particularly the GAM models, an extension that allows the incorporation of nonlinear relationships. In GAM models, the relationship of each predictor with the mean of the response variable can be made by means of a linear or nonlinear function. In practice, the most commonly used functions are nonlinear smooth-type functions (cubic regression splines, thin plate regression splines or Penalised splines).

The purpose of GAM is to maximise the prediction accuracy of a dependent variable and several distributions by specific non-parametric functions of the predictor variable which are connected to the dependent one through a link function. In other words, GAMs structures can be expressed as:

$$g(E(Y)) = \beta + f_1(x_1) + f_2(x_2) + \dots + f_m(x_m)$$

Where  $g(E(Y))$  is the link function which associates the expected value with the predictive variables  $x_1, x_2, \dots, x_m$ , and  $f_1(x_1) + f_2(x_2) + \dots + f_m(x_m)$  is the functional form with an additive series that generates the response variable  $Y$ .

The proposed model selection aims to show a comparative between them. In the case of predicting whether a certain library is vulnerable or not, tree-based methods gives indications of being adequate since they offer suitable analysis results when a large number of features are handled. These models offer an automatic selection of predictors, good scalability, low dependence on outliers, and do not need standardisation, among other properties. Likewise, SVM provides good results when it comes to data sets with binary classification outputs, which suggest a good performance when detecting if a library is vulnerable or not. Finally, to evaluate the possible improvement of results and to carry out a more complete study, the choice of GAM is made, presenting a different approach to the previous one. These models allow greater flexibility in the dependence of each covariate with the response variable.

### 3 Implementation and Evaluation

In the data collection's stage, a total of 3,204 versions of different libraries are downloaded, of which 1,930 are vulnerable and 1,274 non-vulnerable. When it comes to the features to be used in the different ML models, the feature extraction process collects more than 100 different characteristics. This will consolidate the dataset that defines the problem.

After an EDA analysis, it has been possible to verify a large presence of atypical data, the existence of similar records, dependence between numerous features and low dependence of them with the response variable, or unbalancing dataset, among others. These cause a significant reduction in the dataset and the need to perform a balance to not compromise the results.

In the case of the models that follow the first approach (decision trees and the SVM) 84 features have been selected. For the second approach, and since GAM algorithms require exhaustive recompilation of the most significant features, a selection has been carried out, discarding those variables with a high correlation between them. Furthermore, an own selection has been made, taking those that are considered most significant for a vulnerability prediction. Thus, a total of 48 features have been selected for the implementation of the GAM model.

In the current analysis, two different cases have been modelled: one in which a set of paired data is provided, and another whose samples are unpaired. For the former, a vulnerable and non-vulnerable version of each library has been chosen. For the latter, the vulnerable and non-vulnerable samples correspond to different libraries. This distinction is made since GAM models require the use of unpaired datasets.

For the implementation of the different models, the use of the Python programming language has been chosen. The public libraries `RandomForestClassifier`, `GradientBoostingClassifier`, `SVC`, `linear_model.LogisticRegression` and `LogisticGAM` from *sklearn*<sup>7</sup> are used for RF, GBT, SVM and GAM models respectively.

For all cases, samples have been divided into those used for the training process and those for testing. In addition, a grid search has been implemented by cross validation for obtaining the hyperparameters corresponding to each model which provide the best result.

#### 3.1 Evaluation

Next, a comparative study of the different models trained and evaluated in several different scenarios is progressively presented. The goal is to achieve a model capable of identifying vulnerabilities in a real code resource as reliably as possible.

In a first test, vulnerable and non-vulnerable libraries to a known CVE configure the dataset, discarding libraries and versions marked with PVE. The results obtained in the different models with said dataset are not conclusive since the number of

---

<sup>7</sup> [https://scikitlearn.org/stable/user\\_guide.html](https://scikitlearn.org/stable/user_guide.html)

existing samples in it is low for a correct implementation of the selected models. To solve the lack of samples in the dataset, libraries with PVE have been added. After this extension of the dataset, the models have been evaluated in both paired and unpaired data scenarios, with the exception of GAM models whose nature only allows their use with unpaired data.

The results, provided in Table 1, showed low adjustments in all the models and cases therefore a data extension has not provided solutions in improving the modelling. In a comparison between the different models, there is no indication of a significant difference. In addition, a low accuracy with the training data indicates that models are not being able to learn.

Taking into account the possible different scenarios, a second test is performed that aims to improve the model's results. To do this, contextualization is added to the selected dataset by means of re-coding of the output feature. Table 2 shows the results achieved. Since the results with paired and unpaired data do not show any difference, only results of the models evaluated in the set containing paired data are shown, except for the case of GAM models that use unpaired data. In general terms, all the models show a slight improvement after the transformation of the response variable and the trend in the variation between the training and testing adjustments is maintained. The differences between the models are still not significant and the unpaired scenario shows a weaker improvement over this transformation.

Following the results obtained, it is proposed to reformulate the problem and explore the possibility of modifying the response feature to the different types of CVE contained in the libraries. It is revealed that a certain library can have several different vulnerabilities, and a new scenario is obtained, in which the records corresponding to a certain library appear as many times as different CVEs they contain. The new dataset includes all the original libraries considered no-vulnerable, and those vulnerable to a given CVE, discarding those with PVE label. This new dataset contains a total of 1613 records. Due to the fact that data samples are paired, GAM is not performed.

The performance of the algorithms with the new dataset provides results with significant improvements. In this case, the evaluated models present an adjustment of around 75%, being RF the one that provides better results (Table 3). The evaluation of the performance of said model can be seen in Table 4, where precision, recall and

**Table 1** Evaluation of selected models for paired and unpaired data scenarios. It has been taken into account the accuracy training, accuracy testing and the AUC (Area Under the ROC Curve).

Dataset	Model	Accuracy training	Accuracy testing	AUC
Paired data	RF	0.509	0.502	0.518
Paired data	GBT	0.602	0.579	0.539
Paired data	SVM	0.6	0.581	0.541
Unpaired data	RF	0.497	0.496	0.528
Unpaired data	GBT	0.532	0.558	0.569
Unpaired data	SVM	0.552	0.554	0.552
Unpaired data	GAM	0.521	0.523	0.498

**Table 2** Model's evaluation when a re-coding of the output feature is performed.

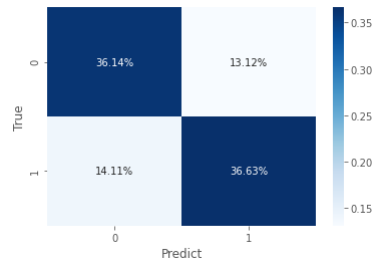
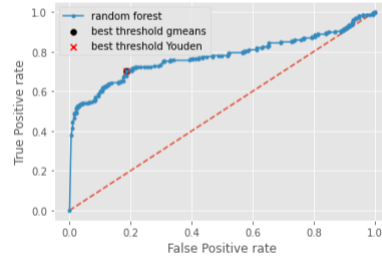
Model	Accuracy training	Accuracy testing	AUC
RF	0.566	0.561	0.623
GBT	0.651	0.634	0.628
SVM	0.653	0.631	0.535
GAM	0.642	0.564	0.573

**Table 3** Model's evaluation considering the number of CVE per library.

Model	Accuracy training	Accuracy testing	AUC
RF	0.754	0.727	0.776
GBT	0.751	0.702	0.719
SVM	0.653	0.621	0.637

**Table 4** Descriptive Parameters of the Results for the RF model.

Outputs	Precision	Recall	F1-Score	%Support
0	0.72	0.754	0.727	0.493
1	0.74	0.653	0.621	0.507

**Fig. 2** Confusion matrix for RF model**Fig. 3** ROC curve and best threshold for RF model.

F1 score are detailed, offering values around 70% in both precision and recall. In addition, the implementation of the RF offers a confusion matrix with false positive and false negative rates greater than 10% (Figure 2). By focusing on the ROC curve and the AUC, performance of the model presents values close to 80% (Figure 3).

## 4 Conclusions

In this paper, a comparison of different ML models for the detection of vulnerable libraries has been presented. During this analysis, some problems have arisen due to the high correlation between the features obtained with the static code tools, the low linear dependence between the input and output features from the model, or the difficulties in obtaining correctly labelled public libraries of real code.

Despite the faced issues, this study has verified that the results obtained do not depend to a great extent on the chosen model, since their results are similar. Further-

more, the addition of records to the sample does not provide relevant improvements. Only a better contextualization of the problem and re-coding the response function has shown slight improvements in the results. The treatment of code resources, the selection of tools and the adequacy of the techniques and procedures have allowed us to obtain an RF model that is around 75% accurate in the classification.

Taking these results into account, a future work is considered which will focus on improving the data set obtained and the selection of features used, contextualising and better understanding the problem, and using the proposed approach to classify different types of vulnerabilities. Likewise, the possibility of redirecting research towards the improvement of knowledge acquisition is raised, focusing on the information provided by the AST that has been slightly treated in this work.

**Acknowledgements** This work was supported by the project BIECO ([www.bieco.org](http://www.bieco.org)) that received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 952702, and by the Ayudas Cervera para Centros Tecnológicos grant of the Spanish Centre for the Development of Industrial Technology (CDTI) under the project ÉGIDA (CER-20191012).

## References

1. E. Sotos Martínez, N. M. Villanueva, and L. Adkinson Orellana. A survey on the state of the art of vulnerability assessment techniques. In *Computational Intelligence in Security for Information Systems Conference*, pages 203–213. Springer, 2021.
2. S. Neuhaus, T. Zimmermann, C. Holler, and A. Zeller. Predicting vulnerable software components. In *Proceedings of the 14th ACM conference on Computer and communications security*, pages 529–540, 2007.
3. A. Schröter, T. Zimmermann, and A. Zeller. Predicting component failures at design time. In *Proceedings of the ACM/IEEE Int. symposium on Empirical Soft. Eng.*, pages 18–27, 2006.
4. Y. Shin and L. Williams. An empirical model to predict security vulnerabilities using code complexity metrics. In *Proceedings of the 2nd ACM-IEEE international symposium on Empirical software engineering and measurement*, pages 315–317, 2008.
5. Y. Shin, A. Meneely, L. Williams, and J.A. Osborne. Evaluating complexity, code churn, and developer activity metrics as indicators of software vulnerabilities. *IEEE transactions on software engineering*, 37:772–787, 2010.
6. M. Gegick, L. Williams, J. Osborne, and M. Vouk. Prioritizing software security fortification through code-level metrics. In *Proceedings of the 4th ACM workshop on Quality of protection*, pages 31–38, 2008.
7. P. Morrison, K. Herzig, B. Murphy, and L. Williams. Challenges with applying vulnerability prediction models. In *Proceedings of the Symposium and Bootcamp on the Science of Security*, pages 1–9, 2015.
8. Z. Bilgin, M.A. Ersoy, E.U. Soykan, E. Tomur, P. Çomak, and Karaçay L. Vulnerability prediction from source code using machine learning. *IEEE*, 8:150672–150684, 2020.
9. M. Jimenez. *Evaluating vulnerability prediction models*. PhD thesis, Univ. Luxembourg, 2018.
10. G. James, D. Witten, T. Hastie, and R. Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.
11. S. Suthaharan. Support vector machine. In *Machine learning models and algorithms for big data classification*, pages 207–235. Springer, 2016.
12. A.J. Dobson and A.G. Barnett. *An introduction to generalized linear models*. Chapman and Hall/CRC, 2018.

# Evaluation of the reliability index of IP addresses in reputation lists

Alberto Miranda-García, Ignacio Samuel Crespo-Martínez, Ángel Manuel Guerrero-Higueras, and Vicente Matellán-Olivera

**Abstract** IP address reputation lists are a collection of IP addresses that have been associated with malicious practices. Therefore, it is essential to evaluate the addresses' reliability in this list to avoid false positives, regardless of the metrics that have been classified as malicious. Reliability is defined as the probability that such IP addresses are a significant threat due to their appearance on different reputation lists. Generating an evaluation index to evaluate the reliability of the elements in a global way allows us to evaluate IP addresses. Thus, a defined metric will be used based on appearances in external reputation lists, the dates they were added and last seen, and other variables such as severity and repetition in the list itself. The proposed system presents an IP address evaluation module composed of a series of asynchronous processes to evaluate the reliability of the elements.

**Key words:** Blacklists, Incident Assessment, Network traffic, IP addresses, Reputation Assessment

## 1 Introduction

Reputation lists are collections of IP addresses belonging to malicious networks (URLs, IP addresses, domain names, etc.) that are generally used as the primary

---

Alberto Miranda-García  
University of León, León, Spain. e-mail: amirg@unileon.es

Ignacio Samuel Crespo-Martínez  
University of León, León, Spain. e-mail: icrem@unileon.es

Ángel Manuel Guerrero-Higueras  
University of León, León, Spain. e-mail: am.guerrero@unileon.es

Vicente Matellán-Olivera  
University of León, León, Spain. e-mail: vicente.matellan@unileon.es

mechanism for access control. There are a large number of publicly available reputation lists. There are lists focused on specific attacks, such as IP addresses classified as Spam, Phishing, Command and Control (C&C), etc. Lists can be used to implement a Spam filter in an email server, to filter data flows in a NIDS (Network Intrusion Detection System) belonging to Command and Control [7], etc. There are two main use cases for applying reputation lists in a NIDS. First, it is imperative to analyze all the data flows received, checking the IP addresses to avoid incoming attacks. On the other hand, another less common use case is to monitor the outgoing traffic [10]. It allows for knowing the impact of the activities carried out by our systems on the blacklists and being able to mitigate a possible infection in our network.

Reputation lists are usually maintained by a company or an entity and can be public, as is the example of the FireHOL or CINSARmy lists, or private. Those in charge of keeping a list updated use metrics [3] or external sources to add a new IP address. However, such external sources are usually private or are not specified. Therefore, the reliability of the elements in these lists is not corroborated.

There are currently more devices connectable to the network than assignable IPV4 addresses. That is why public IP address reassignment techniques are being used, and everything is being migrated to IPV6 [14]. The lack of IP addresses is why an IP address that has been used to send Spam can be reassigned to an ordinary household and still belong to a reputation list. Another aspect to consider is the possibility that a malicious device makes the connection through a proxy or VPN (Virtual Private Network); in this case, one of these would appear on the list.

However, blocking potentially malicious network traffic using reputation lists is not the main objective of this work. The aim is to validate that the addresses contained in the lists maintain certain reliability. Using multiple reputation lists and multiple data sources can be beneficial due to a broader reach. There are articles that studied the effectiveness of reputation lists obtaining positive results [12], stating that public IP address reputation lists do not contain more than a quarter of all IP addresses belonging to malicious activities on the network. It does not mean that it is no longer a critical need to be able to validate the addresses that are listed.

This research aims to validate the reliability of the IP addresses contained in reputations lists. To achieve our goal, we propose a two-step system. On the one hand, the data from defined sources are retrieved and stored. On the other hand, an analysis of the IP addresses is carried out, and reports with detailed information are generated. To validate this system, we propose to analyze the correlation between the index developed in our analysis and the reliability index previously defined in the lists. The generated index is calculated based on addresses, dates and other factors. The proposed approach maintains a highly reliable reputation list, thus avoiding false positives.



## 2 Materials and methods

### 2.1 Reputation Lists

As a data source for this study, a set of public blacklists have been used, containing a list of IP addresses marked as malicious traffic and categorized by their type of attack. In addition, data obtained by Suricata (IDS) and the INCIBE reputation list have also been used, but these data are obtained by collaborating entities whose privacy policy restricts the exchange of information and therefore the source cannot be shared. The external reputation lists are specialized in different types of attacks to cover the most significant categories. The lists and data relating to each of them are described below.

- **SSL Blacklist:** This list has used the associated SSL certificate to add addresses and contains entries older than 30 days. It contains an average of 61 entries. Offered by Abuse.ch [1].
- **Feodo Tracker:** List of addresses associated with Dridex, Heodo (Emotet), Trick-Bot, QakBot (QuakBot) and BazarLoader (BazarBackdoor) malware. The update rate for this list is 5 minutes. Each update has a total of 745 IP addresses on average. Offered by Abuse.ch.
- **BadGuys:** List with IP addresses of all countries, evaluating their origin and category. It maintains a policy of unique addresses with which it tries to filter addresses from other lists and only keeps elements that do not appear in other external lists. This list has an average of 15,000 elements on average for each update. Offered by Cinsarmy [2].
- **Level3:** Set of addresses categorized as attacks, *spywares* and viruses. It contains items detected in the last 30 days and is updated approximately every 3 hours. It contains on average a total of 17,100 entries. Offered by FireHOL [8].
- **iBlocklist:** List of addresses associated with DDOS attacks. It has an approximate update rate of 5 days. It contains an average of 11,000 IP addresses. Offered by FireHOL.
- **Normshield:** List of IPs whose category is *WannaCry* [13]. This list is updated approximately every 2 days. It contains an average of 1050 IP addresses. Offered by FireHOL.

To this set of external lists, the following data sources used in this study are added:

- **INCIBE List:** This list is provided by the National Cybersecurity Institute of Spain and contains IP addresses categorized as malicious. The elements of this list are updated daily with information from third-party sources and companies associated with this entity. This list is not publicly available.

Data have been normalized to evaluate lists. All the addresses obtained from the different sources, except those obtained by the INCIBE list, are formatted with the following attributes: *IP*, *firstseen*, *lastseen* and *active*. On the other hand, the IP addresses obtained from the INCIBE lists and the attributes that maintain the data from other sources include the fields described in Table 1.

**Table 1** Fields in the database.

Field	Description
<i>IP</i>	IP Address
<i>lastseen</i>	Date of last seen in the list.
<i>firstseen</i>	Date of first time seen/added
<i>day</i>	Number of days it has been listed
<i>categories</i>	Categories in which it has been classified
<i>severity</i>	Severity Index
<i>reliability</i>	Reliability Index
<i>onlist</i>	Number of repetitions in the list
<i>geo</i>	Field with latitude and longitude to geo-locate

## 2.2 IDS - Intrusion Detection System

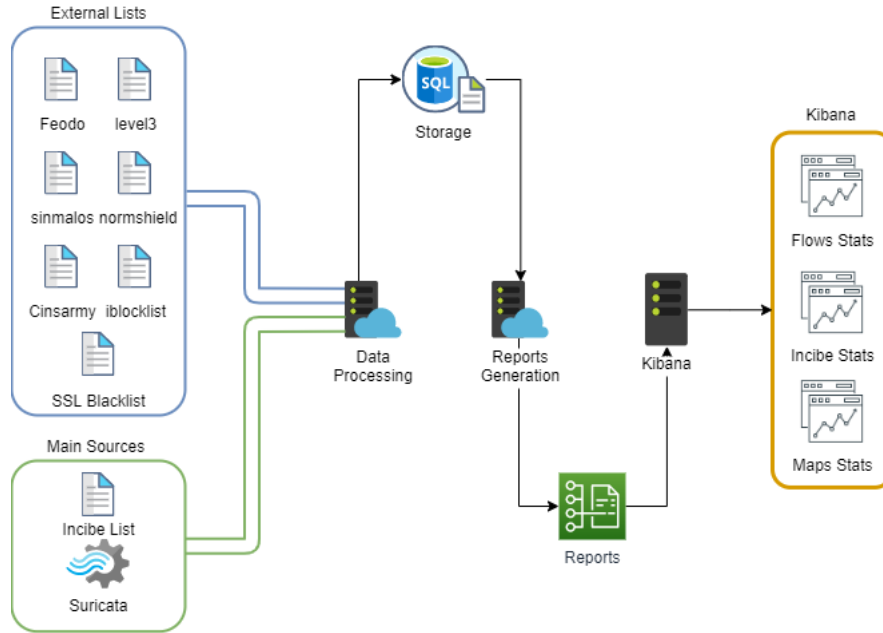
In our work, the data sources have been enriched by adding the records of the operational IDS in RedCAYLE [15], a project managed by the Castilla y León Supercomputing Center Foundation that keeps educational centres, university hospitals, scientific infrastructures and technology parks connected through high-speed links. Specifically, Suricata is used, which is an open-source threat detection engine which combines an intrusion detection system and an intrusion prevention system [16]. This threat detection engine generates logs in real-time through defined access rules, notifying, for example, when it detects a port scan or unauthorized access has been attempted.

In the same way, as with the data obtained by the public reputation lists, all the IP addresses detected in the records are formatted and stored in the database with the following fields: *IP*, *lastseen*, *firstseen*, *day* and *categories*. These records are generated in real-time, but the data dumping in the proposed architecture is carried out periodically daily.

## 2.3 Implementation

For this work, an architecture with different sections is proposed, covering all the previously described needs, the data dump and the analysis. Furthermore, the proposed architecture must keep processes separated to allow the generation of reports at any time in an asynchronous manner so as not to depend on updating the lists and vice versa.

The proposed data collection and analysis processes run on two independent machines, which share the data collection and processing and the analysis and report generation. Therefore, an architecture with different sections have been designed as shown in Figure 1. Another point to remember is that if we want to add a certain level of reliability to our list, adding different data sources as the IDS is needed. But as we have previously pointed out, this can imply a series of problems. For this



**Fig. 1** Overall architecture.

reason, it is essential to eliminate overlaps and treat and normalize all the data that will be used so that all the data collected can be analyzed.

In the first section, data is obtained from the different sources of information. As previously explained, the update rates of each of them are different, so an asynchronous download and processing system has been designed. Keeping a repository in which all the lists and sources are stored is essential. Thus, all data is stored in a NoSQL database. MongoDB has been used since as different [6] studies point out that when it comes to large numbers of entries, the performance of this type of database stands out compared to other databases such as MySQL.

In the section related to reports, a series of processes are executed in charge of data analysis, report generation, and sending the reports and visualization in Kibana [11]. The first process is executed periodically every week because the list update processes are executed asynchronously. The generated reports contain information on each address, from the general information included in the lists to the analysis data. Therefore, in addition to the data included in the database (see Table 1), the fields that contain each element in the reports are shown in Table 2.

The reputation list offered by INCIBE has been used as the sample to validate our proposal. Each element contains a reliability field whose obtaining metrics are unknown. We will evaluate the correlation it maintains with the Evaluation Index we generated in the analyses.

**Table 2** Additional fields in Reports.

Field	Description
<i>reportDate</i>	Report Generation Date
<i>eval</i>	Address evaluation index
<i>count</i>	Number of occurrences in the different data sources
<i>appearList</i>	List of sources in which it appears
<i>active</i>	Classification as active or past threat.

## 2.4 Metrics

The analysis process in this research is the main section of this study, where the *eval* and *active* fields are generated and added to the reports. These fields are calculated based on the different data obtained from the lists and data sources.

On the one hand, the Evaluation Index field (*eval*) takes as reference the fields *count*, *severity* and *onlist*. To calculate this index, the number of external lists in which the IP address appears is considered. Likewise, it is also considered if it has appeared in Suricata. Another influencing factor is the *onlist* field, which, as previously explained, counts the number of appearances on the INCIBE list itself since the same element can appear repeatedly classified with a different attack category. Finally, an average is made and taken as a reference to use the severity field of IP addresses. The equation used to calculate the variable *eval* is an additive metric based on a series of values defined as shown below 1.

$$\begin{aligned} eval = & listsWeight + average(severity[]) * 0.125 \\ & + average(reliability[]) * 0.125 + onlist * 0.05 \end{aligned} \quad (1)$$

The evaluation index is a value between 0 and 10, where 10 is a high level of reliability and 0 is no reliability. The *listsWeight* is the sum of appearances in reputation lists. The appearance of IP addresses in each list has different weights based on the reliability of their sources. Suricata computes by 1.7 because the data provided by this source is given by redCAYLE that is a trusted entity. The weight of the other lists has been adjusted as the tests went on, depending on the percentage of appearance of the IP addresses. Cinsarmy by 1.4, Level3 by 1.2, FeodoTracker by 0.8, iBlockList by 0.7, Normshield by 0.6 and SSLBL by 0.6, being the maximum value for *listsWeight* 7 out of 10. Regarding the severity and reliability indexes, computes by 0.125 each of them, being 2.5 its maximum value between both. Finally, the *onlist* field maintains a great variability between each of the addresses, so it computes by 0.05 and establishes a maximum limit of 0.5 out of 10.

On the other hand, the *active* field is used to evaluate if the IP address remains an active threat. To evaluate this aspect, the difference between the fields *firstseen*, *lastseen* and *reportDate* and the value of the field *day* is taken as a reference. To obtain a value that is later evaluated, the following equation 2 is used. In this way,

we get a value that the closer it gets to 0, the greater the probability of continuing to be an active threat. The field *active* is Boolean, so a threshold is defined for the value calculated with the equation to define the field. After several tests with sample data using formula 2, it has been observed that the threshold that best fits to define whether an IP is considered active or not is the value 3. Those addresses whose value is greater than this are classified as not active.

$$indx = (1 - \frac{day}{diff(last - first)}) * (diff(act - last)) \quad (2)$$

To evaluate the correlation between the variable *variability* of the INCIBE lists, with the evaluation index generated in the analysis phase, and thus be able to validate the results, five ranges have been defined with the reliability values. No Reliability, when the value is between 0 and 2, Low Reliability between 2 and 4, Medium Reliability between 4 and 6, High Reliability between 6 and 8, and Maximum Reliability when the value is between 8 and 10. In order to evaluate this correlation, the Pearson correlation coefficient [4] has been used, which defines the linear dependence between two quantitative variables. For this, the equation in which the Pearson population correlation coefficient is defined as shown in Equation 3, where *Cov* is the covariance and  $\sigma$  is the population standard deviation. The correlation coefficient is defined between -1 and 1. 1 indicates a high correlation between the values.

$$\rho_{X,Y} = \frac{Cov(X,Y)}{\sigma_X \sigma_Y} \quad (3)$$

### 3 Results

The reports have been generated with the data obtained between November 22, 2021 and December 12, 2021. In this period, three different reports have been generated. The first one has a total of 56,937 unique addresses, the second with 64,564 unique addresses and the last with a total of 55,275 unique addresses.

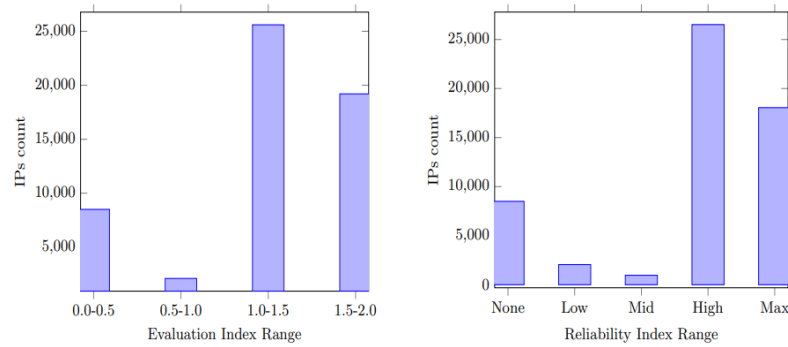
The report data, associated with the three periods, is shown in Table 3, where the mean of the evaluation index ( $\mu$ ) and the standard deviation ( $\sigma$ ) of the data for each reliability range are shown. In this period, the mean of the variations is 0.1166 ( $\sigma$ ). In the second period, the mean of the deviations is 0.1444 ( $\sigma$ ), and in the third one 0.1422 ( $\sigma$ ).

In the analysis of the data belonging to the first period, the correlation coefficient between reliability and the Evaluation Index is 0.9706, which shows a positive association. In the second period, the two variables have a correlation coefficient of 0.9748. And finally, in the third period, they maintain a correlation coefficient of 0.9833.

Figure 2 shows the number of unique IP addresses for each interval of the Evaluation Index, taking as data set those obtained in the first period. In the same figure, if

**Table 3** Reports Data

	Period 1		Period 2		Period 3	
Range Reliability	$(\mu)$	$(\sigma)$	$(\mu)$	$(\sigma)$	$(\mu)$	$(\sigma)$
No Reliability	0.30	0.101	0.47	0.185	0.39	0.141
Low Reliability	0.67	0.069	0.48	0.164	0.51	0.149
Reliability Medium	0.76	0.095	0.85	0.098	0.89	0.101
High Reliability	1.35	0.132	1.13	0.104	1.42	0.139
Reliability Max.	1.95	0.186	1.46	0.171	1.72	0.181

**Fig. 2** Reliability and Evaluation Indexes graphs.

we count the number of unique IP addresses for each reliability interval previously defined, the second graph shown in Figure 2 is generated.

After distributing the number of IP addresses in reliability and evaluation index ranges (Figure 2), the similarity in the distribution can be observed. This reflects that the correlation between the reliability index and our evaluation index generated in the analyses is favourable.

On the other hand, the following results have been obtained regarding the appearances of IP addresses from the INCIBE list in external sources and Suricata. Regarding the first interval, 98.34% of the IP addresses have not been listed in any sources. In the second interval, this percentage was 97.86% and in the third, 98.15%. These data reflect the low appearance rate maintained by the IP addresses on the INCIBE list compared to external lists. In a more detailed way, the appearance percentage has been calculated for each of the external lists, of the addresses that have been listed in other sources. The rates relative for each period are shown in Table 4.

**Table 4** Percentages of appearance by Period.

Reputation List	Period 1	Period 2	Period 3
Level3	36.40%	34.15%	36.92%
Feodo Tracker	27.73%	29.73%	28.84%
iBlockList	13.82%	13.24%	11.21%
CINArmy	12.29%	13.11%	11.98%
Suricata	9.39%	8.98%	10.45%
SSLBL	0.27%	0.59%	0.47%
Normshield	0.09%	0.19%	0.12%

## 4 Conclusions and further work

Malicious IP address blacklists are a great tool for different purposes like attack detection or mitigation. In our case, using external lists and other data sources such as Suricata allows us to generate the Evaluation Index. This index maintains a close correlation with the reliability index given by INCIBE, taking the Pearson correlation coefficient between both values as a reference. This value allows us to verify the reliability of INCIBE's IP addresses, which are added with metrics established by unknown third-party entities. The results obtained in this work have shown a positive association, indicating that the defined evaluation index can be used as an indicator to corroborate the reliability of IP addresses in reputation lists.

Research carried out in this same field [9], addresses the analysis of the reliability of IP addresses based on other factors such as domain information, AS numbers and domain blocks. This methodology focuses more on analyzing the elements individually, unlike this work which focuses mainly on the appearance in different external sources. The approach of these works is also relevant in this field of study since it adds additional information without directly depending on other sources. In a complementary way to the system proposed in this work, adding this type of analysis of each element would enrich the results and effectiveness.

This research opens the doors to companies and/or entities that maintain reputation lists, to validate the elements by taking external sources as references. In the same way that sources such as Suricata have been used in this study, it is possible to implement another type of IDS or IPS (Intrusion Prevention System) [5] in a straightforward way.

In future lines of this work, the objective is to design different metrics and improve existing ones to validate IP addresses more accurately. Another future aim is to expand the data sources emphasizing detection models and external lists aligned with the list's elements to be analyzed. It allows us to have a more accurate Evaluation Index.

**Acknowledgements** The research described in this article has been partially funded by the grant RTI2018-100683-B-I00 funded by MCIN/AEI/ 10.13039/501100011033 and by "ERDF A way of making Europe"; and by the Instituto Nacional de Ciberseguridad de España (INCIBE), under the

grant “ADENDA 8, Adenda de prórroga de la Adenda 4: detección de nuevas amenazas y patrones desconocidos (Red Regional de Ciencia y Tecnología)”, addendum to the framework agreement INCIBE–Universidad de León, 2019–2021.

## References

1. Abuse.ch: Fighting malware and botnets blacklists (2022). URL <https://abuse.ch/>
2. Army, C.: Collective intelligence network security (2022). URL <https://cinsarmy.com/list-download/>
3. Asad, H., Gashi, I.: Dynamical analysis of diversity in rule-based open source network intrusion detection systems. *Empirical Software Engineering* **27**(1), 4 (2021)
4. Benesty, J., Chen, J., Huang, Y., Cohen, I.: Pearson correlation coefficient. In: Noise reduction in speech processing, pp. 1–4. Springer (2009)
5. Chakraborty, N.: Intrusion detection system and intrusion prevention system: A comparative study. *International Journal of Computing and Business Research (IJCBR)* **4**(2) (2013)
6. Dipina Damodaran, B., Salim, S., Vargese, S.M.: Performance evaluation of mysql and mongodb databases. *Int. J. Cybern. Inform.(IJCI)* **5** (2016)
7. Dittrich, D., Dietrich, S.: P2p as botnet command and control: A deeper insight. In: 2008 3rd International Conference on Malicious and Unwanted Software (MALWARE), pp. 41–48 (2008)
8. FireHOL: A firewall for humans (2022). URL <https://firehol.org/>
9. Fukushima, Y., Hori, Y., Sakurai, K.: Proactive blacklisting for malicious web sites by reputation evaluation based on domain and ip address registration. In: 2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications, pp. 352–361 (2011)
10. Ghafir, I., Prenosil, V.: Blacklist-based malicious ip traffic detection. In: 2015 Global Conference on Communication Technologies (GCCT), pp. 229–233 (2015)
11. Gupta, Y.: Kibana essentials. Packt Publishing Ltd (2015)
12. Kühner, M., Rossow, C., Holz, T.: Paint it black: Evaluating the effectiveness of malware blacklists. In: A. Stavrou, H. Bos, G. Portokalidis (eds.) *Research in Attacks, Intrusions and Defenses*, pp. 1–21. Springer International Publishing (2014)
13. Mohurle, S., Patil, M.: A brief study of wannacry threat: Ransomware attack 2017. *International Journal of Advanced Research in Computer Science* **8**(5), 1938–1940 (2017)
14. Nikkhah, M., Guérin, R.: Migrating the internet to ipv6: An exploration of the when and why. *IEEE/ACM Transactions on Networking* **24**(4), 2291–2304 (2016)
15. RedSCAYLE: Red de ciencia y tecnología de castilla y león (2021). URL <https://www.scayle.es/redcayle/infraestructura/>
16. White, J.S., Fitzsimmons, T., Matthews, J.N.: Quantitative analysis of intrusion detection systems: Snort and Suricata. In: *Cyber Sensing 2013*, vol. 8757, pp. 10 – 21. International Society for Optics and Photonics, SPIE (2013)



# Forecasting the Number of Bugs and Vulnerabilities in Software Components using Neural Network Models

Ovidiu Cosma<sup>1</sup>, Petrică Pop<sup>1</sup>, Cosmin Sabo<sup>1</sup> and Laura Cosma<sup>2</sup>

**Abstract** The frequency of cyber attacks has been rising rapidly lately, which is a major concern. Because each attack exploits one or more vulnerabilities in the software components that make up the targeted system, the number of vulnerabilities is an indication of the level of security and trust that these components provide. In addition to vulnerabilities, the security of a component can also be affected by software bugs, as they can turn into weaknesses, which if exploited can become vulnerabilities. This paper presents a comparison of several types of neural networks for forecasting the number of software bugs and vulnerabilities that will be discovered for a software component in certain timeframe, in terms of accuracy, trainability and stability to configuration parameters.

**Keywords:** Security, Software vulnerabilities, Forecasting, Neural networks.

## 1 Introduction

The frequency of cyber-attacks has been rising rapidly lately, which is a major concern. Because each attack exploits one or more vulnerabilities of the software components that make up the targeted system, their number of vulnerabilities is an indication of the system level of trust. In addition to vulnerabilities, the security of a component can also be affected by software bugs, as they can turn into weaknesses, which if exploited can become vulnerabilities. To forecast the cyber-attacks, first, we need to know the trends of the vulnerabilities and bugs that can involve other weakness.

This paper presents a comparison of several types of Neural Networks (NN) for forecasting the number of software bugs and vulnerabilities that will be discovered for a software component in certain timeframe, in terms of accuracy, trainability and

---

Technical University of Cluj Napoca, North University Centre of Baia Mare, e-mail: <sup>1</sup>{ovidiu.cosma, petrica.pop, cosmin.sabo}@mi.utcluj.ro, <sup>2</sup>laura.ov.cosma@student.utcluj.ro

stability to configuration parameters. The experimental part of this paper covers the Ubuntu Operating System and the Robot Operating System (ROS) which represents a set of software libraries and tools needed to build a robot application. The most complete repository that offers detailed information about vulnerabilities and weaknesses is the National Vulnerability Database (NVD), which is the U.S. government repository of standards-based vulnerability management data, represented using the Security Content Automation Protocol (SCAP). This data enables automation of vulnerability management, security measurement, and compliance [11]. NVD includes databases of security checklists, security related software flaws, misconfigurations and product names.

Regarding ROS vulnerabilities, the number of resources is very limited, but ROS is running under Ubuntu OS, and the most of its vulnerabilities affect ROS. The project ROSIN [12], [13] identified more than 200 bugs associated with ROS, that cover a reliable time interval, and these have been used to train our NN models.

This paper is organized in six chapters, as follows: Section 2 presents a literature review regarding vulnerabilities forecasting, Section 3 describes the neural network models we have compared in our experiments, Section 4 presents the data collection used for training our models, Section 5 presents the experimental results, and Section 6 presents conclusions and new research directions.

## 2 Literature Review

Several methods have been proposed to forecast the ICT system vulnerabilities. The proposed methods can be classified into three classes: time series analysis-based models, artificial intelligence models and statistical based models. Next, we will review some of the most methods described in the literature.

The most important time series models introduced to forecast the vulnerabilities have been considered by Gencer and Basciftci [2] who used the Auto Regressive Moving Average (ARIMA) model and deep learning methods in the case of android operating system. Pokhrel et al. [4] described a vulnerability analytic prediction model of Desktop Operating System based on linear and non-linear approaches using time series analysis. Roumani et al. [5] used an exponential smoothing model for vulnerability analysis and prediction.

Some of the recent models for forecasting the vulnerabilities have Long Short-Term Memory (LSTM) cells in their composition and are trained by Gradient Descent and Back Propagation (BP-GD) algorithms. A comparative analysis of several types of deep neural networks was described by Kaushik et al. [3]. The conclusion drawn in [3] was that the Convolutional Neural Networks (CNN), Multilayer Perceptron Models (MLP), Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) models perform well for one step forecasting and less satisfactory for multiple steps forecasting.

Rahimi and Zargham [7] described a novel paradigm for vulnerability discovery prediction based on code properties, called vulnerability scrying. Their proposed

method extracts code complexity and quality properties from a source code and then uses a stochastic model to forecast vulnerabilities. Williams et al. [8] described an integrated data mining framework that depicts automatically how the vulnerabilities evolve over time and detect the evolution of a particular vulnerability. In addition, their described framework has a predictive component that may be used to predict vulnerabilities or to approximate future appearance probabilities of vulnerability groups.

Yasasin et al. [6] and Roumani et al. [5] evaluated various methodologies to forecast vulnerabilities of well-known applications. Both works have concluded that the ARIMA model is the most suitable, since it offers the highest accuracy. More recently, Cosma et al. [1], presented a comparative study of the most important and promising methods for forecasting the ICT systems vulnerabilities.

### 3 Neural Network Models

We used three Types of Neural Network Models for forecasting the number of bugs and vulnerabilities of software components: Long Short Term Memory (LSTM), MultiLayer Perceptron (MLP) and Convolutional Neural Network (CNN). Their general architecture shown in Figure 1 has been adjusted based on preliminary experiments. In order to deliver good results, the models must have sufficient complexity to assure proper learning capacity, but unfortunately complex models are difficult to train, and they quickly enter in over-training. The LSTM model has been equipped with a dropout layer to mitigate the overtraining phenomenon.

All the models have been trained with the Gradient Descent - Back Propagation algorithm. Because there is little data available for training the models, we used the k-fold cross-validation technique. The model parameters and the GD-BP algorithm parameters have been fine-tuned based on grid search. The configuration parameters of the models on which our experiments are based, are shown in Figure 1. The GD-BP algorithm was configured with the following parameters: batch sizes: 9 and 12, learning rates: 0.005 and 0.006.

### 4 Data Collection

In this section we present the information sources we have used in our experiments. The collected data was used for training our NN forecasting models, as is going to be shown in the next section. We retrieved information regarding the Ubuntu operating system vulnerabilities from the National Vulnerability Database (NVD), and information regarding the Robot Operating System (ROS) bugs from the Robust-Rosin repository [12]. The structure of our vulnerabilities data collection is shown in Figure 2. It was defined based on the NVD data structure, transformed in a

JSON format, with some modifications to enable easier identification of relevant information.

The information related to the ROS operating system bugs are taken from the ROSIN project [13] and its public GitHub repository [12]. The repository contains software bugs information in Yet Another Markup Language (YAML) [14] format. In order to be used in our experiments, we scanned the files of the project, we extracted all its YAML files and transformed the content to a JSON format consistent with our data collection.

Finally we collected 3038 Common Vulnerabilities and Exposures (CVE) regarding Ubuntu operating system, which are related to different versions and cover a wide time period, and 220 ROS operating system related bugs.

## 5 Experimental results

In our experiments, we have built two sets of forecasting models: one for the Ubuntu operating system vulnerabilities, and the other one for the Robot Operating System (ROS) bugs. The models were developed in Python language, using the Keras deep learning API [9]. Each set was built to performed four types of forecasts: the number

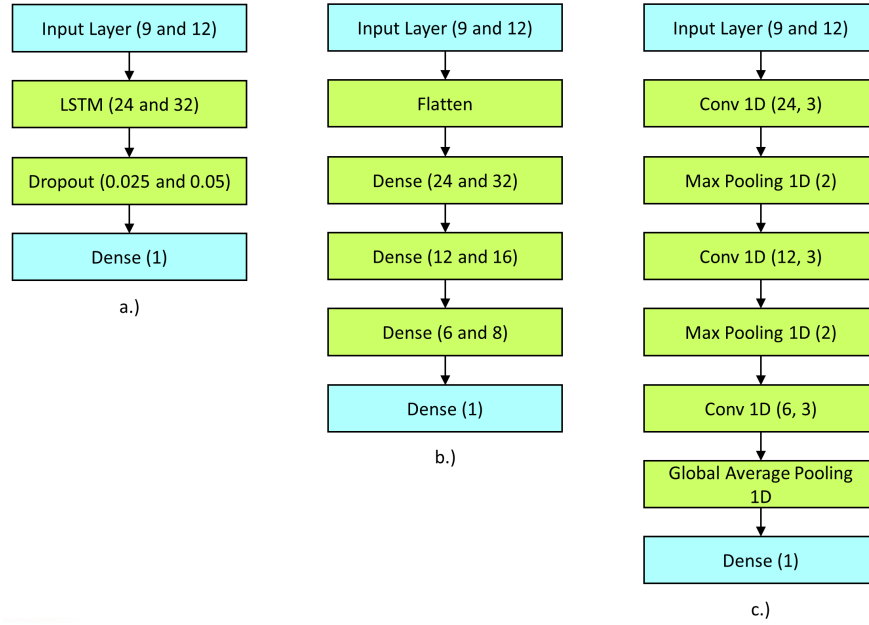


Fig. 1: Model architecture: a. Long Short Term Memory, b. Multilayer Perceptron, c. Convolutional Neural Network

of bugs / vulnerabilities in the next month, and the average number of monthly bugs / vulnerabilities in the next 2, 3 and 6 months. Each type of forecast was performed using three different neural network models: LSTM, MLP and CNN. Each model was built, trained, tested and validated multiple times in a grid-search algorithm, in order to fine-tune its parameters.

The training data for the models in the Ubuntu set was taken from the National Vulnerability Database [11], and from the Ubuntu version history [10]. The input features are the monthly vulnerabilities and the age of the last version (in months), or the averages, depending on the type of forecast.

The training data for the models in the ROS set was taken from the ROBUST-ROSI data set [12]. This data set contains both vulnerabilities and warnings. There are 4 levels of warnings, labeled as *"not-a-bug"*, *"bad-smell"*, *"bad-style"* and *"warning"*. The bugs are labeled by severity as *"minor-issue"* and *"error"*. We associated severity scores from 0 to 3 to the 4 warning levels, based on which we calculated cumulative warning levels for each month. The input features for the models in the ROS set are the number of monthly bugs and the warning level, or the averages, depending on the type of forecast. The input features for the 6 months average number fo monthly bugs forecasting models are presented in Figure 3. It can

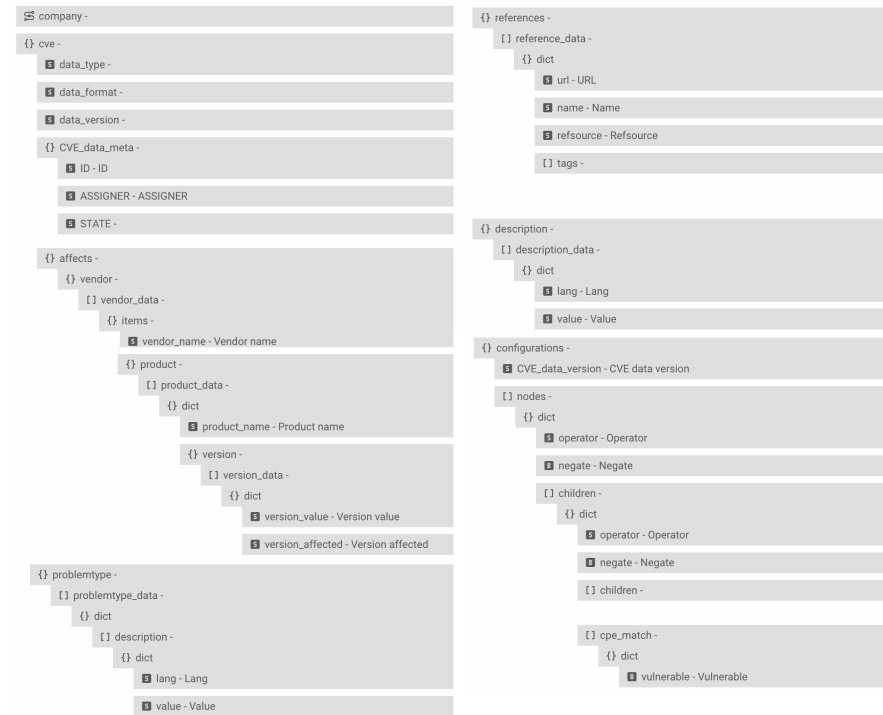


Fig. 2: ROS average number of vulnerabilities per month and average warning level

be seen that there is a correlation between the warning level and the number of bugs that will be discovered in the next period.

For increasing the forecasts accuracy, all the input data was normalized using Z-score normalization, in order to bring the average to 0 and the standard deviation to 1.

The results given by the best forecasting models in the Ubuntu set are shown in Figures 4 - 7. The forecasts cover a period of about 200 months, (from 2005-10 to 2022-04). The forecasting accuracy expressed by the Mean Absolute Error (MAE) computed for the entire period is situated between 3.53 for the 1 month model, and 1.98 for the 6 months average model, which is good, having in mind that the actual value ranges are  $[0, 133]$  and  $[0, 80]$  respectively.

A total of 24 forecasting models were built and tested: 12 in the Ubuntu set, and 12 in the ROS set. A comparison of their performance in terms of accuracy is shown in Figure 8. The MLP model consistently gave the best results for all the forecast types performed by the ROS set models. In the case of the Ubuntu set models, the best results were given by the CNN model for the 2, 3 and 6 months forecasts, and by the MLP model for the 1 month forecast. The LSTM model took the third place in each of the cases.

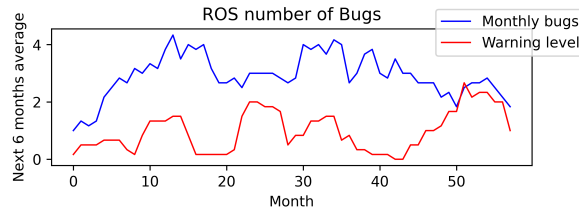


Fig. 3: ROS average number of vulnerabilities per month and average warning level

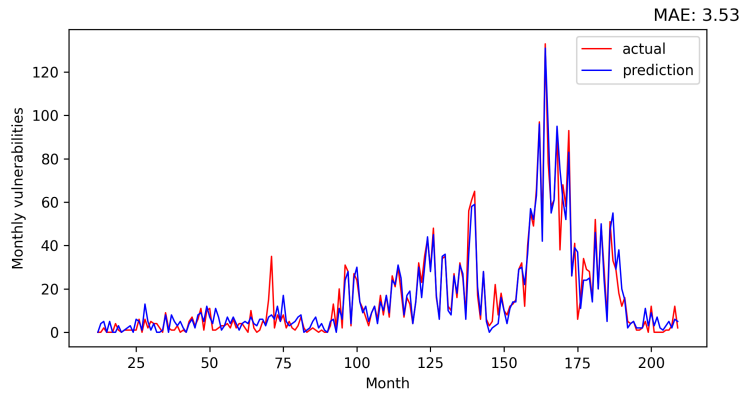


Fig. 4: Ubuntu monthly vulnerabilities forecasting

A complete model specification (CMS) is composed of the model architecture and all its configuration parameters, including the ones referring to the GD-BP algorithm.

For each CMS we have built, trained and evaluated several models. The overall accuracy of each model is expressed by the Mean Absolute Error (MAE) of the forecasts performed for the entire data set.

Thus, a Model Accuracy Result Set (MARS) was determined for each CMS, which allows us to make a comparison of the models in terms of trainability and in terms of sensitivity to configuration parameters.

All the models were trained using the Gradient Descent – Back Propagation (GD-BP) algorithm, which can be easily trapped in a local minimum. If there were no local minimums, the GD-BP algorithm would find the best solution every time. But

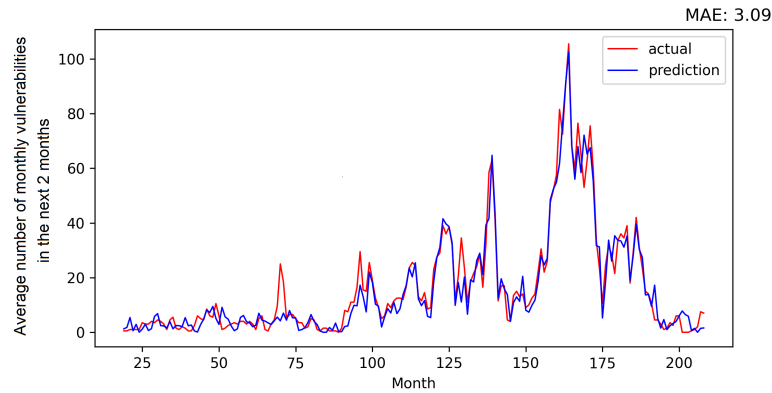


Fig. 5: Ubuntu vulnerabilities, 2 months average forecasting

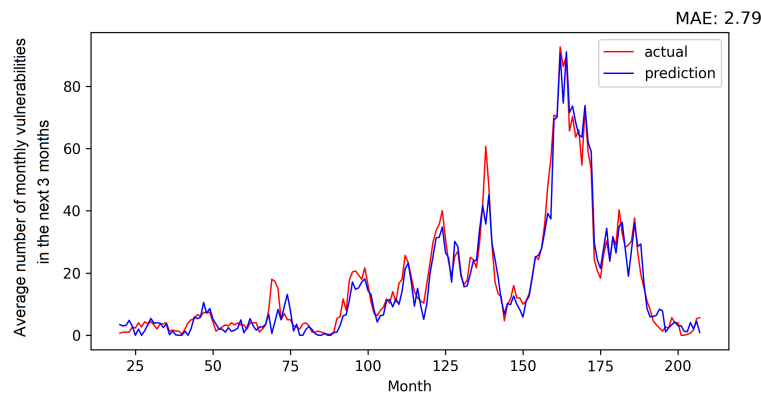


Fig. 6: Ubuntu vulnerabilities, 3 months average forecasting

this is not the case with real problems. The GD-BP algorithm usually ends in a local minimum, which depends on its parameters and the initialization of the model.

A certain Model Architecture (MA) can be considered easy to train, if when it is trained several times for the same CMS, similar results are obtained.

Thus, the MARS STandard Deviation (MARS-STD) calculated for all the models having the same CMS is an indication of the model architecture trainability.

A comparison between the model architectures in terms of trainability is presented in Figure 9. The plots represent the average MARS-STD for each model architecture in each model set. For easy comparison, the actual values were scaled to the  $[0, 100]$  range. The plots in Figure 9 show that the LSTM models have the best trainability. This is no surprise, because they were specially designed to successfully handle long sequences of data. The MLP models are situated at the other extreme. They are hard to train, because the chosen architecture suffers of the vanishing gradient problem, because it has multiple intermediate layers.

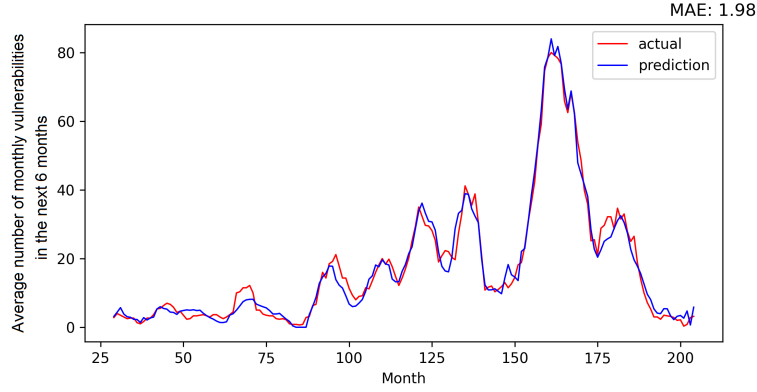
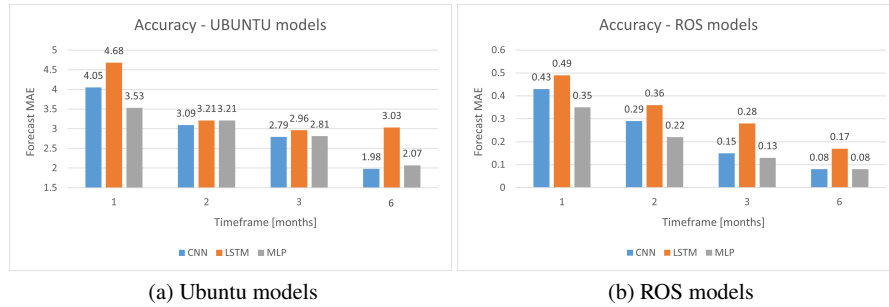


Fig. 7: Ubuntu vulnerabilities, 6 months average forecasting



(a) Ubuntu models

(b) ROS models

Fig. 8: Accuracy



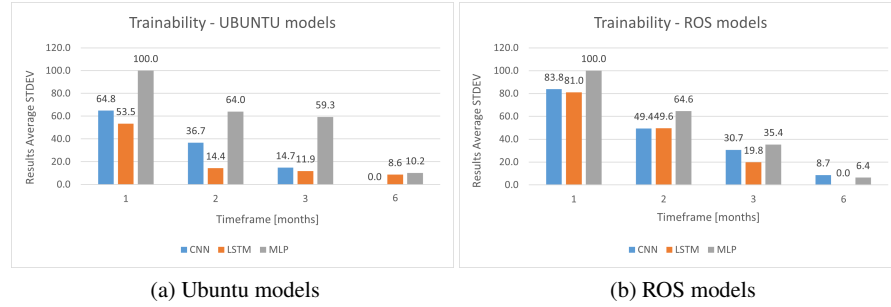


Fig. 9: Trainability

For evaluating the model architectures sensibility to the configuration parameters, we calculated the AVERAGE of each MARS (AV-MARS) first, and then the standard deviation of the AV-MARS for each model architecture in each model set. The results scaled to the range  $[0, 100]$  are presented in Figure 10. The CNN models seem to have the greatest sensibility to the configuration parameters (with three exceptions). This sensibility is not necessary a bad property, but it can be an indication that the fine-tuning of the configuration parameters might be more difficult. The least sensitive to configuration parameters seems to be the MLP architecture.

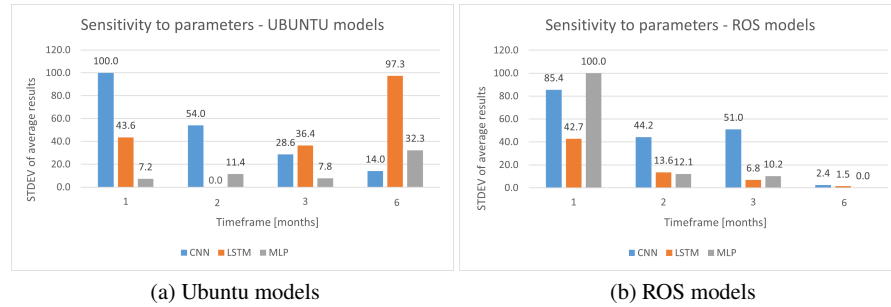


Fig. 10: Sensitivity to configuration parameters

## 6 Conclusions

In this paper, we presented a comparison of several types of neural networks for forecasting the number of software bugs and vulnerabilities that will be discovered

for a software component over a period of time, in terms of accuracy, trainability and stability to configuration parameters.

By analyzing the experimental results, the following conclusions can be drawn: In terms of forecasting accuracy, the best results were given by the CNN models in the case of vulnerabilities, and by the MLP models in the case of software bugs. In terms of trainability, the best results were given by the LSTM models, and in terms of stability to configuration parameters, the MLP models showed the best results. In our next research we will add new models to our study, and will develop a genetic algorithm for improving the trainability of the MLP models which are worst from this perspective.

**Acknowledgements** This work was supported by the project BIECO ([www.bieco.org](http://www.bieco.org)) that received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 952702, and by the UEFISCDI PN-III-P3-3.6-H2020-2020-0039, Contract 15/2020.

## References

1. Cosma, O., Macelar, M., Pop, P.C., Sabo, C., Zelina, I.: A Comparative Study of the Most Important Methods for Forecasting the ICT Systems Vulnerabilities. In: International Conference on Advanced Information Networking and Applications, 224-233, Springer, Cham (2021)
2. Gencer, K., Basciftci, F.: Time series forecast modeling of vulnerabilities in the android operating system using ARIMA and deep learning methods, Sustainable Computing In: Informatics and Systems, 30, 100515, (2021)
3. Kaushik, R., Shikhar Jain, Siddhant Jain, Tirtharaj Dash: Performance evaluation of deep neural networks for forecasting time-series with multiple structural breaks and high volatility. In: CAAI Transactions on Intelligence Technology, 1-16 (2021)
4. Pokhrel, N.R., Rodrigo, H., Tsokos, C.P.: Cybersecurity: Time Series Predictive Modeling of Vulnerabilities of Desktop Operating System Using Linear and Non-Linear Approach. In: Journal of Information Security, 8, 362-382, (2017)
5. Roumani Y., Nwankpa J.K., Roumani Y.F.: Time series modeling of vulnerabilities. In: Computers & Security, 51, 32-40 (2015)
6. Yasasin, E., Prester, J., Wagner, G., Schryen, G.: Forecasting IT security vulnerabilities – An empirical analysis. In: Computers & Security, 88, 101610 (2020)
7. Rahimi, S., Zargham, M.: Vulnerability Scrying Method for Software Vulnerability Discovery Prediction Without a Vulnerability Database. In: IEEE TRANSACTIONS ON RELIABILITY, 62(2), 395-407, (2013)
8. Williams, M.A., Barranco, R.C., Naim, S.M., Dey, S., Hossain, M.S, Akbar, M.: A vulnerability analysis and prediction framework. In: Computers & Security, 92, 101751 (2020)
9. Keras. <https://keras.io/>
10. Canonical: UBUNTU releases. <http://releases.ubuntu.com/>
11. National Institute of Standards and Technology: National Vulnerability Database. <https://nvd.nist.gov/>
12. robust-robin: ROBUST ROS Bug Study. <https://github.com/robust-robin/robust>
13. ROS-Industrial Quality-Assured Robot Software Components <https://www.rosin-project.eu/>
14. YAML.org: Yet Another Markup Language (YAML) 1.0.

# **Special Session on Intelligent Solutions for Cybersecurity Systems**

# Reinforcement Learning model free with GLIE Monte-Carlo on policy update for network topology discovery

Roberto Casado-Vara, Marcos Severt, Ángel Martín del Rey, Héctor Quintián and Jose L. Calvo-Rolle

**Abstract** Providing cybersecurity for computer networks is one of the main concerns for companies in this digital society. Unless this is done, companies can potentially lose huge amounts of data or even lose control of their own computer networks. Knowing the topology of the computer network and the information that is accessible on each of the nodes of the network is very relevant information both to establish impenetrable cyber defenses and to spread malware through the network and take control. In the proposed work, an algorithm has been designed to control the propagation of a malware through an unknown computer network in order to extract the information of its network topology. The results of this new algorithm have been tested in 3 simulations on a virtual copy of the same real computer network of an intelligent building in the city of Salamanca. The main result obtained was that the algorithm is able to discover all the nodes of the network adapting itself to the network characteristics.

**Key words:** Malware propagation, network discovery, Reinforcement learning.

---

Roberto Casado-Vara

Universidad de Burgos, Department of Mathematics and Computation, Burgos, Spain. e-mail: rc-casado@ubu.es

Marcos Severt

Universidad de Salamanca, Spain. e-mail: marcos\_ss@usal.es .

Ángel Martín del Rey

Universidad de Salamanca, Department of Applied Mathematics, Salamanca, Spain. e-mail: del-rey@usal.es

Héctor Quintián

University of A Coruña, CTC, CITIC,

Department of Industrial Engineering, Ferrol, A Coruña, Spain. e-mail: hector.quintian@udc.es

Jose L. Calvo-Rolle

University of A Coruña, CTC, CITIC,

Department of Industrial Engineering, Ferrol, A Coruña, Spain. e-mail: jlcalvo@udc.es

## 1 Introduction

Nowadays, computer networks have a global implementation in all areas, which is the reason for their increasing complexity. This is why generating a topology network is becoming an increasingly difficult but extremely useful task for accurate simulations. Discovering and monitoring the network is an essential task [1]. Moreover, given the dynamic and large-scale behavior of today's networks, the discovery process has to be fully automatic and provide correct results in the shortest possible time. In fact, existing tools and methods for discovering different topologies are of growing interest to application vendors and network administrators, as they are extremely useful for planning and managing any network, no matter how large it will be.

Knowing the network topology is essential to understand how the different devices that share information within the network behave. The importance of having an automatic, efficient and reliable method for the discovery of the network topology, lies in the fact that the detection of errors and troubleshooting is faster allowing to keep our network clean and safe from possible threats. Currently there are several WAN discovery researches that ignore the need to know topology information at the LAN level even though some works have shown that this information is relevant but they assume full knowledge of the routers to each other [2, 3]. Nowadays and with the rise of technologies such as IoT, there are numerous studies on how to discover the topology at different levels, i.e. at the WAN, LAN and physical level. These studies propose a method based on the different communication protocols that exist at these three levels such as, for example [4, 5]. On the other hand, instead of using multiple protocols, there are numerous studies based on the use of a single protocol at a single network level, such as Simple Network Management Protocol (SNMP) [6] or Link Layer Discovery Protocol (LLDP) [7]. If we focus on the network level, another of the most used protocols for the discovery of network topologies is Internet Control Message Protocol (ICMP), which is used by the different devices connected to the network to send error messages and operational information indicating success or failure when communicating with another IP address [8]. The latest experiments conclude that the topology discovery algorithm with multiprotocol fusion has better results and adaptability to topology changes [9]. As for the exploration of the graph that is generated in the network as it is discovered, there are different approaches such as those based on classical graph theory algorithms [10, 11]. Another of the most widely used methods for graph exploration in computer networks today, given their complexity and dynamism, is the use of different machine learning techniques that adapt to the changing topology of the network [12, 13]. This latest paradigm for machine learning-based graph exploration has been enhanced by the incorporation of advanced deep learning techniques [14].

Our research proposes a new strategy for network topology discovery using Markov Decision Process (MDP) and complex network-based models. The novelty of this research is to assume that there is no previous knowledge of the network and that the stochastic process of finding network data is modeled by a MDP. This approach

will allow network administrators or application providers to learn the topology of an unknown network dynamically, automatically and at a low computational level. This new strategy for the discovery of the topology of a network has 2 stages: in the first stage, the definition of the MDP that defines the possible states in a node of the topology graph is performed; in the second stage, a graph exploration algorithm is applied to find new nodes in the hidden network. The results of this research, even in their early stages, are relevant since the algorithm allows us to discover the topology of the network by defining the way in which the malware navigates the network. This proposed algorithm improves on the current state of the art as it is able to adapt and learn as it moves through the network topology. This paper is organized as follows: The details of our proposal are presented in Section 2. Section 3 presents three simulations, the results of which validate the performance of the proposal. Finally, Section 4 concludes the conducted research and proposes future lines of work.

## 2 Markov Decision Process

### 2.1 Preliminaries

An MDP is denoted by a tuple  $(S, A, R, P, s_0)$  where  $S$  represents the state space,  $A$  denotes for permitted actions,  $R$  gathers rewards,  $P$  stands for a state transition matrix, and  $s_0$  is the initial state. Given a state  $s_t = s \in S$  and an action  $a_t = a \in A$  at time  $t$ , the probability of reaching state  $s'$  at time  $t + 1$  is defined by the transition matrix  $P(s, a, s')$ , which denoted as

$$P(s, a, s') = P(S_{t+1} = s' | S_t = s, a_t = a) \quad (1)$$

Given an initial state  $s_0$ , the model runs continuously according to the dynamics of the environment defined by eq. 1 up to the point where it reaches a goal state. An MDP fulfills the Markov property, which essentially states that the future process is independent of the past given the present. Two kinds of value functions exist in MDPs, which are state value  $V(s)$  and state-action value  $Q(s, a)$ . Actions that an agent would take is a policy  $\pi$ , being a mapping from a state  $s$  and an action  $a$  to the probability  $\pi(a|s)$  of taking action  $a$  in state  $s$ . Thus the value function of a state  $s$  following the policy  $\pi$ , denoted as  $V_\pi(s)$  can be taken as the expecting future rewards, i.e.,

$$V_\pi(s) = E_\pi \left[ \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | s_t = s \right] \quad (2)$$

where  $\gamma$  is a discount factor. The state-action value of picking an action  $a$  at state  $s$  by following policy  $\pi$  is

$$Q_\pi(s, a) = E_\pi \left[ \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | s_t = s, a_t = a \right] \quad (3)$$

There have been several algorithms for solving the MDP, i.e., for determining the optimal policy, and the associated value functions, including the Monte Carlo method, dynamic programming method or the Q-learning approach, to name just a few [15, 16].

## 2.2 MDP for finding information in complex networks

In this section, we will develop an MDP model for finding information in complex networks with zero previous knowledge of the network.

### 2.2.1 States

In our model,  $s = (p_t, l_t, t)$  consist in three components, called, a graph position  $p \in P = \{\text{finite set of positions}\}$ ,  $l \in L = \{\text{permission levels}\}$ , and  $t \in T$  the current time. To determine what possibilities are once the exploratory malware arrives at a certain node, we create a finite set of permission levels  $L$  within the node itself, regardless of the type of equipment it is. In this research we have designed the different permission levels about the network that a user of a given computer connected to the network has. The set of permission levels is defined as follows: In the first level, which we will call base level, we assume that the equipment is simply reached and we do not have any kind of permission beyond being able to interact with it; The second level, which we will call guest level, assumes that we have guest level access to the specific computer, this allows us to obtain very limited information about it but it is an advance in terms of permissions over the previous level; The third level, which we will call basic permissions level, we assume that we have certain operational capacity within the equipment. This allows us to have access to certain information such as the equipment's network interfaces, IP addresses and network masks. This allows us with the use of network level protocols such as ICMP to know what other equipment exists within the topology; The last and fourth level, which we will call root level, allows us to have global access to the whole computer. This means having access to any type of command or to the use of network sniffers that would allow us to know the global traffic existing in this equipment and consequently those equipment that have direct communication with it. For example,  $s = (3, 2, 5)$  is a state which the malware is in  $3^{rd}$  node, with the guest level when time  $t = 5$ .

### 2.2.2 Actions

The set of allowed actions in this MDP is called  $A$ . These actions are variable depending on the number of neighbor nodes of the current node. In this situation, the agent has to choose between the neighbor nodes to move in. Notice that some of the movements may be non-reachable, we thus add a huge penalty with the reward to prevent the agent from taking the action which leads the agent to the non-reachable

neighboring network. The allowed movement is between two nodes linked by an edge. When moving to a new node in the topology, the pentesting action is performed to determine the permission status of that node. The pentesting consists of a set of tests that allow to determine security flaws within the system and that can allow, among other things, to reach a higher degree of privileges within the system. Notice that just as pentesting can involve a higher level of privileges, it can be detected and either expelled from the system or forced to reduce these privileges.

### 2.2.3 State transition

In this section we describe the state transition process that describes the states evolution process in this MDP. We claim in section 2.2.1 that the states in our MDP are a tuple  $s = (p_t, l_t, t)$ , where  $p$  is the position in the graph,  $l$  is the permission level and  $t$  is the current time. Despite the fact that  $t$  is changing following its own laws,  $p_t$  and  $l_t$  have some stochastic process in order to determine the next state. Although, one can think that  $p_t$  has an easy way to describe the state transition to  $p_{t+1}$ , under no circumstances will we know anything about the next nodes in the graph before we discover a new node. Since we reach a new node, we discover all of the new nodes linked to the current node. In this situation, under the current limitations, the state transition for  $p_t$  to  $p_{t+1}$  is controlled by the algorithm decisions. In the other hand, states that are reached when pentesting is being performed, where the probability of being in each state depends exclusively on the state in which it was in the previous pentesting attempt. When a pentesting test is performed, the probability of advancing in level decreases as the privileges of that level increase, and there is also the probability of remaining in the same level or of being expelled from the device, thus returning to the base level. Since the objective of the investigation is the discovery of the network topology, it has been assumed that there is always a probability of advancing from the base level to the level of maximum privileges.

### 2.2.4 Reward function

For RL based methods, blindness of exploration in unstructured environments is a major problem. Therefore, path planning problems may be inefficient and weakly robust. [17]. To deal with this problem, we will design the reward function considering all likely effects that will significantly affect the learning task of the algorithm. In an unstructured environment, the agent has to discover all the nodes of an unknown network. As a consequence, our reward function will include the number of nodes yet unknown, whether a node has been visited more than once and the time that has been spent since the start of the algorithm.

Inspired in the idea of the Kronecker delta, we design the number of unknown nodes remaining in the network at time  $t$ .  $U_n$  means the set of unknown nodes yet undiscovered at time  $t$ ,  $V_n$  indicates the set visited nodes more than once and  $n_t$



means the new node at time  $t$ . This function is modeled as shown in (4).

$$f_{unknown}(n_t, U_n, V_n) = \begin{cases} 100 & \text{if } n_t \in U_n \\ -10\sqrt{x} & \text{if } n_t \notin U_n \end{cases} \quad (4)$$

where  $x \in V_n$  indicates the number of times that a node has been visited.

By combining unknown nodes and repeatedly visited nodes, we describe the reward function as (5):

$$R(n_t, U_n, V_n, t) = f_{unknown}(U_n) - t \quad (5)$$

We use time as a constraint, since we want to model the risk of being detected by the network's cyberdefenses according to the time that the malware remains in the network discovering all its nodes.

### 2.2.5 Solving MDP

In the problem of discovering a network whose nodes are all unknown, the MDP model is either unknown or uncertain, but the experience can be sampled. Therefore, we can design a reinforcement learning algorithm with model free control approach. This type of method is the most optimal when the MDP is unknown or uncertain. Let  $V$  be the action value function and let  $\pi$  be the policy, we will update the policy evaluation with Monte Carlo policy evaluation, where  $V = v_\pi$ . So, we have to estimate  $v_\pi$ .

Since  $Q(s, a)$  is mode free, we face the estimation of  $v_\pi$  with greedy policy improvement

$$\pi^*(s) = \operatorname{argmax} Q(s, a) \quad \text{with } a \in A, s \in S \quad (6)$$

This problem is similar to solve the policy evaluation with Monte Carlo (i.e.  $Q = q_\pi$ ). Equivalently, starting from  $Q, \pi$ , with  $Q = q_\pi$  and  $\pi = \epsilon$ -greedy, we will reach  $q_*, \pi_*$  the optimal  $q$  and  $\pi$ . Every step, we use Greedy in the Limit with Infinite Exploration (GLIE), where all action-state are explored many times

$$\lim_{k \rightarrow \infty} N_k(s, a) = \infty \quad (7)$$

then the policy converges as

$$\lim_{k \rightarrow \infty} \pi_k(a|s) = 1 \quad (8)$$

with  $a = \operatorname{argmax} Q_k(s, a)$ .

We sample the  $k^{th}$  episode using  $\pi : \{S_1, A_1, R_1, \dots\} \sim \pi$ . For each state  $S_t$  and action  $A_t$  in every episode,

$$N(S_t, A_t) \leftarrow N(S_t, A_t) + 1 \quad (9)$$

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t, A_t)} (G_t - Q(S_t, A_t)) \quad (10)$$

where  $G_t$  is the accumulated reward,  $\epsilon \leftarrow \frac{1}{k}$ ,  $\pi \leftarrow \epsilon\text{-greedy}(Q)$ .

**Theorem 1** *GLIE Monte-Carlo control converges to the optimal action-value function*

$$\lim_{t \rightarrow \infty} Q(s, a) \rightarrow q_*(s, a) \quad (11)$$

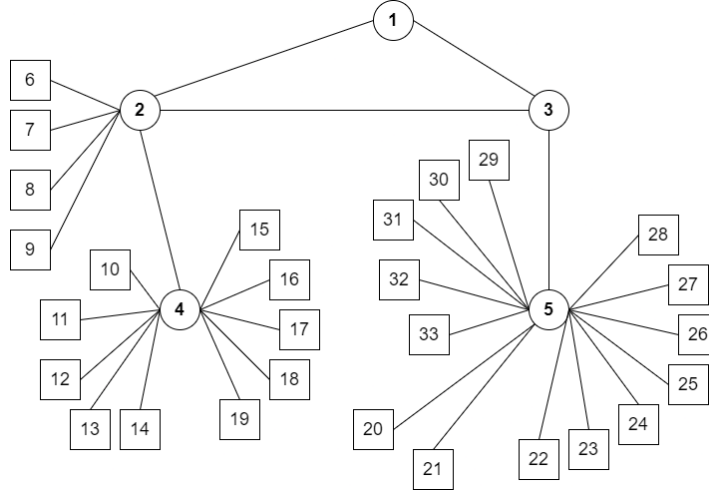
**Proof** See [15]. □

### 3 Performance evaluation

In this section, we evaluate through an experimental case study the proposed method for network topology discovery using reinforcement learning techniques and a markov decision process to model the acquisition of information by the intelligent agent at each node.

#### 3.1 Case study description

For this purpose, we use a real network topology made up of several subnetworks corresponding to a building to which we have had access for its study in order to be able to propose conditions as close to reality as possible. To test the proposed model, we have selected a non-trivial computer network of a real building. The chosen building is the University of Salamanca's R&D building and we have created a virtual network based on the real one. In this network topology we have two different types of computer equipment, on the one hand the routers and on the other hand any device that connects and communicates with the rest of the network. In this regard, it is important to note that the type of equipment has a significant influence when it comes to obtaining new information once we are performing the pentesting inside it. The network is constructed from the network topology formed by the different connected equipment in a building. Thus, a graph is constructed in which the vertices are different computers and the edges are the physical connections between them (i.e., there is no alteration of the connection between two nodes connected by an edge). An illustrative example of this graph can be found in figure 1, where the round figures are the routers and the squares are the rest of the computer equipment. This is how a network typology graph is constructed. This network will be considered as a complex network and will be of undirected type. Since the algorithm does not initially know this topology, it will build this adjacency matrix dynamically by adding more nodes as they appear after each information discovery process.



**Fig. 1** Graph of the network used in the case study. In circles the routers and in squares the computer equipment

### 3.2 Results

We will compare three selected iterations on the use case, where each of them is different according to the initial selected node for discovering the whole network. We compare the different results with the accumulated reward obtained during the discovery process of the network as a metric. We can thus evaluate the performance of the algorithm when starting on a certain computer rather than on another one. Furthermore, whenever the algorithm does not find new information in any of the nodes, it will automatically return to a visited node that is available from its current node in order to navigate and discover the entire topology of the network. For the simulations performed, we have chosen as initial nodes, node #14 belonging to the subnetwork formed by devices #10 to #14 and router #4; node #20 belonging to the subnetwork formed by devices #20 to #21 and router #5; and finally, node 6 belonging to the subnetwork formed by devices #6 to #9 and router #2. In this way, any of the building subnetworks represented in the topology of this use case is covered. In figure 2, we can see the comparative graph of the reward accumulated by the agent when applying the Reinforcement Learning algorithm for each of the three optimal simulations when starting at each node.

In table 1, we notice different statistics measurement calculated for the total accumulated rewards obtained in each of the three optimal simulations performed. In this way it is possible to compare whether or not there is a notable difference between starting at a given node and starting at another node belonging to another subnetwork.



**Fig. 2** Reward obtained in each of the three simulations carried out for each step.

Simulation	Start Node	Standard Deviation	Total Reward	Mean Reward	Total Steps
1	14	66,85	810,99	13,74	59
2	20	65,44	855,37	14,25	60
3	6	66,26	836,35	13,93	60

**Table 1** Statistical data collected from the three optimal simulations.

## 4 Conclusion

This paper has investigated the problem of discovering the whole nodes of a network dynamically and automatically. Using our algorithm, it is possible to move through the network topology and discover it by extracting new information from each node making this process dynamic and efficient. Moreover, this algorithm makes it possible to discover the network by previously defining how the agent has to identify every node, which increases the efficiency in achieving specific objectives within the network. On the other hand, we can conclude that we have found the optimal solution to the problem with the reward we have chosen above. Future work will focus on applying this algorithm to more complex networks consisting of more differentiated device types. Moreover, we will test the efficiency of this algorithm by applying new constraints to the network topology discovery and navigation process. Also, we would also want to test more complex functions to try to model reality more accurately. In future work, we want to test mathematical malware propagation model since with this algorithm we find out partial or total information about the network [18, 19, 20].

## References

1. Wang, C., Huang, N., Bai, Y., & Zhang, S. (2018). A method of network topology optimization design considering application process characteristic. *Modern Physics Letters B*, 32(07), 1850091.
2. Qaqos, N. N., Zeebaree, S. R., & Hussan, B. K. (2018). Opnet Based Performance Analysis and Comparison Among Different Physical Network Topologies. *Academic Journal of Nawroz University*, 7(3), 48-54.
3. Rana, A., Kumar, A., & Ali, H. (2021). An overview of the network topologies for enterprises. *Asian Journal of Multidimensional Research*, 10(10), 143-149.
4. Liu, Y., Wang, H., Cai, L., Shen, X., & Zhao, R. (2021). Fundamentals and advancements of topology discovery in underwater acoustic sensor networks: A review. *IEEE Sensors Journal*.
5. Zhou, S., Cui, L., Fang, C., & Chai, S. (2018, July). Research on network topology discovery algorithm for internet of things based on multi-protocol. In 2018 10th International Conference on Modelling, Identification and Control (ICMIC) (pp. 1-6). IEEE.
6. Zhang, X. (2018). An optimization algorithm of network topology discovery based on SNMP protocol. *Journal of Computer and Communications*, 6(01), 104.
7. Popic, S., Vuleta, M., Cvjetkovic, P., & Todorović, B. M. (2020, November). Secure topology detection in software-defined networking with network configuration protocol and link layer discovery protocol. In 2020 International Symposium on Industrial Electronics and Applications (INDEL) (pp. 1-5). IEEE.
8. Beverly, R. (2016, November). Yarrp'ing the Internet: Randomized high-speed active topology discovery. In *Proceedings of the 2016 Internet Measurement Conference* (pp. 413-420).
9. Tsai, P. W., Tsai, C. W., Hsu, C. W., & Yang, C. S. (2018). Network monitoring in software-defined networking: A review. *IEEE Systems Journal*, 12(4), 3958-3969.
10. Majeed, A., & Rauf, I. (2020). Graph theory: A comprehensive survey about graph theory applications in computer science and social networks. *Inventions*, 5(1), 10.
11. Bhushan, B., Sahoo, G., & Rai, A. K. (2017, September). Man-in-the-middle attack in wireless and computer networking—A review. In 2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA)(Fall) (pp. 1-6). IEEE.
12. Cheng, Y., Geng, J., Wang, Y., Li, J., Li, D., & Wu, J. (2019). Bridging machine learning and computer network research: a survey. *CCF Transactions on Networking*, 1(1), 1-15.
13. Qiao, L., Zhang, L., Chen, S., & Shen, D. (2018). Data-driven graph construction and graph learning: A review. *Neurocomputing*, 312, 336-351.
14. Georgousis, S., Kenning, M. P., & Xie, X. (2021). Graph deep learning: State of the art and challenges. *IEEE Access*, 9, 22106-22140.
15. Sutton, R. S., & Barto, A. G. (1998). Introduction to reinforcement learning.
16. Shou, Z., Di, X., Ye, J., Zhu, H., Zhang, H., & Hampshire, R. (2020). Optimal passenger-seeking policies on E-hailing platforms using Markov decision process and imitation learning. *Transportation Research Part C: Emerging Technologies*, 111, 91-113.
17. Xie, J., Shao, Z., Li, Y., Guan, Y., & Tan, J. (2019). Deep reinforcement learning with optimized reward functions for robotic trajectory planning. *IEEE Access*, 7, 105669-105679.
18. del Rey, A. M., Vara, R. C., & González, S. R. (2022). A computational propagation model for malware based on the SIR classic model. *Neurocomputing*, 484, 161-171.
19. Hernandez Guillen, J. D., Martin del Rey, A., & Casado-Vara, R. (2021). Propagation of the Malware Used in APTs Based on Dynamic Bayesian Networks. *Mathematics*, 9(23), 3097.
20. Guillen, J. H., Del Rey, A. M., & Casado-Vara, R. (2019). Security countermeasures of a SCIRAS model for advanced malware propagation. *IEEE Access*, 7, 135472-135478.

# Obfuscating LLVM Intermediate Representation Source Code with NSGA-II

Juan Carlos de la Torre<sup>✉</sup>, José Miguel Aragón-Jurado<sup>✉</sup>, Javier Jareño<sup>✉</sup>, Sébastien Varrette<sup>✉</sup> and Bernabé Dorronsoro<sup>✉</sup>

**Abstract** With the generalisation of distributed computing paradigms to sustain the surging demands for massive processing and data-analytic capabilities, the protection of the intellectual property tied to the executed programs transferred onto these remote shared platforms becomes critical. A more and more popular solution to this problem consists in applying *obfuscating* techniques, in particular at the source code level. Informally, the goal of obfuscation is to conceal the purpose of a program or its logic without altering its functionality, thus preventing reverse-engineering on the program even with the help of computing resources. This allows to protect software against plagiarism, tampering, or finding vulnerabilities that could be used for different kinds of attacks. The many advantages of code obfuscation, together with its low cost, makes it a popular technique. This paper proposes a novel methodology for source code obfuscation relying on the reference LLVM compiler infrastructure that can be used together with other traditional obfuscation techniques, making the code more robust against reverse engineering attacks. The problem is defined as a Multi-Objective Combinatorial Optimization (MOCO) problem, where the goal is to find sequences of LLVM optimizations that lead to highly obfuscated versions of the original code. These transformations are applied to the back-end pseudo-assembly code (*i.e.*, LLVM Intermediate Representation), thus avoiding any further optimizations by the compiler. Three different problem flavours are defined and solved with popular NSGA-II genetic algorithm. The promising results show the potential of the proposed technique.

---

Juan Carlos de la Torre

Superior School of Engineering, University of Cádiz, Spain.

e-mail: [juan.detorre@uca.es](mailto:juan.detorre@uca.es)

José Miguel Aragón-Jurado

Superior School of Engineering, University of Cádiz, Spain.

e-mail: [josemiguel.aragon@uca.es](mailto:josemiguel.aragon@uca.es)

Javier Jareño

Superior School of Engineering, University of Cádiz, Spain.

e-mail: [javi.jarenodorado@alum.uca.es](mailto:javi.jarenodorado@alum.uca.es)

Sébastien Varrette

Faculty of Science, Technology and Medicine, University of Luxembourg.

e-mail: [sebastien.varrette@uni.lu](mailto:sebastien.varrette@uni.lu)

Bernabé Dorronsoro

Superior School of Engineering, University of Cádiz, Spain.

School of Computer Science, The University of Sydney, Australia.

e-mail: [bernabe.dorronsoro@uca.es](mailto:bernabe.dorronsoro@uca.es)

## 1 Introduction

One of the major concerns for most organizations and companies nowadays is the protection of their information and developments, which is particularly sensible with the advent of distributed computing paradigms facilitating code sharing and remote executions. Software protection is usually achieved through the implementation of a number of security mechanisms at different levels in their information systems to avoid unauthorized accesses, tampering, or any other kind of attack.

Moreover, software could be used to track and find flaws or vulnerabilities in the exposed security system that could be used with malicious purposes. However, securing a software against attacks is a complex and tedious task. With the generalisation of Cloud Computing services and the explosion of novel Web Services developed over interpreted languages (such as Javascript, Python or Ruby), source code is often directly exposed. Furthermore, compiled software is vulnerable too, with more and more advanced reverse engineering techniques available to extract its source code [21]. In order to protect the source code of software programs, it is usual to rely on software *obfuscation* techniques. Informally, the goal of obfuscation is to conceal the purpose of a program or its logic without altering its functionality, thus preventing reverse-engineering on the program even with the help of computing resources. An illegible code is more robust against plagiarism and possible attacks. These issues, together with the fact that obfuscating a code is a relatively inexpensive process, makes it a popular technique among software producers to protect their codes. In practice, there is a large number of code obfuscation techniques in the literature [16]. Among them, only a few works propose the use of techniques based on Evolutionary Algorithms (EAs) for obfuscating code.

This work proposes a novel methodology for obfuscating code combining the reference LLVM<sup>1</sup> compiler infrastructure [22] and its Intermediate Representation (IR) together with Genetic Algorithms (GAs) [13]. More specifically, LLVM IR is a pseudo-assembly language that can be generated from a wide range of programming language such as C/C++ and Objective C (through the Clang frontend), Python (*via* the Numba JIT (Just In Time) compiler suite or the Python bindings for LLVM provided by `llvmpy`) or Julia, among others. The proposed methodology is based on the definition of the problem as a Multi-Objective Combinatorial Optimization (MOCO) one. For that, we will make use of the modular and reusable toolchain technologies featured within LLVM (in particular the set of optimization passes it provides). The problem is then to find a sequence of such passes so that the resulting code is *as much obfuscated* as possible, a concept which will be formalized later. Because the proposed technique applies optimization transformations, different to those typically used in obfuscation methods, it is complementary to the existing ones, and it can be applied to their result in order to get an even more obfuscated code. Therefore, the proposed method does not intend replacing existing obfuscation techniques, but complementing them for further obfuscations. In order to effectively

---

<sup>1</sup> Since December 2011, "LLVM" is officially no longer an acronym and simply a brand that applies to the LLVM umbrella project. For more information, see [www.llvm.org](http://www.llvm.org).

guide the search of the GA towards obfuscated codes, we study different definitions of the problem, based on several metrics that evaluate different aspects of the level of obfuscation of a code, taken from [5]. The problem flavours proposed here also consider the execution time of the resulting program under the assumption that obfuscating a code should not negatively impact on the user experience.

The main contribution of this work is thus the design of a novel approach to obfuscate software (in LLVM IR language) based on the application of source code optimization transformations. In addition, this article provides (i) the mathematical modelling of the problem of obfuscating software source code as a combinatorial optimization problem, (ii) the definition of three different multi-objective optimization problems for source code obfuscation, and (iii) the resolution of the three problems with NSGA-II [11], a well-known multi-objective optimization algorithm. The structure of the paper is as follows. Section 2 briefly presents the main techniques for software obfuscation, as well as some outstanding related works. After that, the problems proposed in this work are introduced in Section 3, and the technique used to solve it is described in Section 4. The configuration of the experiments and the obtained results are given in sections 5 and 6, respectively. Finally, the main conclusions from this study are drawn in Section 7.

## 2 Background

A large number of applications for source code obfuscation transformations exists in the literature [8, 3, 16]. They affect different aspects of a program structure and can be classified into three main categories: (1) *data obfuscation*, composed of all the operations that modify and obscure the data structures used in a program. This includes the expansion of constants, consisting in replacing constant values by arithmetic operations that give the same value, the encryption of variables and constants, for instance by applying some homomorphic operations to change constant values or arrays dimension, dead code insertion, the replacement of arithmetic operations by others, more complex, with the same result, or renaming variables and functions, among others; (2) *layout obfuscation*, embedding all the transformations changing the information induced by the code formatting. This includes for example scrambling techniques for identifier names or for the code indentation; (3) *flow control obfuscation* which affects the aggregation, ordering or computations performed within the program control-flow and thus modifies the structure of the software [18]. This includes operations as functions in- or out-lining, basic sequence codes destruction, *i.e.*, identifying basic blocks of instructions in the code and replace them by more complex ones, use of control instructions from the Operating System, as Structured Exception Handler or Vectored Exception Handler (Windows) or `set jmp` and `long jmp` (Linux), or the insertion of opaque predicates, for instance, code sequences with conditionals that are always true (or false), but its value can only be known in compilation time. Hybrid approaches exist and mix these different categories [9, 18]. In all cases, only a few of the existing obfuscation techniques make use of LLVM framework [22]. These works, such as Obfuscator-LLVM [18], benefit



from the easiness of LLVM to implement code transformation patterns. However, the present work follows a different approach, making use of the LLVM optimization passes for the purpose of obfuscating code, different to that for which they were conceived. This way, the method proposed here is complementary to the obfuscation operations existing in the literature. With regards the mathematical modelling of software source code obfuscation as an optimization problem, the literature is relatively limited. In [20], the authors make use of GAs to protect the abstract data of control flow. For that, they propose a novel metric to measure the control flow complexity, based on entropy theory. A multi-objective Genetic Programming (GP) algorithm was proposed for the obfuscation of C code in [6] and for JavaScript programs in [5]. GP makes arbitrary changes in the Abstract Syntax Tree (AST) of the source code, so the method cannot guarantee that the obfuscated software keeps the semantics of the original one. As objective functions, the authors analyze six different ones, three of them are used in this work. Finally, Petke presents in [25] the idea of using Genetic Improvement for code obfuscation, although it is not implemented. This is a similar approach to the one just described above [5, 6], given that Genetic Improvement is a kind of GP specially tailored for working with source codes.

The approach presented in this work considers a large number of transformations in the code, unlike [20], focused on only control flow modifications, while it ensures that the semantics of the program are unaltered, an important distinctive feature with respect to the other two EA-based approaches [5, 25]. In addition, it is not limited by the software size, as it is the case of the Genetic Programming based ones.

### 3 Problem Definition

Obfuscating a code is the art of modifying it to conceal its purpose or its logic without altering its functionality, thus preventing the tampering or the reverse engineering of the program [8]. It is a useful technique in order to protect the source code of a program, limiting plagiarism, reducing the vulnerabilities discovery probability, or securing sensible information, among others.

This work proposes the use of a set of generic code transformations to obfuscate a source program  $P$ , which is defined as a Combinatorial Optimization problem. Let  $T = t_1, \dots, t_k$  a set of generic source code transformations, which transform a source program  $P$  into a semantically equivalent one  $P'$  having the same *observable behaviour* [8]. More precisely, if  $P$  fails to terminate or terminates with an error condition, then  $P'$  may or may not terminate, otherwise  $P'$  must terminate and produce the same output as  $P$ . The problem is defined as finding a sequence of such transformations  $S = [s_1, \dots, s_N]$ ,  $s_i \in T$ , such that the defined obfuscation quality of the source code is maximized. Every  $s_i$  takes the value of the identifier of a transformation, and the sequence  $S$  defines the order in which the different transformations must be applied. This order is important because it directly influences the resulting output code. In addition, it is allowed for the same transformation to be applied more than once (*i.e.*, it can happen that  $s_i = s_j$ ,  $i \neq j$ ) since, for most

existing transformations, its repetitive application produces further changes in the source code, specially when they are applied in combination with others [27].

There is not a single metric that is accepted to accurately measure the level of obfuscation of a code [5, 24]. This work explores several metrics to measure different aspects of the code obfuscation quality as fitness function in the optimization process:

1. *Program Length (L)*: number of lines of the source code.
2. *Cyclomatic Complexity (CC)* [23]: it is a well-known metric to compute the complexity of a software by measuring the number of predicates it contains. It is computed on the Control Flow Graph (CFG) of the code as:  $CC = e - n + 2 \cdot b$ , where  $e$  and  $n$  are the number of edges and nodes of the CFG, respectively, and  $b$  is the number of basic blocks in the program *i.e.*, a code sequence with input branches only at the entry and without branches out, apart from the exit.
3. *Nesting Level Complexity (NLC)* [14]: in the CFG, the nesting depth of a branch node (representing a control structure) is defined as 1 plus the nesting depths of its parent control structure nodes, if any.

In addition, it is also important that the user experience with the obfuscated code is not worsened with respect to the original one. Therefore, we define the problem as multi-objective, where one objective is one of the metrics defined below, and the second one is the execution time. This way, solving the problem means finding accurate trade-off solutions with high obfuscation levels and quick execution times.

## 4 Evolutionary Multi-objective Optimization

Evolutionary Multi-objective Optimization (EMO) is a field of evolutionary computation where several conflicting objectives are to be optimized at the same time. Two objectives are in conflict if, for a given optimal solution, improving one of the objectives leads to worsening another one(s). The result of an EMO is not one single solution but a set of trade-off ones, called the *Pareto front* [7, 10].

In this work, the Non-dominated Sorting Genetic Algorithm II [11], or NSGA-II for short, is used to solve the proposed multi-objective problems. NSGA-II is among the most well-known algorithms for multi-objective optimization, and it is currently used in many different fields [2, 12, 15, 19, 26]. It extends the GA [13] paradigm which work on a set of individuals, called the population. Individuals are tentative solutions to the problem, therefore they encode the values of the problem decision variables. GAs iterate over the population of individuals in order to evolve it towards better solutions using the genetic operators: (i) *selection* of two parents from the population, (ii) *recombination* of the information in these two parents to generate some offspring solutions, (iii) *mutation* of the offsprings, consisting in performing some slight change in the value of some variable(s), and (iv) *replacement*, a mechanism that decides the individuals that will compose the population for the next iteration (also called generation). In NSGA-II, an auxiliary population (with the same size as the original one) is built in each generation by iteratively applying the genetic operators. Then, both the current and the auxiliary populations are merged

into one single new population of the same size for the next generation. The *Ranking* (an ordering of solutions in terms of how many other solutions they dominate) and *Crowding* (more isolated solutions in the current Pareto front have better fitness) processes are used to select the solutions for the next generation population.

## 5 Experimental Setup

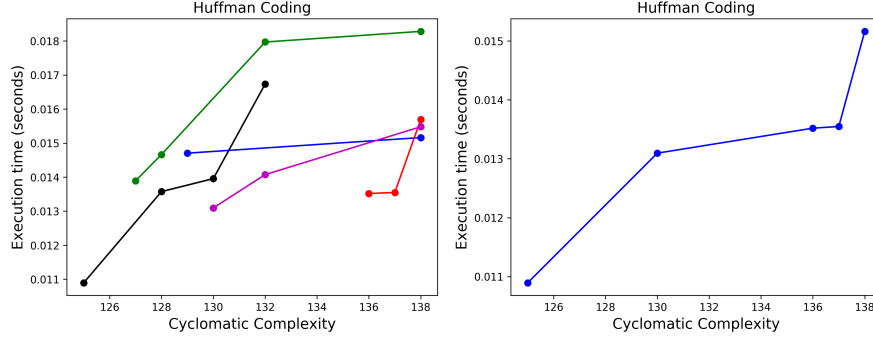
This work makes use of the modular LLVM compiler infrastructure [22] and, specifically, the source code transformations it provides among its tool chain technologies, called *passes*. These passes are the set of available transformations  $T$ , and it is composed of 87 optimizations in LLVM 9.0.1, the version that was used in this work. Before evaluating the quality of a solution, the sequence of transformations it contains must be applied to the original source code of the program to obfuscate it. This is done using the LLVM optimizer and analyzer (interfaced by the `opt` command). The first step is to generate the LLVM IR, a pseudo-assembly language. Then, the different transformations proposed by the solution under evaluation are applied on the LLVM IR code, in the order established by the solution. The binary file is created from LLVM IR code, and it is executed five times, and the median of these measurements is taken as the execution time objective of the fitness function.

As mentioned before, NSGA-II [11] is used to find accurate solutions to the different problem flavours defined. The selected implementation is the one provided in the JMetalPy framework [4], and the genetic operators originally proposed by the authors are used. After some preliminary experiments, we set the length of the chromosome of individuals to 20, meaning that this will be the allowed number of passes to apply, and the mutation and recombination probabilities were 0.2 and 0.8, respectively. Population size was set to 50 individuals, randomly initialized, and the termination condition is performing 10,000 fitness function evaluations. As a case of study, we chose three well known algorithms in computation<sup>2</sup>:

1. *AVL tree* [1]: an algorithm that performs an order relation of the data by building a balanced binary tree, so that the difference between the depths of the left-most and right-most branches is not higher than 1.
2. *Hamiltonian cycle*: given some graph, this algorithm finds a path that starts and ends in the same node, traversing the other nodes exactly once.
3. *Huffman coding* [17]: a data compression algorithm that represents symbols using binary codes of different lengths, in such a way that no code is the prefix of another. The most frequent symbols are assigned to shorter codes.

Because of the stochastic nature of NSGA-II algorithm, we performed in this paper five independent executions of the algorithm to obfuscate each of the three studied programs. They were tested on an Intel Xeon CPU E5-2620 v2 server before being performed on the Intel broadwell computing nodes featured by the Iris cluster within the HPC facilities of the University of Luxembourg [28].

<sup>2</sup> Available in: [github.com/jctor/Obfuscating-LLVM-Intermediate-Representation-Source-Code-with-NSGA-II-CISIS\\_2022](https://github.com/jctor/Obfuscating-LLVM-Intermediate-Representation-Source-Code-with-NSGA-II-CISIS_2022)



**Fig. 1** Example of how the best tradeoff solutions from the Pareto fronts found in every independent run (on the left-hand side) are compiled into a single Pareto front approximation (right-hand side).

## 6 Validation and Experimental Results

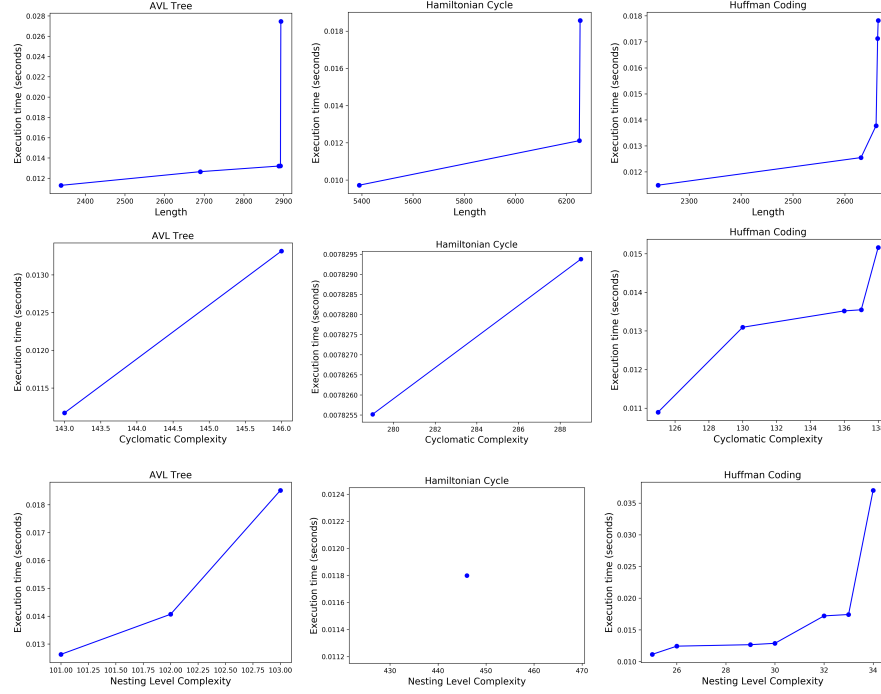
Five runs of NSGA-II algorithm are executed to obfuscate each of the studied programs, considering the three problem flavours previously described in Section 3. We build a Pareto front of the best tradeoff solutions by taking the best *non-dominated* solutions from the obtained Pareto front approximations in every run [12]. This process is represented in Fig. 1, where the plot on the left-hand side shows the Pareto front approximations obtained in the five independent runs, in different colours, and the non-dominated solutions out of these fronts are used to generate the Pareto front approximation shown on the right. In this front, the first solution (from left to right) belongs to the black front, the second to the purple one, the two next ones are from the red front, and the last one is obtained from the blue front.

In practice, the best non-dominated solutions found in the five independent runs of the algorithm carried out in every case are shown in Fig. 2. As it can be seen, the number of non-dominated solutions is low in all cases, ranging from 7 solutions (optimization of runtime and nesting complexity on Huffman coding program) to only 1 (optimization of the same two objectives as before, but on Hamiltonian cycle program). The values of the metrics for the original programs are shown in Table 1.

The proposed methodology finds obfuscated programs outperforming the original one in most cases (except for cyclomatic complexity metric on AVL Tree and Huffman Coding). The *Nesting Level Complexity* (NLC) metric within the obtained obfuscated programs was enhanced by as much as 345% in the best case (for Hamiltonian Cycle program), while the obtained improvements are 22,62% and 30,77%

**Table 1** Characteristics of the original studied programs selected for the obfuscation process.

Case Study ( <i>original</i> )	Program Length (L)	Cyclomatic Complexity (CC)	Nesting Level Complexity (NLC)	Runtime (ms)
AVL Tree [1]	2,860	146	84	7
Hamiltonian Cycle	6,049	282	100	7
Huffman Coding [17]	2,611	138	26	6



**Fig. 2** Best tradeoff solutions found for every program and metric after the five independent runs.

for AVL Tree and Huffman Coding, respectively. The high number of nested loops and conditional branches present in the source code of Hamiltonian Cycle is the reason for the major increase on its nesting complexity. This outstanding solution could outperform all other solutions found during the experimentation. However, the problem seems to be multi-objective, as evidenced for the other programs. However, the method shows good performance for this metric, considerably improving the nesting complexity of AVL Tree program, despite the fact that it does not have nested conditional branches.

Regarding the *program length* metric (L), our methodology finds versions of the problem with 33, 203 and 53 more lines than the original ones, for AVL Tree, Hamiltonian Cycle and Huffman Coding, respectively, representing an increase of 1,14%, 3,37%, and 2,03% on their number of lines. The proposed method finds difficulties to improve the *Cyclomatic Complexity* (CC) of the different programs, being only able to improve it for Hamiltonian Cycle, for which a solution was found with a value of 289 for this metric, 7 units higher (*i.e.*, a 3% increase) than the value of the original program. For the other two programs, the most obfuscated codes kept the same cyclomatic complexity as their original counterparts. Finally, the *runtimes* of the obtained obfuscated programs are in general longer with respect to the original programs, but they are below 0.04 seconds in all cases. Some additional experiments confirmed that the time increase observed in solutions from the Pareto front is due to the parallel executions the genetic algorithm is performing to evaluate solutions.

## 7 Conclusions and Future Work

This work presents a novel methodology for software code obfuscation which can be used together with other obfuscation methods to achieve highly obfuscated programs. The method relies on the LLVM compiler infrastructure and exploits the LLVM Intermediate Representation (IR) pseudo-assembly code generated from the input program, avoiding this way any further out-of-control code transformations performed at compilation time. The problem is defined as a multi-objective combinatorial optimization one, and it consists on finding sequences of LLVM optimizations (called *passes*) that lead to highly obfuscated codes. Three different metrics have been studied to measure the quality of the code obfuscation (*i.e.*, the number of lines, the cyclomatic complexity, and the nesting complexity), and three flavours of the problem are defined, each using one of these metrics together with the obfuscated program execution time. They are solved with the well-known NSGA-II algorithm. Results show how the proposed method is able to significantly improve the number of lines and the nesting complexity of the programs for the three cases of study, but the cyclomatic complexity could only be improved for one of them. Regarding the computation time, they increased for the obfuscated programs in the Pareto fronts, but they were always under 0.04 seconds. Additional experimentation confirmed that this increase is due to the parallel executions made by the genetic algorithm.

As future work, we plan to study how the proposed method can be combined with other obfuscation techniques to improve the quality of the resulting code, as well as evaluating its impact on the result after applying reverse engineering techniques. In addition, it is important to investigate on more meaningful metrics that better quantify the level of obfuscation of the code. Finally, it will be interesting to study the impact of the proposed methodology on other more complex and longer programs than the studied ones in this paper, and to enhance the methodology for improving the results on the cyclomatic complexity if they remain marginal on the larger programs too.

**Acknowledgements** This work was supported by Junta de Andalucía and ERDF under contract P18-2399 (GENIUS), the Ministerio de Ciencia, Innovación y Universidades and the ERDF (iSUN – RTI2018-100754-B-I00), and ERDF (OPTIMALE – FEDER-UCA18-108393). J.C. de la Torre acknowledges the Ministerio de Ciencia, Innovación y Universidades for the support through FPU grant (FPU17/00563). B. Dorronsoro acknowledges “ayuda de recualificación” funding by Ministerio de Universidades and the European Union-Next GenerationEU. The experiments presented in this paper were carried out using the HPC facilities of the University of Luxembourg [28] [hpc.uni.lu](http://hpc.uni.lu)

## References

1. Adelson-Velskii, G.M., Landis, E.M.: An algorithm for the organization of information. *Soviet Mathematics Doklady* **3**, 1259–1263 (1962)
2. Al-Rashed, A.A., Alsarraf, J., Alnaqi, A.A.: Exergy optimization of a novel hydrogen production plant with fuel cell, heat recovery, and MED using NSGAII genetic algorithm. *International Journal of Hydrogen Energy* (2022)
3. Behera, C.K., Bhaskari, D.L.: Different obfuscation techniques for code protection. *Procedia Computer Science* **70**, 757–763 (2015). DOI 10.1016/j.procs.2015.10.114. Proceedings of the 4th Int. Conference on Eco-friendly Computing and Communication Systems

4. Benítez-Hidalgo, A., Nebro, A.J., García-Nieto, J., Oregi, I., Del Ser, J.: jMetalPy: A Python framework for multi-objective optimization with metaheuristics. *Swarm and Evolutionary Computation* **51**, 100598 (2019). DOI 10.1016/j.swevo.2019.100598
5. Bertholon, B., Varrette, S., Bouvry, P.: JShadObf: A JavaScript Obfuscator based on Multi-objective Optimization Algorithms. In: Proc. of the IEEE Intl. Conf. on Network and System Security (NSS 2013), *LNCS*, vol. 7873, pp. 336–349. Springer Verlag, Madrid, Spain (2013)
6. Bertholon, B., Varrette, S., Martinez, S.: ShadObf: A C-source Obfuscator based on Multi-objective Optimization Algorithms. In: 27th IEEE/ACM Intl. Parallel and Distributed Processing Symposium (IPDPS 2013), pp. 435–444 (2013)
7. Coello, C., Lamont, G.B., van Veldhuizen, D.A.: *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer (2007)
8. Collberg, C., Nagra, J.: *Surreptitious Software: Obfuscation, Watermarking, and Tamperproofing for Software Protection*, 1st edn. Addison-Wesley Professional (2009)
9. Dang, B., Gazet, A., Bachaalany, E., Josse, S.: *Practical Reverse Engineering: x86, x64, ARM, Windows Kernel Reversing Tools, and Obfuscation*. Wiley (2014)
10. Deb, K.: *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley (2009)
11. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* **6**(2), 182–197 (2002)
12. Dorronsoro, B., Ruiz, P., Danoy, G., Pigné, Y., Bouvry, P.: *Evolutionary Algorithms for Mobile Ad Hoc Networks*. Nature-Inspired Computing series. Wiley/IEEE Computer Society (2014)
13. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley (1989)
14. Harrison, W., Magel, K.: A complexity measure based on nesting level. *SIGPLAN Notices* **16**(3), 63–74 (1981)
15. He, C., Ge, D., Yang, M., Yong, N., Wang, J., Yu, J.: A data-driven adaptive fault diagnosis methodology for nuclear power systems based on NSGAII-CNN. *Annals of Nuclear Energy* **159**, 108326 (2021)
16. Hosseinzadeh, S., Rauti, S., Laurén, S., Mäkelä, J.M., Holvitie, J., Hyrynsalmi, S., Leppänen, V.: Diversification and obfuscation techniques for software security: A systematic literature review. *Information and Software Technology* **104**, 72–93 (2018)
17. Huffman, D.A.: A method for the construction of minimum-redundancy codes. *Proceedings of the IRE* **40**(9), 1098–1101 (1952). DOI 10.1109/JRPROC.1952.273898
18. Junod, P., Rinaldini, J., Wehrli, J., Michielin, J.: Obfuscator-LLVM – Software Protection for the Masses. In: *Int. Workshop on Software Protection*, pp. 3–9. IEEE (2015)
19. Kar, M.B., Kar, S., Guo, S., Li, X., Majumder, S.: A new bi-objective fuzzy portfolio selection model and its solution through evolutionary algorithms. *Soft Computing* **23**, 4367–4381 (2019)
20. Kim, J.I., Lee, E.J.: A technique to apply inlining for code obfuscation based on genetic algorithm. *Journal of Information Technology Services* **10**(3), 167–177 (2011)
21. Linn, C., Debray, S.: Obfuscation of executable code to improve resistance to static disassembly. In: *10th ACM Conf. on Computer and Communications Security*, p. 290–299. ACM (2003)
22. LLVM: The LLVM Compiler Infrastructure. <https://llvm.org/>
23. McCabe, T.: A complexity measure. *IEEE Trans. on Software Eng.* **SE-2**(4), 308–320 (1976)
24. Mohsen, R.: *Quantitative measures for code obfuscation security*. Ph.D. thesis, ICL (2016)
25. Petke, J.: Genetic improvement for code obfuscation. In: *Genetic and Evolutionary Computation Conference Companion*, p. 1135–1136. ACM (2016)
26. Santiago, A., Dorronsoro, B., Fraire, H.J., Ruiz, P.: Micro-genetic algorithm with fuzzy selection of operators for multi-objective optimization:  $\mu$ FAME. *Swarm and Evolutionary Computation* **61**, 100818 (2021)
27. de la Torre, J.C., Ruiz, P., Dorronsoro, B., Galindo, P.L.: Analyzing the influence of LLVM code optimization passes on software performance. In: *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Applications*, pp. 272–283. Springer (2018)
28. Varrette, S., Cartiaux, H., Peter, S., Kieffer, E., Valette, T., Olloh, A.: Management of an Academic HPC & Research Computing Facility: The ULHPC Experience 2.0. In: *Proc. of the 6th ACM HPC and Cluster Technologies Conf. (HPCCT 2022)* (2022)

# A Deep Learning-based approach for Mimicking Network Topologies: the Neris Botnet as a Case of Study

Francisco Álvarez-Terribas  
Roberto Magán-Carrión  
Gabriel Maciá-Fernández  
Antonio M. Mora García

**Abstract** The number of connected devices to Internet is growing every year, making almost everything in touch. However, this scenario increase the probability of systems and communications of suffering security attacks since the attack surface increases proportionally. To counteract against security attacks and threats Network Intrusion Detection Systems (NIDSs) are one of the most used security defenses nowadays. They rely on the use of predefined dataset's for their training and evaluation. However, datasets inner characteristics directly affect the robustness, reliability and performance of NIDSs. In this work, we propose the use of a Variational Autoencoder (VAE) to accurately generate network topologies. For that, we consider the IP addresses as a categorical information to generate them. Previous works avoid to use IPs to generate synthetic network samples thus losing relevant contextual information for NIDSs. Results show the feasibility of the proposed system to mimic the Neris Botnet behavior and characterizing its node roles.

---

Francisco Álvarez-Terribas<sup>[0000-0001-8462-0175]</sup>

Network Engineering & Security Group Dpt. of Signal Theory, Communications and Telematics  
University of Granada (Spain), e-mail: franciscoat@correo.ugr.es

Roberto Magán-Carrión<sup>[0000-0002-7744-7308]</sup>

Network Engineering & Security Group Dpt. of Signal Theory, Communications and Telematics  
University of Granada (Spain), e-mail: rmagan@ugr.es

Gabriel Maciá-Fernández<sup>[0000-0001-9256-453X]</sup>

Network Engineering & Security Group Dpt. of Signal Theory, Communications and Telematics  
University of Granada (Spain), e-mail: gmacia@ugr.es

Antonio M. Mora García<sup>[0000-0003-1603-9105]</sup>

Network Engineering & Security Group Dpt. of Signal Theory, Communications and Telematics  
University of Granada (Spain), e-mail: amorag@ugr.es



## 1 Introduction

The Cisco Annual Internet Report (2018–2023) [6] forecast a huge increment of the number of connected devices to Internet. It foresees more than 29 billions of connected devices, more than three times the global population. Moreover, it is expected to be boosted by the real deployment of new communications technologies *e.g.*, 5G, allowing heterogeneous devices from Internet of Things (IoT) ecosystems for an easy and affordable Internet access. This scenario produces huge volumes of heterogeneous network traffic with high rates. Despite of the benefits of such a hyper-connectivity bring to the society, it is also a two-edged sword. From the point of view of the security, this scenario increments the attack surface also increasing the risk of suffering attacks. According to the ENISA Threat Landscape 2020 report (ETL) [9], the sophistication of threat capabilities seriously increased in 2019, having detected over 200,000 daily new variants of malware targeting diverse objectives. Because of the previous reasons, additional security measures are needed to deal with all kinds of security threats both known and unknown (zero-day attacks). For this purpose, traditionally, Intrusion Detection Systems supported by different technologies, techniques and algorithms have been used [16]. IDSs rely on the use of previously gathered datasets for training, validating and testing them. In particular, NIDSs rely on the use of network traffic datasets for different purposes like attack classification or anomaly detection. However, the main drawback of these systems, mainly based on Machine Learning (ML) techniques, is that they require adequate and reliable datasets for their training, in which the existence of notable differences between the distribution of the positive class, attack traffic, and the negative class, background traffic, the latter following a normal behavior. This fact, together with the use of unsuitable datasets most of them outdated, synthetically generated and with not enough duration [16], have a notable impact on the performance and limits the practical application deployment of NIDSs.

One of the main performance issues in classification systems in general and NIDSs in particular is the class imbalance problem seen in network datasets used to fit and validate these kind of systems. Moreover, ML-based models should be robust enough to, first, be able to detect unknown attack samples and, second, to avoid attack targeting them. These issues has been tackled by authors in the research literature for model enhancement [21] mainly focusing on the new traffic samples generation from only continuous features, leaving apart categorical features which are, on one hand, very interesting in the intrusion detection problem and, on the other hand, difficult to correctly generate it. For instance, the synthetically generation of IP addresses needs of considering the network topology and IP prefixes which is not trivial. Here is where the Deep Learning (DL)-based approaches could be useful. In this work we propose a DL-based network attack topology generation through the use of VAE [12] approaches. By means of this technique we are not only able to generate the network topology of the attacks but node roles and their relationships are also accurately generated. In particular, we successfully mimic a hierarchical botnet topology and its node roles and behavior from network samples of the well-known Neris Botnet [13].

The rest of the document is structured as follows. First, the state of the art will be introduced in Section 2 and then the used dataset. The proposed methodology in Section 3 will be described. After that, we introduce and describe the experimental environment of the study in Section 4 followed by the results we have obtained in Section 5. Later we will complete the document with the conclusions reached and future work lines in Section 6.

## 2 Background

The Synthetic Minority Oversampling Technique (SMOTE) algorithm [5], alongside other derived algorithms like [11, 4], have been the most used oversampling methods for enriching imbalanced datasets. Briefly, SMOTE algorithm selects one instance from the minority class and computes which are its nearest neighbors. Then, it interpolates synthetic samples between the selected instance and its neighbors.

As a result of the irruption of DL paradigm during the last years, different authors started using these techniques trying to solve the synthetic data generation problem. For example, Vu *et al.* [20] proposed the application of deep generative adversarial models to the Network Information Management and Security Group (NIMS) dataset [2]. They improve the classification performance over SMOTE derived algorithms. Afterwards, Engelmann *et al.* [8] introduced a more robust generative adversarial model architecture ready to work with categorical features. To achieve that, they applied pre-processing techniques such as the one-hot encoding. This technique creates a representation of each categorical feature with a low value (represented as 0) or a high value (represented as 1). It is worth mentioning that this kind of models, alongside VAEs, are being widely used for synthetic image generation [10].

Another interesting methodology is proposed in [14, 7]. In this work, the dataset's samples are encoded into a latent space, afterwards, SMOTE or other derived algorithm, are applied. This over-sampled latent space is subsequently decoded into the original feature space for the synthetically generated samples.

Graph based approaches have been widely applied in many fields and context. One of them is the computer networks that can be directly represented as graphs, as it is shown in multiple studies for evaluating physical and logical network topologies [18]. Also very specialized topology generation tools, such as Boston university Representative Internet Topology gEerator (BRITE) [17] have been devised and tested. Despite their good performance generating synthetic network topologies these tools are not designed to generate complete network traces. In our work, we propose a methodology that, theoretically, could mimic whichever network topology represented in network datasets to robust ML-based NIDSs.

All the previously mentioned solutions behave appropriately with continuous features but they cannot be applied in problems where high dimensional categorical data is involved. It is, for instance, the IP addresses in network traffic datasets. In this context the use of pre-processing techniques like one-hot encoding is very limited.

We propose in this work the use of VAE approaches due to their capacity to generate synthetic samples in some other context of use *e.g.*, image generation. Because of that, we can obtain synthetic network traffic data for managing categorical data, *i.e.* nodes (IP) and their connections. As a result, new synthetic network topologies will be created from existing network traffic samples following similar distributions but slightly different from the original.

### 3 Network topology generation with deep learning

This section firstly introduces the dataset used in this work. After that, the methodology we followed for the synthetic network traffic data generation with VAEs is also described.

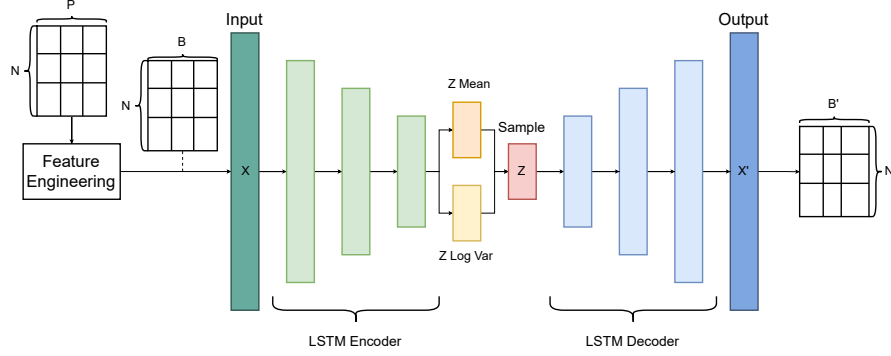
#### 3.1 UGR'16 Dataset: the Neris botnet

The UGR'16 dataset [15] contains real anonymized NetFlow traffic samples captured in a tier-3 ISP for 4 months. This data is divided into two partitions, *calibration set* and *test set*. The *calibration set* contains 3 months of captured normal real network traffic and one month for the *test set*. Additionally, the latter contains synthetically generated attacks. They are *low-rate DoS*, *high-rate DoS*, *Port Scanning* and *Botnet*. Apart from that, anomalous traffic detected by three well-known network IDSs were also labeled as *UDP port scan*, *SSH scan* and *spam campaigns*.

In this work, we will focus on the Botnet attack because it is one of the most complex attack kind found in the UGR'16 dataset. It allows us to test our proposal to mimic really hard to reproduce attacks. The nearly 2 million network samples used for the Botnet attack generation belongs to the behavior and topology of the famous Neris Botnet. Neris has a hierarchical structure where there are a *botmaster*, the command and control (C&C) servers and the bots. A C&C server communicate to bots by Hyper Text Transfer Protocol (HTTP) to command them in some way. In this case, they are commanded for sending spam emails to perform click fraud. Interested readers are kindly referenced to the work for a more detailed explanation.

#### 3.2 Proposed methodology

In Fig. 1 the architecture of the proposed solution can be shown. There two modules are clearly distinguished. The first one defines an initial feature engineering stage. It transforms the original features ( $P$ ) into a new ones ( $B$ ) following a binary encoding approach having a derived dataset of  $X = N \times B$  dimension, with  $B > P$ , as input of the VAE module.



**Fig. 1** Proposed methodology. It is composed by an initial feature engineering process followed by a VAE with LSTM based layers for encoding and decoding.

In regards to the second module, a VAE, modeled with Long Short-Term Memory (LSTM) [3] layers, is also shown. It is in charge of encoding the original samples into a variable latent space. This latent space follows a Gaussian distribution characterized by its mean, the  $ZMean$ , and its standard deviation, the  $ZLogVar$ . Choosing and decoding new values ( $Z$ ) of this distribution, allows us to generate new synthetic samples. By means of the use of LSTM layers, the model is able to generate the underlying network topology seem from the original network samples but taking into account the context of them and their temporal relationships. It allows us to accurately mimic, no matter the network dataset used, how a network nodes interact, the network topology, the node characteristics and roles that characterize the behavior of the different attacks. It is useful for training of NIDSs, making them more robust against unseen attacks (zero-day attacks).

## 4 Experimental design

This section describes the experimental environment, as well as the configuration of the experiments to evaluate their behavior later.

The implementation of the model has been carried out using the Tensorflow library [1]. Moreover, the model has been configured with three LSTM layers in both the encoder and the decoder with a learning rate of  $1e-5$  for a total of 10 epochs. The type and dimension of the involved layers are shown in Table 1. For fitting the model, the 70% of the data has been used leaving the rest of the dataset samples for testing it.

To accurately obtain the temporal relation among samples, batches of 75 observations were chosen. This parameter is likely to be optimized in some way since it is expected to have a significant impact on the performance of the model. It should be studied in detail in a future work.

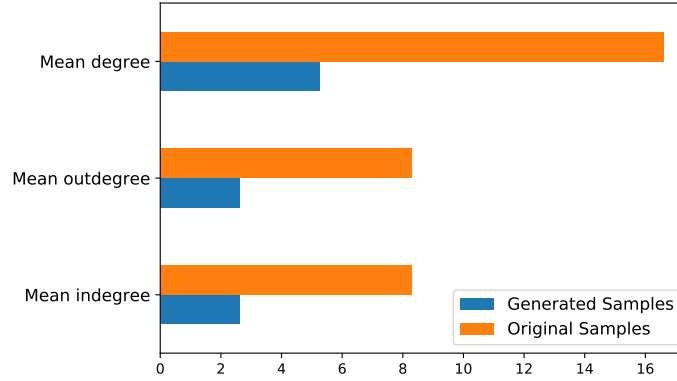
**Table 1** Detail of the autoencoding layers.

Encoder Layers	shape	Decoder Layers	shape
Dense 1	75x214	LSTM 1	75x104
LSTM 1	75x214	LSTM 2	75x164
LSTM 2	75x164	LSTM 3	75x214
LSTM 3	75x104	Dense 1	75x214

All the experiments run in a Linux based operating system, with an Intel Xeon Silver 4208 CPU (2.10GHz and 32 cores) with 32 GB of RAM memory. Three 12GB Nvidia RTX 2080ti GPUs were used to reduce the model training elapsed time.

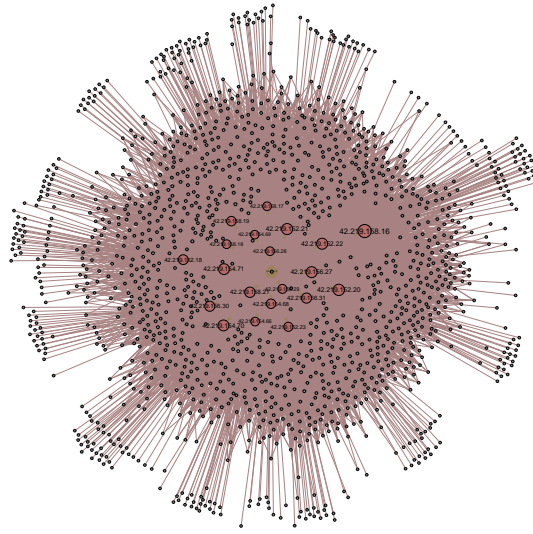
## 5 Results and discussion

To evaluate the performance of the proposed system, a numerical and visual analysis has been carried out from the perspective of graph theory. A comparison between the mean node degrees of original topology and the generated one is shown in Fig. 2. In

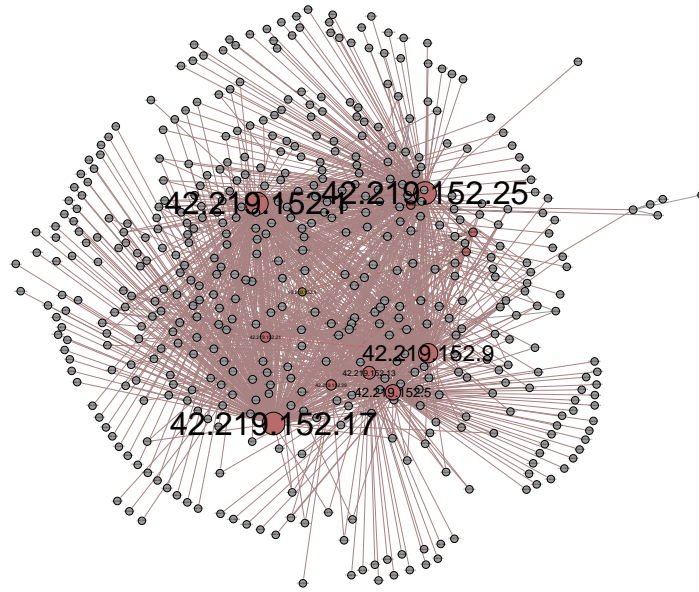
**Fig. 2** Mean degree of network nodes both from the original topology and from the generated one.

comparison with the original topology, the node degree of the generated topology is lower in general. Visually, the original and generated topology are shown in Figs. 3 and 4 respectively. In these figures, the size of the nodes is proportional to their degree. They mainly differ in the number of generated nodes though the node roles are almost all represented according to the ones found in the Neris Botnet network samples.

This fact is observed in detail in Figs. 5 and 6 for the original topology and the generated one, respectively. In these figures, a botmaster node can be found (green



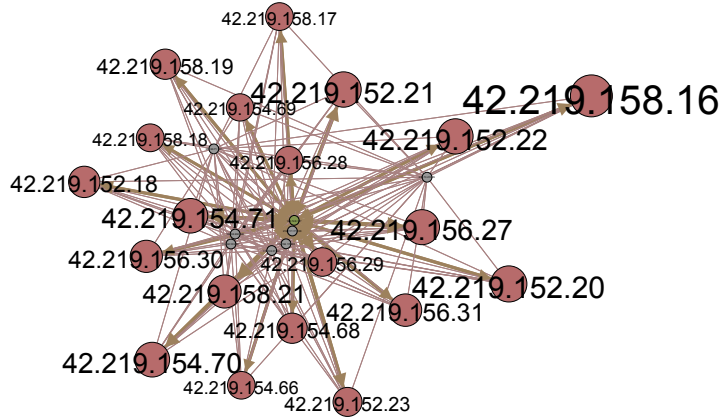
**Fig. 3** An overall view of the original Neris Botnet topology.



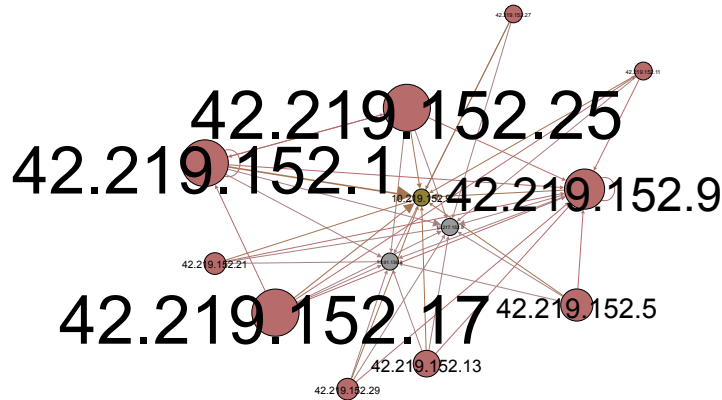
**Fig. 4** An overall view of the generated Botnet topology.

color) which is surrounded by and connected to the C&C servers (red color). Bots are the smaller ones (gray color) due to their low degree.

From the previous results, it can be concluded the reliability of the system in characterizing the three main actors found in the Neris botnet: bots, C&C servers and the botmasters. However, much more work should be done to accurately represent both the network dimension, in terms of the number of different generated nodes, and the connections among them.



**Fig. 5** Detail of the original Neris Network topology after leaving out bots nodes. The botmaster node (green color) is centered and surrounded by the C&C servers (red color). The large is the size of a node, the bigger its degree is.



**Fig. 6** Detail of the generated topology after leaving out bots nodes. The botmaster node (green color) is centered and surrounded by the C&C servers (red color). The large is the size of a node, the bigger its degree is.

## 6 Conclusions and Future work

Nowadays, devices, things and people are sharing information making them prone to be attacked due to the continuously increasing attack surface. This way, intelligent, robust and automatic ML-based solutions must be devised to counteract known and unknown security attacks. However, they rely on the use of predefined datasets which most of them are obsolete, unrealistic or with enough duration to accurately represent the context of use.

In this work we proposed a VAE based solution to accurately mimicking to a great extent the structure and topology of the network samples. In particular, the Neris Botnet behavior has been reproduced in which the involved node roles are well characterized too. In comparison with state-of-the-art solutions, our work smartly managed the IPs, as categorical variables, usually avoided in the literature. Eluding to use categorical information of network datasets reduces the environment contextual data useful for reliable and robust NIDSs.

As future work, further experimentation should be performed to firmly conclude the relevance of using VAE or some other DL-based techniques to accurately mimic network topologies. Moreover, we will work on to generate complete network observations that include all the original variables and not only those that define the network topology such as IP addresses. This way, the application of more sophisticated Deep Learning models would also be interesting, *i.e.* DL with attention relying on the use of transformers [19]. Finally, the impact of the generated synthetic data generation on the performance, robustness and reliability of NIDS systems will be evaluated.

## References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015)
2. Alshammari, R., Zincir-Heywood, A.N.: Can encrypted traffic be identified without port numbers, ip addresses and payload inspection? *Computer networks* **55**(6), 1326–1350 (2011)
3. Bakker, B.: Reinforcement learning with long short-term memory. In: T. Dietterich, S. Becker, Z. Ghahramani (eds.) *Advances in Neural Information Processing Systems*, vol. 14. MIT Press (2001)
4. Barua, S., Islam, M.M., Yao, X., Murase, K.: Mwmote—majority weighted minority oversampling technique for imbalanced data set learning. *IEEE Transactions on Knowledge and Data Engineering* **26**(2), 405–425 (2014). DOI 10.1109/TKDE.2012.232
5. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research* **16**, 321–357 (2002)
6. Cisco: Cisco Annual Internet Report (2018–2023) White Paper. <https://bit.ly/3jpAgNx> (2020). [Online; Accessed englishJuly 4, 2022]



7. Dablain, D., Krawczyk, B., Chawla, N.V.: Deepsmote: Fusing deep learning and smote for imbalanced data. *IEEE Transactions on Neural Networks and Learning Systems* pp. 1–15 (2022). DOI 10.1109/TNNLS.2021.3136503
8. Engelmann, J., Lessmann, S.: Conditional wasserstein gan-based oversampling of tabular data for imbalanced learning. *Expert Systems with Applications* **174**, 114582 (2021). DOI <https://doi.org/10.1016/j.eswa.2021.114582>. URL <https://www.sciencedirect.com/science/article/pii/S0957417421000233>
9. ENISA: ENISA Threat Landscape (2020) White Paper. <https://www.enisa.europa.eu/news/enisa-news/enisa-threat-landscape-2020> (2020). [Online; Accessed english July 4, 2022]
10. Fajardo, V.A., Findlay, D., Jaiswal, C., Yin, X., Houmanfar, R., Xie, H., Liang, J., She, X., Emerson, D.: On oversampling imbalanced data with deep conditional generative models. *Expert Systems with Applications* **169**, 114463 (2021). DOI <https://doi.org/10.1016/j.eswa.2020.114463>
11. He, H., Bai, Y., Garcia, E.A., Li, S.: Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), pp. 1322–1328 (2008). DOI 10.1109/IJCNN.2008.4633969
12. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
13. Lab, S.: CTU-13 Dataset. Capture 42. Neris botnet. (2011)
14. Lim, S.K., Loo, Y., Tran, N.T., Cheung, N.M., Roig, G., Elovici, Y.: Doping: Generative data augmentation for unsupervised anomaly detection with gan. In: 2018 IEEE International Conference on Data Mining (ICDM), pp. 1122–1127 (2018)
15. Maciá-Fernández, G., Camacho, J., Magán-Carrión, R., García-Teodoro, P., Therón, R.: UGR'16: A new dataset for the evaluation of cyclostationarity-based network IDSs. *Comput. Secur.* **73**, 411–424 (2018). DOI 10.1016/j.cose.2017.11.004
16. Magán-Carrión, R., Urda, D., Diaz-Cano, I., Dorronsoro, B.: Towards a reliable comparison and evaluation of network intrusion detection systems based on machine learning approaches. *Applied Sciences* **10**(5) (2020). DOI 10.3390/app10051775
17. Medina, A., Lakhina, A., Matta, I., Byers, J.: Brite: An approach to universal topology generation. In: MASCOTS 2001, Proceedings Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, pp. 346–353. IEEE (2001)
18. Sterbenz, J.P., Çetinkaya, E.K., Hameed, M.A., Jabbar, A., Qian, S., Rohrer, J.P.: Evaluation of network resilience, survivability, and disruption tolerance: analysis, topology generation, simulation, and experimentation. *Telecommunication systems* **52**(2), 705–736 (2013)
19. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: *Advances in Neural Information Processing Systems*, vol. 30 (2017)
20. Vu, L., Bui, C.T., Nguyen, Q.U.: A deep learning based method for handling imbalanced problem in network traffic classification. In: *Proceedings of the Eighth International Symposium on Information and Communication Technology, SoICT 2017*, p. 333–339. Association for Computing Machinery, New York, NY, USA (2017). DOI 10.1145/3155133.3155175. URL <https://doi.org/10.1145/3155133.3155175>
21. Xiong, P., Buffett, S., Iqbal, S., Lamontagne, P., Mamun, M., Molyneaux, H.: Towards a robust and trustworthy machine learning system development: An engineering perspective. *Journal of Information Security and Applications* **65**, 103121 (2022). DOI 10.1016/j.jisa.2022.103121

# Evaluating Classifiers' Performance to Detect Attacks in Website Traffic

Daniel Urda<sup>1,\*</sup>, Nuño Basurto<sup>1</sup>, Meelis Kull<sup>2</sup>, and Álvaro Herrero<sup>1</sup>

**Abstract** Websites are an appealing target for attackers due to the large number of users that make use of them, resulting in a massive exchange of information. Different types of vulnerabilities and anomalies can be present in this context, such as SQL Injection or DDoS attacks. To cope with all this, different types of Computational Intelligence-based techniques, capable of detecting these undesired events, are used. To this end, feature selection methods allow a dimensionality reduction that subsequently helps classification algorithms to achieve high performance results. In the present paper, both feature selection methods and classifiers are evaluated on an open dataset called CSIC2010 v2. Furthermore, a deep study of the features within this dataset has been carried out with the aim of extracting as much information as possible, generating two alternative datasets with this information that are employed for comparison purposes in this research. The interesting results that have been obtained contributes to an improvement on the detection of web attacks.

**Keywords:** intrusion detection, cybersecurity, machine learning, supervised learning, feature selection

---

\* Corresponding author: [durda@ubu.es](mailto:durda@ubu.es)

1. Daniel Urda, Nuño Basurto, Álvaro Herrero  
Grupo de Inteligencia Computacional Aplicada (GICAP), Departamento de Ingeniería Informática,  
Escuela Politécnica Superior, Universidad de Burgos, Av. Cantabria s/n, 09006, Burgos, Spain.  
e-mail: [nbasurto@ubu.es](mailto:nbasurto@ubu.es), [ahcosio@ubu.es](mailto:ahcosio@ubu.es)

2. Meelis Kull  
Department of Computer Science, University of Tartu, Estonia. e-mail: [meelis.kull@ut.ee](mailto:meelis.kull@ut.ee)

## 1 Introduction and previous work

Nowadays, the Internet is a daily-used tool for personal, institutional or governmental purposes, among many others. The classic scenario of having internal and exclusive networks for trusted users and/or collaborators is not more a reality. Instead, the use of Internet implies the accessibility to tons of information and data to millions of people, most of them anonymous, around the world. In this sense, many threats and unknown vulnerability issues, like SQL Injection [16] or DDoS attacks [15], could be exploited by malicious actors who may want to damage any digital system or benefit from the access they may achieve to them [5]. As occur in many other areas such as Healthcare [6], Bioinformatics [19] or Industry [20], Artificial Intelligence (AI), and particularly Machine Learning (ML), arises as a powerful tool to address difficult and unsolved problems in cybersecurity based on traffic and/or access data generated by communications throughout time [13, 14], such as intrusion detection [9].

More precisely, an accurate detection of anomalies and vulnerabilities in website traffic could potentially have a positive impact in terms of the network performance, the quality of the service provided and the users' experience with the corresponding website application. To this end, there exists some well-known datasets, such as CSIC2010 [7] (that some of the authors has previously work with) or CSE-CIC-IDS2018 [17], that have been employed to develop AI/ML-based solutions in order to detect a given outcome of interest. Particularly, the former dataset consists of thousands of web requests automatically generated at the "Information Security Institute" of the Spanish Research National Council for an e-commerce website application. On the one hand, unsupervised learning techniques were applied to this dataset to provide an overview of HTTP traffic and to identify anomalous situations, concluding the difficulties found to clearly differentiate among both categories of instances with the models tested [1]. On the other hand, supervised learning techniques have also been used to discriminate among normal and anomalous traffic based on this dataset. For instance, in [8] several well-known ML models were trained over features obtained from a unsupervised language representation model for embedding HTTP requests, and, more recently, a deep learning-based approach was presented in [2] showing how these kind of models can effectively predict the traffic of visiting websites. Furthermore, an ensemble of three deep learning models trained to detect web attacks separately is presented in [12] achieving a lower false positive and negative rates, and another deep learning-based approach employed a Convolutional Neural Network (CNN) to automatically extract semantic features that are used to train a Support Vector Machine classifier, achieving state-of-the-art performance [21].

Although these previous works have shown the benefits of using AI/ML-based approaches to address the problem of accurately predicting normal or anomalous traffic on the CSIC2010 dataset, the authors of this work consider that some its original features, such as the full URL request or the payload, contain rich information to improve the discrimination performed by ML models and have not been fully and properly exploited before, at the same time that existing work do not provide enough detail of how they processed these kind of features to feed and train ML classifiers. In this sense, authors aim at further preprocessing the original features of this dataset

to extract new and relevant ones that may increase the predicting performance of ML models. Besides, and due to the increase on the number of input variables, there might be a need of using feature selection techniques to select only those relevant features, among the whole set of variables, which have direct impact on the target outcome, a step which is commonly performed when developing ML-based solutions where samples are at least described by several dozens or a few dozens of variables and that, to the best of authors knowledge, has not been previously considered over this dataset.

Therefore, this paper aims at evaluating the performance of ML-based solutions by proposing to use new and richer CSIC2010 dataset instances. For this purpose, two new instances,  $D_1$  and  $D_2$ , are generated as a result of further preprocessing the payload feature within the original dataset ( $D_1$ ), and this one together with the full URL request field ( $D_2$ ). Both datasets are used to develop three well-known classifiers (a linear model like the Least Absolute Shrinkage and Selection Operator, LASSO, as baseline, and two more sophisticated ones like  $k$ -Nearest Neighbours and Support Vector Machines) in order to accurately predict normal and anomalous traffic. Furthermore, this work evaluates the impact of using two well-known feature selection techniques (information gain as a filtering method and the LASSO as an embedded one) to choose a subset of relevant features prior to building the above-mentioned classifiers.

The rest of the paper is organized as follows. Section 2 presents the ML methods used as classifiers, as well as the feature selection methods employed in the comparison. Next, the CSIC2010 dataset and the feature extraction step performed to generate two new and richer instances are described in detail in Section 3. Then, the results of the evaluation carried out in this work are presented and analyzed in Section 4. Finally, Section 5 contains some conclusions and future work.

## 2 Applied methods

This work employs three well-known and standard ML methods as classifiers, and two well-known feature selection techniques. Concerning the ML methods used, authors chose a simple linear classifier, such as LASSO, to be the baseline model in the study. Secondly, two other methods which allow capturing non-linear relationships in the data were chosen with the goal of outperforming the baseline model: one distanced-based method like  $k$ -Nearest Neighbour, and one kernel-based one like Support Vector Machines. Overall, these three chosen methods approach the given problem from different perspectives, thus providing a good evaluation framework from the ML point of view. Similarly, two standard feature selection algorithms, such as Information Gain and LASSO, were chosen to retain relevant features and train the above-mentioned classifiers, since these two algorithms have been recently employed in related problems [10, 13].

## 2.1 Information Gain

The Information Gain (IG) algorithm [11] aims at entropy reduction to carry out the transformation of the dataset. Within this study, it is used as a feature selection method, establishing the gain of each variable  $X_j$  with respect to the class label  $y$ , where the  $H$  is referred to the entropy value, this is depicted in Equation 1:

$$IG(y, X_j) = H(y) + H(X_j) - H(y, X_j) \quad (1)$$

## 2.2 The LASSO

It is a simple and widely used classifier which models the dependent variable as a linear combination of the independent variables [3]. For a binary classification problem, Equation 2 depicts how the binary outcome (i.e.,  $y_i \in [0, 1]$  and  $i = 1..N$  samples) is modelled as a function of the input variables and the parameters' vector  $\beta$ :

$$y_i = f(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}) \quad , \quad (2)$$

where  $f(\cdot)$  is the logistic or sigmoid function. The parameters' vector  $\beta$  is learned by solving a minimization problem very similar to the one solved for logistic regression, although LASSO adds a  $L_1$ -regularization term to the error term in order to control overfitting issues by pushing as many  $\beta$  coefficients as possible to zero if these are not relevant to predict the target outcome. Equation 3 shows the minimization problem solved by LASSO:

$$\hat{\beta}_\lambda = \arg \min_{\beta} \|y - f(\beta X^T)\|_2^2 + \lambda \|\beta\|_1 \quad , \quad (3)$$

where  $\lambda$  is a hyper-parameter of the model that controls the strength of the regularization (i.e., the higher the value of  $\lambda$ , the more  $\beta$  coefficients will be pushed to zero). Therefore, LASSO can also be employed as a feature selection method which retains relevant features associated to the target outcome. Once a LASSO model is fitted to data, the subset of selected features would be those for which their corresponding  $\beta$  coefficient is distinct from zero (otherwise, it will imply that the feature does not contribute to the target outcome).

## 2.3 Support Vector Machines

The Support Vector Machines (SVM) [4] is one of the most widely used classifiers, it is based on Statistical Learning Theory. This machine learning model aims to identify a hyperplane that is able to maximize the margins of separation of the

different classes that have the data (in this research 2), in the training set. In this way, universalizing the archetype can distinguish between classes in the new instances (test set). One of the characteristics of this model is its great sensitivity to the modification of the kernel used, its corresponding hyper-parameters and cost. In this study, a well-known kernel function, such as the Radial Basis Function (RBF) which has been shown to perform well in a wide variety of problems and that only has one single hyper-parameter  $\gamma$ , has been used. This kernel function is defined as follow in Equation 4:

$$K(X, X') = \exp(-\gamma \|X - X'\|^2) \quad (4)$$

## 2.4 k-Nearest Neighbour

The k-Nearest Neighbour (kNN) algorithm is capable to detect intrusion attacks establishing the distance between the different instances. This is accomplished by generating the local density of a test element  $X_i$ , by implementing a hyper-sphere containing its  $k$ -th nearest neighbors, establishing that number. Later, if this calculated density is lower than an established limit, the anomaly is detected; on the other hand, if it is high, is determined that it belongs to the target set [18].

## 3 Dataset on Web Attacks

This dataset was designed for the simulation of attacks produced in HTTP queries. It was developed by the Spanish Research National Council (CSIC) in 2010 and simulates the access of users to an e-Commerce web application, where purchases are made through the use of a shopping cart and providing different personal data. The set comprises a size of 223,585 samples and 18 features, where each sample is labeled either as normal or anomalous traffic.

A lot of preprocessing work has been done to this original dataset, where samples belonging to the same session (i.e., those that have the same session ID in the *cookie* variable) are grouped together in order to obtain the full payload for that give HTTP request, thus finally resulting in  $N = 13,569$  samples. In addition, an in-depth study has been carried out in the following variables with aim of extracting new possibly relevant features:

- **method:** This variable originally contains three types of methods: "PUT", "POST" and "GET". They have been discretized into three new variables.
- **payload:** It contains the useful information of the data in the form of several *Key=Value* pairs, where the header and the metadata are excluded. With the aim of realizing a large study, the keys within this payload have been extracted taking as reference the word prior to the "=" sign, resulting in a total of 19 keys (e.g., "ID", "password" or "DNI", among others). With this information, 19 binary

variables were generated to indicate whether that session includes the key in its payload or not. On the other hand, the value assigned to a given key (everything after the “=” symbol) was processed and transformed to its length. This way, 38 new variables are generated, in total, using the keys and their corresponding values identified within the payload. Additionally, two more variables were added: “total.length” which sums up the size of all keys’ values within the session, and “num.keys” that sums the different keys present in the session. An example of the original payload variable in the same session is next shown:

modoA=insertar      precio=1764      B1=Pasar+por+caja

where in our preprocessing step, 3 out of the 19 possible keys would be identified (i.e., *modoA*, *precio* and *B1*) and, consequently, their associated binary features created by this step will be set to one. Besides, their corresponding variables measuring the length of their value will be set to 8, 4 and 14, respectively. The remaining 32 variables are, therefore, set to 0.

- **url:** The processing of this variable has been done in a similar way to the payload. Firstly, the path of the *URL* was analyzed and validated if it accomplished that this path always started with “http://localhost:8080/”. Then, the path is subsequently divided by their different directories, using the symbol “/” as the split character. With the objective of extracting the file extension of the resource accessed in the given path (24 extensions like “.jsp”, “.gif” or “.jpg” were identified across the dataset), each one of those extensions were included as new binary variables. Furthermore, 4 more variables are generated, “isValidURLPath”, which is extracted in the first step of the study, and when it takes a TRUE value, the rest of the above-mentioned variables associated to the *URL* of this session will be set to 0. The other variables are “numDir”, which counts the number of the directories accessed in the path (it is a number between 0 and 4), “lengthDir” which sums the total length of the several directories identified in the path, and “lengthFich” which is the sum of the filenames accessed. An example of the original *URL* variable is:

http://localhost:8080/tienda1/publico/autenticar.jsp

Table 1 shows a summary of the number of samples and variables present in the original and the two new generated datasets instances ( $D_1$  and  $D_2$ ). Particularly,  $D_1$  only includes features extracted from the *payload* variable, while  $D_2$  adds on top the ones extracted from the *url* variable with the goal of analyzing the impact of the extra set of features provided with  $D_2$  in the classifiers’ performance. Besides, the class distribution within each dataset instance is also shown. The higher presence of anomaly samples can be explained due to the higher chance of finding anomalies in smaller and more isolated sessions in contrast to normal traffic.

Table 1: Summary of the CSIC dataset version.

ID	Samples	Features	Normal Class	Anomaly Class
<b>Original</b>	223585	18	104000	119585
<b><math>D_1</math></b>	13569	50	4303	9266
<b><math>D_2</math></b>	13569	78	4303	9266

## 4 Results

The methodology carried out deals with the comparison of the different classification methods, using feature selection techniques on both datasets. For the validation of the results, all methods described in Section 2 have been executed using a stratified 10-fold cross-validation strategy, which performs a total of 10 executions. This strategy divides the dataset in 10 partitions of equal size and keeping the original class distribution within each partition. Then, 9 partitions (known as train set) are used to fit the models and the remaining one (known as test set) to measure their performance in new unseen data, changing iteratively the train and test sets on each of the 10 executions.

Different widely-used metrics have been calculated to measure the performance of each model trained: *Precision*, *Recall*,  *$F_1$  Score*, *g-mean*, *Area Under the ROC Curve (AUC)* and *Accuracy*. Due to the space limitation of the paper and the characteristics of the data, authors decided to show only the *Accuracy* and *AUC* figures in this section.

Concerning feature selection methods, a specific criteria was established to determine which features are selected. In the case of LASSO, those features with their corresponding  $\beta_j$  coefficient greater than 0.0001, in absolute values, were selected (i.e., those with a zero or extremely low value are discarded). On the other hand, the criterion used for IG is that, given a set of gain values for each variable computed by the method, only those with a value higher than the arithmetic mean will be retained.

Given the great sensitivity of SVMs, where a small modification of their parameters can lead to large changes in the performance metrics, a study was carried out in order to empirically choose a value for its *gamma* hyper-parameter, resulting in a pseudo-optimal value of 0.015. Therefore, only the cost hyper-parameter was varied to train different SVMs and test their performance in new unseen data.

Following the above, first the results returned by the classifiers on both datasets are analyzed without performing any feature selection. In this sense, Table 2 shows the values reached for *Accuracy* and *AUC*. For the first metric, the best result is achieved by a SVM with a cost of 3000 and over the  $D_2$  dataset (**0.736**). On the other hand, using the smaller version of the dataset (feature-wise) does present similar performance values, although the other two classifiers (LASSO and kNN) slightly improves by using  $D_1$ . In the case of *AUC*, there are little differences between both



Table 2: 10-fold cross-validation average performance for the different classifiers analyzed and with no feature selection performed.

	$D_1$ (50 features)		$D_2$ (78 features)	
Classifier	Accuracy	AUC	Accuracy	AUC
LASSO	0.719	0.796	0.718	0.797
kNN (k=10)	0.725	0.806	0.718	0.798
SVM (cost=3000)	0.734	0.866	0.736	0.866

datasets, where kNN works better for  $D_1$  and in both datasets the SVM achieves the highest value (**0.866**).

Additionally, a new comparison that includes the use of feature selection techniques is presented in Table 3. To this end, a new column has been added indicating, for each given method, the average number of selected features across the iterations of the evaluation strategy employed.

Table 3: 10-fold cross-validation average performance for the different classifiers and the two feature selection methods analyzed.

		$D_1$ (50 features)			$D_2$ (78 features)		
FS method	Classifier	# Features	Accuracy	AUC	# Features	Accuracy	AUC
IG	LASSO	27.2	0.715	0.789	29	0.715	0.789
	kNN (k=10)		0.723	0.801		0.716	0.793
	SVM (cost=3000)		0.704	0.828		0.704	0.828
LASSO	LASSO	21.8	0.719	0.796	35.1	0.716	0.796
	kNN (k=10)		0.724	0.803		0.719	0.797
	SVM (cost=3000)		0.700	0.829		0.700	0.829

A great fit is observed in the features of both datasets, where IG leaves a close number in both cases (27.2 and 29), while LASSO retains a lower number of features in the case of  $D_1$  (21.8), but a higher number than IG in  $D_2$  (35.1). This difference in features results in a significant improvement in execution time.

In order to continue with a homogeneity of results in this table, the same parameters have been selected for the classifiers as in the previously mentioned table. It can be seen that LASSO and SVM get similar results in both datasets, although kNN achieves a greater improvement in terms of Accuracy and AUC when  $D_1$  is used instead of  $D_2$ . In general and regarding the results obtained, kNN could be said that

is the best classifiers in terms of Accuracy, while SVM is the one if we focus in the AUC metric.

Finally, the results analysed from the different perspectives to be taken into account in this research are shown in radar plots (Figure 1), where all the calculated metrics can be found.

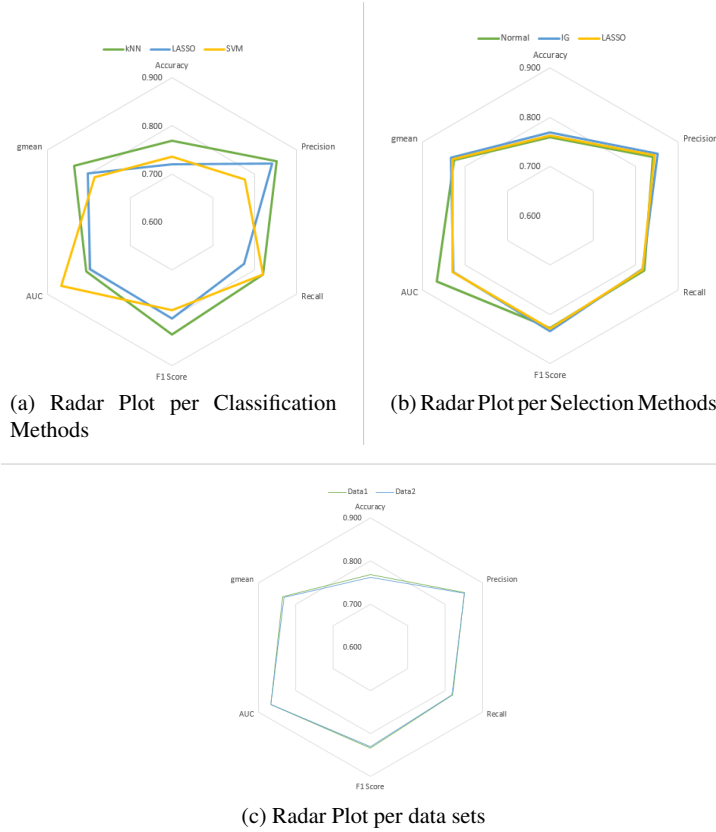


Fig. 1: Radar plot diagrams showing all the calculated performance metrics for the different approaches.

Figure 1a shows the results per classifier, in which kNN stands out over the others, achieving higher values in every metric except AUC, where SVM stands out. Regarding LASSO, although it achieves better values than SVM in some cases, in none of them is it better than kNN. Secondly, in the representation of features selection methods shown in Figure 1b, *Normal* corresponds to the classifiers trained without using feature selection techniques. There are small differences between IG and LASSO, and both of them improve the Normal runs in all metrics except for the AUC, in which the Normal runs clearly stand out. The general trend between

the two selection techniques shows that IG performs slightly better than the others. Finally, in the comparison between the two datasets shown in Figure 1c, there is a very strong similarity, although it can be seen that the use of  $D_1$  allows all methods to, in general, achieve a slight improvement in most metrics rather than using  $D_2$ .

## 5 Conclusions and Future Work

This paper presents an evaluation of different strategies to accurately detect vulnerabilities or anomalies in website traffic based on new dataset versions generated from the well-known CSIC2010 v2 dataset. Concerning the generation of the two new dataset versions considered in the analysis, a thorough work has been carried out for pre-processing the original version of the dataset, carefully analyzing and understanding its features. More specifically, great attention has been paid to the *payload* and *URL* features in the original dataset, which contain rich information. Thus, the two new datasets have been used for the evaluation purpose in this work, which also takes into account the use of several feature selection techniques (IG and LASSO) and classification models (LASSO, kNN and SVM).

In general terms, it can be concluded that feature selection techniques have allowed to reduce the dimensionality of the problem, which leads to a less complex problem to be solved by ML techniques with no significant loss in terms of performance of the classifiers. Among the classification methods, the values achieved by SVM in terms of AUC stand out, making clear the capability of this algorithm to distinguish between anomaly and normal website traffic. On the other hand, kNN has shown a performance improvement with a lower dimensionality (i.e., by using  $D_1$ ), standing out over LASSO in all executions.

These results suggest the use of other more advanced feature selection techniques which may potentially help to improve classification performance as future work. Likewise, the use of new and more advanced classification techniques, such as neural network-based models or ensembles, may be interesting in the development of new research in order to help achieving better performance rates.

## References

1. Atienza, D., Herrero, Á., Corchado, E.: Neural analysis of http traffic for web attack detection. In: Á. Herrero, B. Baruque, J. Sedano, H. Quintián, E. Corchado (eds.) International Joint Conference, pp. 201–212. Springer International Publishing, Cham (2015)
2. Bao, R., Zhang, K., Huang, J., Li, Y., Liu, W., Wang, L.: Research on website traffic prediction method based on deep learning. In: D. Jiang, H. Song (eds.) Simulation Tools and Techniques, pp. 432–440. Springer International Publishing, Cham (2022)
3. Bishop, C.: Pattern Recognition and Machine Learning. Springer, New York, Inc, Information Science and Statistics, Berlin (2006)
4. Cortes, C., Vapnik, V.: Support-vector networks. *Machine learning* **20**(3), 273–297 (1995). DOI 10.1007/BF00994018

5. ENISA: ENISA Threat Landscape Report 2020. [Online; Accessed 9-June-2020] <https://bit.ly/3gdsB10>
6. Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S., Dean, J.: A guide to deep learning in healthcare. *Nature Medicine* **25**, 24 – 29 (2019)
7. Giménez, C.T., Villegas, A.P., Álvarez Marañón, G.: HTTP DATASET CSIC 2010. [Online; Accessed 2-June-2022] <https://www.isi.csic.es/dataset/>
8. Gniewkowski, M., Maciejewski, H., Surmacz, T.R., Walentynowicz, W.: Http2vec: Embedding of HTTP requests for detection of anomalous traffic. *CoRR* **abs/2108.01763** (2021). URL <https://arxiv.org/abs/2108.01763>
9. Go, G.M., Bu, S.J., Cho, S.B.: Insider attack detection in database with deep metric neural network with Monte Carlo sampling. *Logic Journal of the IGPL* (2022). DOI 10.1093/jigpal/jzac007. URL <https://doi.org/10.1093/jigpal/jzac007>. Jzac007
10. Hassani, H., Hallaji, E., Razavi-Far, R., Saif, M.: Unsupervised concrete feature selection based on mutual information for diagnosing faults and cyber-attacks in power systems. *Engineering Applications of Artificial Intelligence* **100**, 104150 (2021). DOI <https://doi.org/10.1016/j.engappai.2020.104150>. URL <https://www.sciencedirect.com/science/article/pii/S0952197620303870>
11. Kent, J.T.: Information gain and a general measure of correlation. *Biometrika* **70**(1), 163–173 (1983). DOI 10.1093/BIOMET/70.1.163. URL <https://academic.oup.com/biomet/article/70/1/163/240380>
12. Luo, C., Tan, Z., Min, G., Gan, J., Shi, W., Tian, Z.: A novel web attack detection system for internet of things via ensemble classification. *IEEE Transactions on Industrial Informatics* **17**(8), 5810–5818 (2021). DOI 10.1109/TII.2020.3038761
13. Magan-Carrion, R., Urda, D., Diaz-Cano, I., Dorronsoro, B.: Improving the reliability of network intrusion detection systems through dataset aggregation. *IEEE Transactions on Emerging Topics in Computing* pp. 1–1 (2022). DOI 10.1109/TETC.2022.3178283
14. Magán-Carrión, R., Urda, D., Diaz-Cano, I., Dorronsoro, B.: Towards a reliable comparison and evaluation of network intrusion detection systems based on machine learning approaches. *Applied Sciences* **10**(5) (2020). DOI 10.3390/app10051775
15. Mirkovic, J., Reiher, P.: A taxonomy of ddos attack and ddos defense mechanisms. *SIGCOMM Comput. Commun. Rev.* **34**(2), 39–53 (2004). DOI 10.1145/997150.997156
16. Pinzón, C., Herrero, Á., De Paz, J.F., Corchado, E., Bajo, J.: Cbrid4sql: A cbr intrusion detector for sql injection attacks pp. 510–519 (2010)
17. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp* **1**, 108–116 (2018)
18. Sukchotrat, T.: Data mining-driven approaches for process monitoring and diagnosis. Ph.D. thesis, University of Texas (2008). URL <https://www.proquest.com/dissertations-theses/data-mining-driven-approaches-process-monitoring/docview/276043339/se-2?accountid=14529>
19. Urda, D., Aragón, F., Bautista, R., Franco, L., Veredas, F.J., Claros, M.G., Jerez, J.M.: BLASSO: integration of biological knowledge into a regularized linear model. *BMC Systems Biology* **12**(5), 361–372 (2018)
20. Wuest, T., Weimer, D., Irgens, C., Thoben, K.D.: Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research* **4**(1), 23–45 (2016). DOI 10.1080/21693277.2016.1192517
21. Yu, L., Chen, L., Dong, J., Li, M., Liu, L., Zhao, B., Zhang, C.: Detecting malicious web requests using an enhanced textcnn. In: 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), pp. 768–777 (2020). DOI 10.1109/COMPSAC48688.2020.0-167

# **ICEUTE Conference**

# Evaluation of an interactive guide for robotics self-learning

Álvaro Ovejero, Jesús Enrique Sierra-García, Matilde Santos

**Abstract** With the arrival of Industry 4.0, the use of robots in the workplace has increased and, therefore, the demand for specialized personnel, capable of integrating robotic applications. When a new robot arrives at a company, it should be running at full capacity in a short time. However, many companies lack staff with the necessary training and skills to do so, and new employees require a quick and efficient learning process. However, it is not easy to adapt training courses to today's changing flexible production processes. Self-directed training can be a useful alternative in these situations, as long as it is carried out with well-prepared teaching materials, organized content and self-assessing tasks. Therefore, in this work, an interactive guide for *Robotics* self-learning is proposed and evaluated. This training action helps ensure that training procedures can be adjusted and synchronized with the demands of the rapidly evolving manufacturing processes. This interactive and autonomous learning makes easier entering the labor force.

**Keywords** self-learning, interactive guide, robotics.

## 1 Introduction

The arrival of Industry 4.0 has increased the incorporation of robots in industries, and therefore, the demand for people capable of implementing robotic applications [1-2]. In fact, when a new robot arrives at a company, it needs to be set up and be ready to work with maximum efficiency in a very short time. However, many companies do not have the staff with the necessary specialized knowledge, and thus a learning process is necessary to train them for this task.

There are at least two alternatives to do it. On the one hand, you can use the manual and try to figure out by yourself how the robot works. This way is often too long and tedious because the manuals contain information that is not always well structured, is

---

A. Ovejero

Electromechanical Engineering Department, University of Burgos, Burgos, Spain

J. E. Sierra-García

Electromechanical Engineering Department, University of Burgos, Burgos, Spain

e-mail: jesierra@ubu.es

M. Santos

Institute of Knowledge Technology, Complutense University of Madrid, Madrid, Spain

e-mail: msantos@ucm.es

too general, and it may be difficult to find the relevant information you are looking for quickly. On the other hand, it is possible to attend a training course on the robot. This option is usually easier but much more expensive, takes time, needs to be scheduled, and a trainer may not be always available. Another drawback is that normally the worker needs to leave the company for a while during the training course and production can be affected.

In these cases, when the students already have a certain professional maturity and perhaps work experience, self-directed training can be a good option if there are adequately prepared teaching tools [3-4], with structured content and self-assessing exercises that allow knowing if the skills and competences have been acquired correctly. This allows the training to adapt and synchronize with the demands of the rapidly changing production processes and makes easier entering the labor force.

Some previous works have developed tools to facilitate the self-learning methods in the engineering fields. To mention a few examples, Santos et al. designed a computer learning assisted tool called ISETL (Integrated System for Electronics Technology Learning) to facilitate the Electronics fundamentals understanding [5]. Augmented reality is introduced in classroom for teaching practice of electrical engineering to improve the student's autonomy and self-ability [6]. Diaz et al. present an interactive tool to obtain the root locus of a system as part of the control engineering training [7]. In [8] a simulator of a submarine connected to the cloud is used as a learning tool.

In this work we develop a proposal to address this type of practical learning. The creation of an interactive guide for rapid self-learning of robotics applications is presented. The guide is structured by functionalities instead of by components, in order to speed up the information search and the training process. To do so, we have considered the most common operations with a robotic arm. Each section of the guide provides a step-by-step procedure, a video example, and a self-assessment exercise. The guide has been experimentally evaluated, showing that it allows users with no previous experience with robots to acquire solid training in a short time. This approach could be applied to other complex engineering systems such as wind turbines [9] or mobile robots [10-11].

The rest of the paper is structured as follows. The interactive guide is described in section 2. Section 3 details the experiments carried out to evaluate the usefulness of the guide. The paper ends with the conclusions and future work lines.

## **2 Interactive guide for robotics self-learning**

Before developing the interactive guide about how to use a robot arm, a comparative study was carried out of the manuals from three big robot manufacturers: ABB [12], Fanuc [13] and Yaskawa [14]. In this study it was observed that there were great similarities in the handling of robotics application although with different robots.

- The robot programming of each company is carried out through a very similar console in all cases, with the same buttons for the movement of the robot arm and its operation. As a sample, all of them include, among others, the dead man's button.

- The definition of the movements is done in the same way, with all three having the same types of movements, MOVJ, MOVL and MOVC.
- The three manufacturers allow programming in different coordinate systems, thus facilitating the work of the programmer. These coordinates are also the same for the three brands, axis-to-axis coordinates, cartesian coordinates and tool coordinates.
- The creation of tool control points (TCP) is quite similar for all of them.
- All of them allow the input of variables in almost the same way.

The hypothesis here is that we can unify the learning of the operation of almost any robotic arm. This could speed up the process of starting up robots in industries. Thus, this work presents an interactive guide to train industrial programmers to work with robot-arms with six degrees of freedom.

The main goal is that a new user with no robot experience should be able to learn to operate a robot quickly, autonomously, and interactively using only the guide and the robot. Therefore, the guide should include videos and self-assessment exercises to facilitate the self-learning, it should be interactive, and it must present the information in a structured way to accelerate the learning process. In addition, the guide should be applicable in different devices such as computers, tablets, mobiles, regardless the operative system.

The objective of this guide is not to replace the manual of the robot but rather it is a complement to this manual. With the guide we describe the basic functions of the robot and how to work with them. The guide provides autonomous, interactive, entertaining, and dynamic learning, without the need to attend training courses.

As the guide must work for any kind of device with any operative system, it was developed as an interactive document in PDF format with hyperlinks to videos specifically prepared for the learning. The videos are a fundamental pillar since they allow a much faster learning experience by showing step-by-step the involved processes. The guide is divided into sections by functionalities. Each section considers an essential function of the robotic arm and is fully developed in only one page.

All sections share the same layout: the step-by-step procedure is located on the left, a video example is presented on the middle of the page, and a self-assessment exercise on the right. Fig. 1 shows the layout of one of the sections.



YASKAWA

## Creating a programme with Mov L

UNIVERSIDAD  
DE BURGOS

5


<p><b>Steps:</b></p> <p>1° Create a new programme: "JOB" → "CREATE NEW JOB"</p> <p>2° Move the robot manually to the first position. Intro that position. "INSERT" → "ENTER"</p> <p>3° Enter all robot positions as in step 2.</p> <p>4° Choose the type of movement (MovL) and the speed (mm/s) of movement.</p> <p>5° Run the programme.</p>	<p><b>Example:</b></p>  <p><b>Notes:</b></p> <ul style="list-style-type: none"> <li>- Know the movement of the robot manually.</li> <li>- Take into account the limits of movement of the robot.</li> <li>- Know the cyclic movement and step by step.</li> </ul>	<p><b>Self appraisal:</b></p> <p>Make a program that goes through 4 different points at a constant speed. Perform the same program with various speeds and finish at the midpoint.</p>
<p>X Step-by-step procedure</p>	<p>X Video example</p>	<p>X Self-assessment exercise</p>

Fig. 1. Example of layout of the sections

In this figure it is possible to observe how each section contains a designed exercise to put into practice that specific functionality. This exercise complements the explanation and allows the students to check if they have acquired the required knowledge. The solution of each exercise is further explained with a video to facilitate the self-learning.

### 2.1 Didactic objectives

The guide was defined to achieve a list of didactic objectives. Some examples are given below although each teacher can propose a different approach.

- To know the functions of the console or Teach Pendant
- To know how to work with a robotic arm in manual mode and in automatic mode
- To know how to move between different points at different speeds, using different types of coordinate systems.
- To know how to define points by using the console and using them to create a program for the correct use of the robot.
- To include variables in the programs, thus being able to design programs to perform different applications

## 2.2 Sections

Most manuals have the information related to the robot arm structured by components or elements, instead of by functionalities. This makes it more complicated and slows down the search of information and the self-learning, especially for people without previous experience. Accordingly, this guide has been split into sections by considering the most common functionalities.

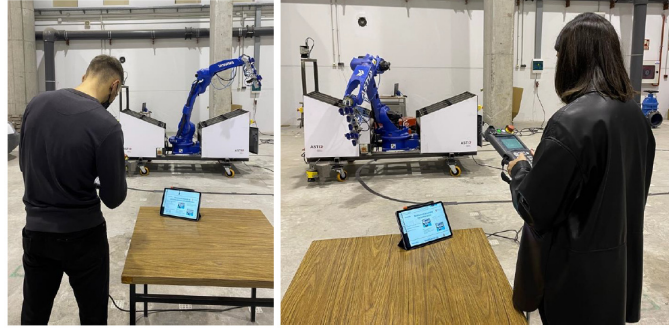
- S1. To move the robot in manual mode by using different types of coordinates
- S2. To move the robot in manual mode at different velocities
- S3. To define a TCP
- S4. To execute a program
- S5. To use the command MovJ
- S6. To use the command MovL
- S7. To use the command MovC
- S8. To define a program with different movements
- S9. To execute a program cyclically
- S10. To modify a program
- S11. To adjust a target position in a program
- S12. To define a user coordinate system

## 3 Results

To assess the feasibility and usefulness of the self-learning guide, a pilot study was carried out with 20 people. The target population was divided into three age groups: 18-35 years old, 35-45 years old, and 45-60 years old, and two different categories: with and without knowledge on robotics. In the study a GP25 robotic arm of Yaskawa was used [14]. Fig. 2 shows a couple of pictures that were taken during the experiments. In the background, it is possible to see the industrial robot arm mounted on a trailer with wheels to facilitate its transport.

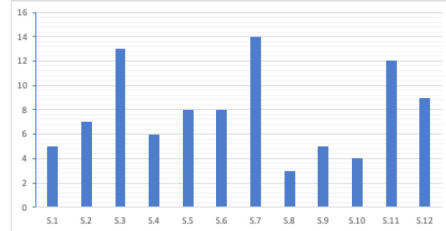
During the experiment, each person followed the entire guide, step by step, carrying out all proposed self-evaluation exercises. For each trainee who participated in the experiment the following quantitative and qualitative information was recorded:

- Time required to complete each section of the guide (quantitative)
- Valuation of user-friendliness (quantitative)
- Satisfaction level (quantitative)
- Strengths points (qualitative)
- Weak points (qualitative)
- Possible improvements (qualitative)



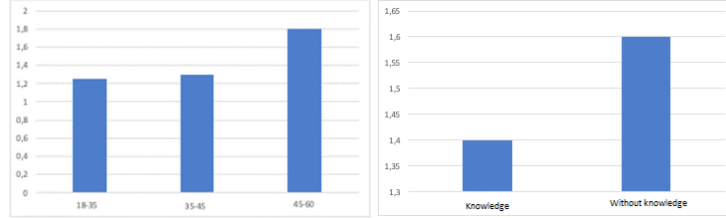
**Fig. 2.** Experimental study.

Fig. 3 shows the average time (minutes) required to complete each of the 12 sections the guide is divided into (Section 2.2). Each bar represents a section, from S1 to S12. The section that was the fastest in being completed is number 8, where the user learns how to execute a program cyclically, a very agile operation that demands short time. On the contrary, section 7 was the longest one, where the trainee learns how to program different types of movements of the robot arm. In addition, recording all the points and the movements from one point to another takes time. This explains the different times required by some sections. In any case, all times required by each section are reasonable and short enough to be easily handled during the training.



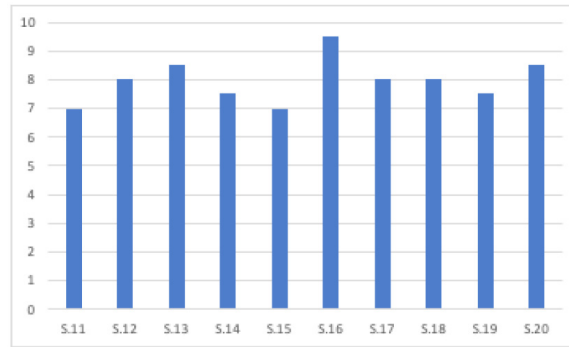
**Fig. 3.** Average time (minutes) to complete each section of the guide.

Fig. 4 shows the average time required to complete the guide, considering the age range and the level of knowledge on robotics. As expected, older people need more time as they are not so familiar with technology. The people within the range of 45-60 years old requires an approximately 50% more time than the younger ones. This difference is not so noticeable when we consider the previous experience with robots. Of course, people without knowledge require always longer times, no matter the age, but in this case the difference is only 14%, that is, 1.4 hours instead of 1.6 hours. In any case, the time required to complete the guide is short, which proves its usefulness.



**Fig. 4.** Average time (hours) to complete the guide classified by age range (left) and previous knowledge (right)

Fig. 5 shows the global value assigned by users to each section of the guide. In general, the guide receives a very positive satisfaction level, being the minimum 7 and the maximum 9.5, with an average satisfaction mark of 8.



**Fig. 5.** User satisfaction level per section

In addition to the quantitative results, qualitative information about the user experience was collected. The main strengths highlighted by the users were:

- Very visual guide, simple to understand, you can find the information you are looking for quite easily.
- The support of the videos is a great tool for a more dynamic learning, making easy to follow each step in a clearer way.
- It has an adequate extension for a medium-advanced learning of the use of the robot arm.
- The hyperlinks of the dynamic guide makes learning faster and more comfortable when going through the different elements of the Guide.
- Having a self-evaluation makes each person aware of its learning progress.

A possible identified improvement was to include more examples of use. Accordingly, it has been included as future works.

## 4 Conclusions and future works

In this work, we propose and evaluate an interactive handbook for industrial robotic arms self-learning. To speed up the search for specific information and the training process, the guide is organized by functionalities rather than by components. This makes it easier to learn the most standard robotic arm procedures. The guide includes a step-by-step approach, a video demo, and a self-assessment activity for each section. The guide has been tested in the lab by users with different background and age range. It has proved to be an efficient and friendly tool to provide quick instruction to people who have no prior knowledge with those robotic devices.

As future works, we plan to carry out the evaluation with a larger amount of people, perform a quantitative and qualitative comparison with other tools, and incorporate the recommendations proposed by some users as: to include more video examples, translate the content to other languages, and to include sections to program the robot in a higher-level programming language. In addition, other robotic arm brands examples will be included in the guide.

## Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Innovation under the project MCI/AEI/FEDER number RTI2018-094902-B-C21.

## References

1. Roldán-Gómez JJ, de León Rivas J, Garcia-Auñón P, Barrientos A (2020) A review on multi-robot systems: current challenges for operators and new developments of interfaces. *Revista Iberoamericana de Automática e Informática Industrial* 17(3):294-305. <https://doi.org/10.4995/riai.2020.13100>
2. Espinosa F, Santos C, Sierra-García JE (2020) Transporte multi-AGV de una carga: estado del arte y propuesta centralizada. *Revista Iberoamericana de Automática e Informática industrial* 18(1):82-91.
3. Lerma E, Costa-Castelló R, Griño R, Sanchis C (2021) Tools for teaching digital control in engineering degrees. *Revista Iberoamericana de Automática e Informática Industrial* 18(2):189-199. <https://doi.org/10.4995/riai.2020.13756>
4. García-Álvarez FM., Santos M (2020). Educational-Oriented Mobile Robot: Hidden Lessons. In *International Conference on EUropean Transnational Education* (pp. 61-71). Springer, Cham.
5. Santos G, Mandado E, Silva R, Doiro M (2019) Engineering learning objectives and computer assisted tools. *European Journal of Engineering Education* 44(4):616-628.
6. Martín-Gutiérrez J, Guinters E, Pérez-López D (2012) Improving strategy of self-learning in engineering: laboratories with augmented reality. *Procedia-Social and Behavioral Sciences* 51:832-839.
7. Díaz JM, Costa-Castelló R, Dormido S (2021) Un enfoque interactivo para el análisis y diseño de sistemas de control utilizando el método del lugar de las raíces. *Revista Iberoamericana de Automática e informática industrial* 18(2):172-188.

8. Pérez J, Fornas D, Marín R, Sanz PJ (2017) UWSim, un simulador submarino conectado a la nube como herramienta educativa. *Revista Iberoamericana de Automática e Informática Industrial* 15(1):70-78.
9. Sierra-García JE, Santos M (2021) Lookup table and neural network hybrid strategy for wind turbine pitch control. *Sustainability* 13(6):3235.
10. Sierra-García JE, Santos M (2020) Mechatronic modelling of industrial AGVs: a complex system architecture. *Complexity*, vol. 2020.
11. Nakimuli W, Garcia-Reinoso J, Sierra-Garcia JE, Serrano P, Fernández IQ (2021) Deployment and evaluation of an industry 4.0 use case over 5G. *IEEE Communications Magazine* 59(7):14-20.
12. ABB (2022) Available: <https://new.abb.com/products/robotics/es/robots-industriales/irb-120>. Accessed 28 March 2022
13. Fanuc (2022) Available: <https://www.fanuc.eu/es/es/robots>. Accessed 28 March 2022.
14. GP25 (2022) Available: [https://www.yaskawa.es/productos/robots/handling-mounting/productdetail/product/gp25\\_699](https://www.yaskawa.es/productos/robots/handling-mounting/productdetail/product/gp25_699). Accessed 26 March 2022.

# Gamifying the classroom for the acquisition of skills associated with Machine Learning: a two-year case study

Antonio M. Durán-Rosal, David Guijo-Rubio, Víctor M. Vargas, Antonio M. Gómez-Orellana, Pedro A. Gutiérrez, Juan C. Fernández

**Abstract** Machine learning (ML) is the field of science that combines knowledge from artificial intelligence, statistics and mathematics intending to give computers the ability to learn from data without being explicitly programmed to do so. It falls under the umbrella of Data Science and is usually developed by Computer Engineers becoming what is known as Data Scientists. Developing the necessary competences in this field is not a trivial task, and applying innovative methodologies such as gamification can smooth the initial learning curve. In this context, communities offering platforms for open competitions such as *Kaggle* can be used as a motivating element. The main objective of this work is to gamify the classroom with the idea of providing students with valuable hands-on experience by means of addressing a real problem, as well as the possibility to cooperate and compete simultaneously to acquire ML competences. The innovative teaching experience carried out during two years meant a great motivation, an improvement of the learning capacity and a continuous recycling of knowledge to which Computer Engineers are faced to.

---

Antonio M. Durán-Rosal

Dept. of Quantitative Methods, Universidad Loyola Andalucía, e-mail: amduran@uloyola.es

David Guijo-Rubio

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: dguijo@uco.es

Víctor M. Vargas

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: vvargas@uco.es

Antonio M. Gómez-Orellana

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: am.gomez@uco.es

Pedro A. Gutiérrez

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: pagutierrez@uco.es

Juan C. Fernández

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: jfcaballero@uco.es

## 1 Introduction

Machine Learning (ML) [2] allows computers to learn from data without having to be programmed to do so. This area, which combines knowledge of statistics, mathematics and artificial intelligence, is part of what is known as Data Science (DS) [1]. DS is based on developing scientific methods, processes and systems to extract knowledge from existing datasets to identify, analyse, understand or even improve current processes in these fields. DS and ML are gaining special interest in recent years due to their great interdisciplinarity. Numerous fields are benefiting from the advances in these areas, such as meteorology [13] or renewable energies [11], among others. Furthermore, it is common to find organisational processes in any sector and industry applying these techniques [4]. Moreover, governments have moved toward creating governmental strategies around Artificial Intelligence (AI). It is not difficult to find promising numbers on DS and ML collected by different organisations, e.g. *LinkedIn* [9], *Glassdoor* [16], the *US Bureau of Labor Statistics* [5] or *IBM* [20].

A person who works with DS and ML models is known as a Data Scientist. Their main objective is to understand and analyse the fundamental phenomena that happen, using techniques and theories drawn from various fields. Specifically, the Data Scientist profile is related to knowledge in mathematics, statistics, programming languages, and practical experience in analysing real data and elaborating predictive models. Until now, this discipline has been developed through different professional profiles, such as mathematicians or statisticians, although in recent years, the profile of Computer Engineer has been the most appropriate, given the knowledge and skills in the aforementioned areas.

The critical interdisciplinarity and the need for automatic and efficient information processing make the profession of Data Scientist one of the most sought-after jobs today. However, as it is a theoretical and practical area at the same time, it is a handicap for students to acquire the necessary competences at the university stage. Hence, universities must provide not only theoretical but also practical knowledge, which is difficult due to the breadth of knowledge to be satisfied in the BSc in Computer Engineering. Recently, one of the ways to make lessons more practical and entertaining is gamification. It is defined as the use of game design elements in non-game contexts [10]. Furthermore, it has received increased attention and interest in academia and practice due to its motivational power [26]. Competitions between students, including points, leader-boards, rewards and prizes, are considered one of the ways to include gamification in the classroom. In this context, platforms such as *Kaggle*<sup>1</sup>, founded in 2010, have emerged with the primary objective of connecting companies whose main function is to provide data to solve a given problem through competitions, with researchers and Data Scientists, who bring solutions to these problems through existing techniques in DS and, more specifically, ML.

Motivated by previous works of teaching DS and ML to students of the BSc Computer Engineering [3, 25, 15, 22], the major objective pursued with this work was

---

<sup>1</sup> <https://www.kaggle.com>



to complement the knowledge, mainly theoretical, taught in the different subjects of this Degree, with a completely practical experience, repeating the process during two years in order to extract more consistent conclusions. With such practical experience, the students of the different subjects benefited from professionals trained in this type of task, given that this is the field of research of the professors associated with these subjects [17]. In this way, the students could apply a wide range of techniques presented during the development of the subjects and, thus, face the complexity of a real problem. Besides, and transversely to the project, another competence developed was the ability to work remotely.

For the reasons above, *Kaggle* was used as an Information and Communication Technologies (ICT) tool in two subjects of the BSc in Computer Engineering offered by the University of Córdoba (Spain). In this way, the primary theoretical learning received during the BSc is complemented and applied to an eminently practical problem through a competition via *Kaggle*. The underlying idea was that competition, considered an element of gamification [18, 6], would increase the students' motivation to find the best solution to the proposed problem. Hence, the students would face the standard workflow that is carried out when tackling a problem of this type, consisting of preprocessing of the databases, design and training of the models, validation of the models obtained, adjustment of parameters, and finally, evaluation of the predictive models. This innovation experience was part of the development of two Teaching Innovation Projects (TIPs) developed by the authors of this work. Both TIPs were awarded by the University of Córdoba according to one of the priority lines of action established in the modality: transfer of theoretical knowledge into practice.

Before the *Kaggle* competition, the teaching staff associated with the subject tutored lent their experience in the field of ML and DS to address a real problem and provided guidance on how to tackle the competition. The main objective of these workshops was to provide specific training, on the *python*<sup>2</sup> programming language and the *scikit-learn* library [23]. It is important to highlight the use of these tools in ML: *python* is the most widely used language for these tasks due to its enormous versatility, learning curve, and ease of use, among other characteristics [27]. In addition, the students used only free software throughout the TIPs, joining the line of action in favour of it set by the Conference of Rectors of Spanish Universities and with the support and collaboration of the *Free Software Group*<sup>3</sup> of the University of Córdoba. On the other hand, *InClass Competitions* service in *Kaggle* was used. This service is offered free to the teaching and research community for its use with students. Its main objective is to make it easier for DS faculty to run a competition exclusively for students so that the level is tailored to them, and the remainder of the *Kaggle* community does not have access. The results of the competitions were very satisfactory, not only because of the increase in knowledge on the part of the students but also because of the high level of motivation with which they attended the

---

<sup>2</sup> <https://www.python.org>

<sup>3</sup> <https://www.uco.es/aulasoftwarelibre/>

workshops. Besides, the level and rate of participation in the competitions increased during the second year.

## 2 Gamification in Data Science and Machine Learning

As mentioned above, gamification includes game elements in non-game contexts. This supports motivation and performance in skills such as problem-solving, collaboration and communication, which are essential competencies in the Data Scientist profession. In the last decade, gamification has supported learning in many application areas. Of particular interest are ML and DS [19], being Computer Engineering one of the areas with the most prominent application of gamification [21].

Points, leaderships, avatars, prizes and rewards are the most common elements in gamification experiences, so competitions are the ultimate way to bring all these components together. In this regard, a game-based competition for teaching artificial intelligence was proposed in [7]. For this purpose, the Nine Men's Morris game was selected to develop an artificial player using heuristic search, optimisation and ML concepts. The competition consisted of a round-robin tournament where each player had to play against the others. The results confirmed that the experience was a success. The students reported an improvement in their skills and declared an increased interest in the course topics, and in AI in general. The effectiveness of the learning approaches adopted was also confirmed.

More important, and as aforementioned, *Kaggle* is the platform par excellence for carrying out ML problem-solving competitions. Related to this point, a methodology for teaching ML using a game-like *Kaggle* competition to make the learning more engaging and fun was proposed in [8]. Specifically, the methodology consisted of seven steps, including 1) teaching knowledge, 2) briefing competition guidelines, 3) giving sample solutions, 4) registering for the competition, 5) submitting and discussing the results, 6) reflecting on the learning experience, and 7) scoring the teams. As a result, students were motivated to look for different ways to solve problems using ML techniques, and also to improve their understanding on how to apply the theoretical concepts and algorithms to real-world problems.

In [24], the authors conducted a study to determine whether the use of *Kaggle InClass* enhances problem learning in DS and ML. Sixty-one students were divided into two groups for regression and classification problems, building prediction models individually for 16 days, and forming groups for other 7 days. The students that participated in the competition performed better in the exam. Moreover, they found the competition exciting and valuable for their learning in the course.

More recently, García-Algarra [12] proposed a methodology for teaching ML concepts in non-STEM (Science, Technology, Engineering and Maths) areas. The course went through different phases to teach a group of people without experience in this field or programming about related concepts. In the last phase of the course, there was a *Kaggle*-like competition forming groups between people who did not know each other, which was worth 10% of the course grade, with the winning group getting an additional 10%. This provoked a high degree of interest and effort, which contributed to improve the understanding of the concepts taught regarding ML.

### 3 Development of the innovation experience

This work was focused on the subjects “Introduction to Machine Learning” and “Introduction to Computational Modelling” of the 3rd and 4th year of the BSc Computer Engineering of the University of Córdoba, respectively, during the academic years 2017/2018 and 2018/2019.

#### 3.1 Objectives

The main objectives associated with the implementation of the two TIPs were the following:

1. To improve the acquisition of practical aspects of the specific competences belonging to the Data Scientist professional profile.
2. To instruct students in the use of the *Kaggle* platform as a tool to learn the applicability of ML concepts in a fun and entertaining way.
3. To broaden the knowledge of programming languages and DS.
4. To combine a cooperative and competitive environment for problem-solving.
5. To foster an atmosphere of cooperation and competitiveness at the same time among the students.
6. To confront students with the complexity of solving real-world problems using ML techniques.

#### 3.2 Materials and methods

The above objectives were carried out through the activities described below, which were common to the two TIPs.

**Activity 1. Development of the competition.** The first step was to find a real-world problem resulting attractive for the students. For this, a real significant wave height prediction problem in the Gulf of Alaska (United States) was selected. This problem, focused on the need to anticipate events in the natural environment and in the field of renewable energies, had been previously tackled by members of the research group, thus, its difficulty could be adapted to be affordable by the students of the subjects. The dataset was built using 18 input variables related to meteorological observations, which were both obtained from sensors installed on buoys and also from a reanalysis model. For the significant wave height prediction, the continuous variable was discretised into 4 categories according to the significant wave height: *low*, *medium*, *moderate* and *very high*. Then, the problem was adapted according to the *Kaggle* platform, which requires three datasets: 1) the training dataset for building the models, 2) the public testing dataset to validate them, and 3) the private testing dataset to check the fairness of the public validation. For these two validations, the students had to upload the predictions obtained by their developed models to the *Kaggle* platform, which automatically generates a public and a private ranking. Note that the private ranking on a previously unseen testing dataset is only displayed at the end of the competition.

**Activity 2. Multi-session practice.** In order to introduce the students to the *Kaggle* platform, a 3-session practice was carried out. In these sessions, it was explained how the platform works and the real problem they were going to tackle. For this, the dataset detailed in Activity 1 is considered as well as several case studies were proposed as examples, providing their source code, so that they could test and practice with different ML methods, favouring the understanding of each of them.

Furthermore, the work performed by the students in this Activity was going to be evaluated using the following criteria: 40% of the mark was scored based on the private ranking as it measures the quality of the developed model, and the remaining 60% was scored according to the workflow implemented by the students to obtain the models: preprocessing and visualisation of the data, analysis of data (outliers, extreme values and correlated variables), feature's selection, etc. After the first session, the competition was launched, closing one week after the last practical session.

**Activity 3. Working team for tackling complex competitions.** In order to tackle complex competitions on *Kaggle*, a working team was established between teachers and students. The main goal of this working team was to cooperate for the benefit of the team and to compete against other teams. Meetings were planned frequently during tutoring hours, both face-to-face and virtually.

**Activity 4. ML training.** Apart from the multi-session practice, a series of workshops on ML were given by the teachers in collaboration with the *Free Software Group* of the University of Córdoba. The main objective was to complement the training received by the multi-session practice (Activity 2). Up to now, the students were instructed to create predictive models using Weka [14]. Therefore, it was a good opportunity to teach the *python* programming language together with the *scikit-learn* library. Two sessions were held each year<sup>4</sup>.

**Activity 5. Evaluation of the TIPs.** For assessing the practical knowledge acquired by the students of both subjects, two voluntary test-type questionnaires were carried out: the first one before the start of the multi-session practice, and the second one after the end of the competition. Both questionnaires included the same 20 questions: 16 for assessing the students' level regarding ML, and the remaining 4 regarding self-assessment questions, on a Likert scale, about the *Kaggle* platform, *python* and *scikit-learn*.

## 4 Results

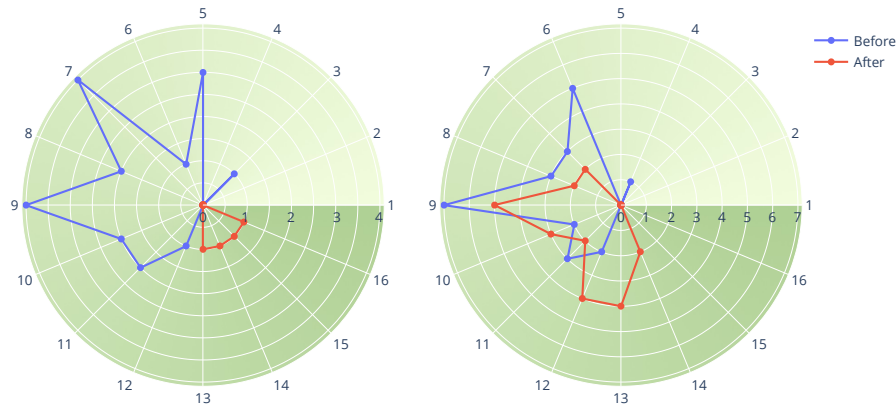
In this section, the results of both TIPs are presented, compared and discussed. These results are divided between those obtained through Activity 5, reflecting the evaluation of the TIPs through the questionnaires completed by the students, and those obtained from the teaching innovation experience, through the considerations of the work team.

---

<sup>4</sup> Please, check the agenda of each tutorial in the GitHub repositories <https://github.com/ayrna/tutorial-scikit-learn-asl> and <https://github.com/ayrna/taller-sklearn-asl-2019>, for the first and the second TIP, respectively.

#### 4.1 Assessment questionnaires

The forms were completed in two phases, before carrying out the set of activities and after their completion. Moreover, the form was divided into two parts, the first part, consisted of 16 multiple-choice questions, to obtain the students' level and 4 self-assessment Likert scale questions focused on the main tools used in the project. First, the results of the multiple-choice questions are analysed before and after the activity for both years considered. After that, the self-assessment results are presented, again for both years and before and after the sessions.



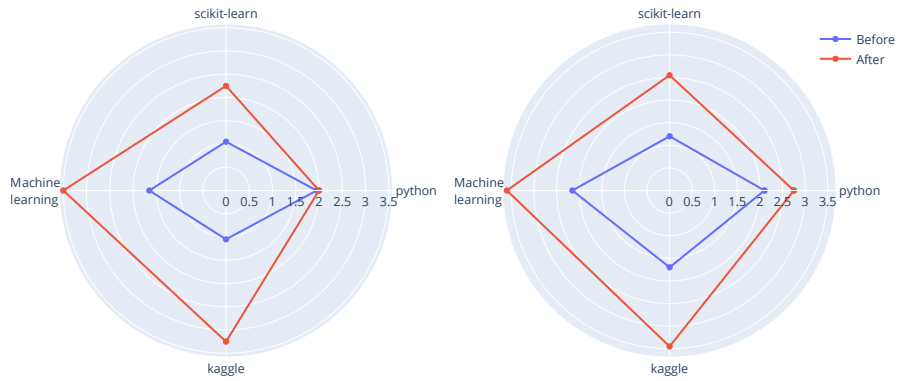
**Fig. 1** Results obtained in the multiple-choice questions before and after the sessions for the first (left) and the second (right) years. Outer values represent the total score obtained (from 1 to 16) and the inner values indicate the number of students who obtained each of these scores.

In the first year, a total of 19 students took part in the questionnaires. While all of them did the initial tests, only a small part of them completed the final test. The completion of the test form before the sessions allowed us to know the initial level of the students in relation to ML and DS tasks. Each of the questions of the first part were evaluated with 1 point, thus reaching a maximum of 16 points. The average score obtained by the students before the sessions was 7.9 out of 16, which is 4.9 out of 10. Regarding the test performed at the end of the workshops, the average score raised to 14.5 out of 16, which is 9 out of 10. Also, the range of scores went from 3-12 to 13-16. These results are shown in a graphical way in the left chart of Fig. 1, where the outer values represent the total score obtained (from 1 to 16) and the inner values indicate the number of students who obtained each of these scores. For instance, two students obtained a score of 11 in the questionnaire carried out before the sessions, whereas one student obtained a score of 15 in the questionnaire performed after the sessions.

In the second year, the total number of students that took part in the activity raised to 26. The evaluation method was the same as the one described for the first year. For the first part, the average score obtained by the students before taking part in this

project was 8.4 out of 16, which is 5.3 out of 10. Then, after their participation in the organised sessions, their average score raised to 10.6 out of 16, which is 6.6 out of 10. Also, the interval of scores was shifted to the upper part of the ratings, given that in the initial test it was 4-12 and, in the end, it was 7-14. These results, which show that the students improved their knowledge and ML skills, are graphically shown in the right chart of Fig. 1.

Both charts represented in Fig. 1 show that the scores of the questionnaires taken before the activity are concentrated on the low-medium part of the rating range while, those taken after the sessions, they are mostly located on the medium-high part. In fact, in the first year, there is no overlapping between both scores, given that the final results are only in the 13-16 range, i.e. the 4 students obtained excellent scores. For the case of the second year, there is an important overlapping between both sets of scores, but the scores obtained after the activity are clearly biased towards the highest scores.



**Fig. 2** Results obtained in the self-assessment questions before and after the sessions for the first (left) and the second (right) years. The magnitude of the chart represents the mean score obtained by the students on each of the skills (outer values) following the Likert scale (from 0 to 4).

Concerning the second part of the assessment questionnaires, which is related to the self-assessment questions, the results for both years are shown in Fig. 2. The left chart shows the results for the first year while the second is related to the second year. On each of the charts, the results before and after the activity are shown. In this case, the magnitude represented by the chart is the average value obtained, following the Likert scale (from 0 to 4), by the students on each of the skills (outer values). It is worth noting that, in both years, the polygon related to the skills before the activity is contained in the polygon obtained with the final results. This fact shows that the students have improved their knowledge related to all the technologies considered. The results for both years are quite similar, but the improvement achieved in the second year was more obvious than the one obtained in the first.

To sum up, we can obtain some general conclusions related to the obtained results for both years:

- The students improved their general knowledge about ML and DS tasks.
- The individual results on each of the topics or technologies have also been improved and the students are aware of the improvement, as was shown by the self-assessment results.
- The level of interest and participation on the second year was higher than in the first one.

## 4.2 Results of the teaching innovation experience

The results obtained from this teaching innovation experience are as follows:

- The students were made aware of the importance and interest that the professional profile of Data Scientist has nowadays, as well as the growing demand that exists around this figure, becoming a successful professional career for the students of the BSc in Computer Engineering.
- The students were introduced to the various tasks performed by the Data Scientist, including knowledge of mathematical and statistical notions of the models used.
- The students were introduced to a series of professional and current technologies for modelling real problems using ML techniques, such as *python* and the *scikit-learn* library.
- It ensured that students were able to contribute and receive new practical knowledge from the large community on the *Kaggle* platform, made up of well-established researchers and Data Scientists from a wide range of fields.
- The use of ICT in the classroom was encouraged and good participation in the workshops, which increased in the second year, was achieved.
- Group work was promoted by encouraging cooperation between participants in the same group and healthy competition between groups, so that the feedback and relationships between them culminated in improved knowledge for all.
- The students' self-confidence to get involved in real problems where they can apply their theoretical and practical knowledge to provide a tangible solution was increased.
- Carrying out the same type of workshop in two different years helped reinforce the robustness of the results achieved with them, given that, in both years, the results were similar and in both cases were quite positive.

## 5 Conclusions

After analysing the results, we believe that this kind of TIPs substantially improve the student's ability to face a real work situation after finishing their degree. In this sense, we worked with the Data Scientist profile, one of the most demanded profiles in today's society, having a great impact worldwide. The projects developed across two different years provided students with the motivation and capacity for continuous learning and recycling of knowledge that graduates in Computer Engineering must undergo.

Undoubtedly, the degree of participation and interest of the students in both years was very high, especially in the second year, where a higher number of students took part in the workshops. They spent a lot of time working on the activities that were prepared for these projects due to the difficulty and the study of a novel field that has not been approached by them so far. As a result of these experiences in *Kaggle*, several students became interested in carrying out Final Degree Projects related to research, ML and DS.

It is worth noting that the effort made by the organiser team that worked on the TIPs was notable. It was necessary to adapt all the workshop material and the proposed work for the activity to the level of the students. Nevertheless, the level of satisfaction achieved made it enormously worthwhile. In the future, new editions are expected to be held to increase the knowledge of this job profile, which is currently in demand and growing during the last years.

**Acknowledgements** The two Teaching Innovation Projects have been funded by the University of Córdoba with references 2017-1-5008 and 2018-1-5015. This work has also been partially subsidised by the “Agencia Española de Investigación (España)” (grant reference: PID2020-115454GB-C22 / AEI / 10.13039 / 501100011033); the “Consejería de Salud y Familia (Junta de Andalucía)” (grant reference: PS-2020-780); and the “Consejería de Transformación Económica, Industria, Conocimiento y Universidades (Junta de Andalucía) y Programa Operativo FEDER 2014-2020” (grant references: UCO-1261651 and PY20\_00074). David Guijo-Rubio’s teaching was funded by the University of Córdoba through grants to Public Universities for the requalification of the Spanish university system from the Ministry of Universities funded by the European Union - NextGenerationEU (Ref. UCOR01MS). The teaching of Víctor M. Vargas was funded by the “Programa Predoctoral de Formación al Profesorado Universitario (FPU)” of the Ministry of Science, Innovation and Universities (Ref. FPU18/00358). The teaching of Antonio M. Gómez-Orellana was funded by “Consejería de Transformación Económica, Industria, Conocimiento y Universidades de la Junta de Andalucía” (Ref. PREDOC-00489).

## References

1. van der Aalst, W.: Data Science in Action, pp. 3–23. Springer Berlin Heidelberg, Berlin, Heidelberg (2016)
2. Alpaydin, E.: Machine learning. MIT Press (2021)
3. Brunner, R.J., Kim, E.J.: Teaching data science. *Procedia Computer Science* **80**, 1947–1956 (2016)
4. Brynjolfsson, E., Mitchell, T., Rock, D.: What can machines learn, and what does it mean for occupations and the economy? In: *AEA Papers and Proceedings*, vol. 108, pp. 43–47 (2018)
5. Bureau, E.T.: 11.5 mn job openings by 2026, sky-high salaries: Why data science is booming. *The Economic Times* (2020). URL <https://bit.ly/3tsDVwZ>
6. Caponetto, I., Earp, J., Ott, M.: Gamification and education: A literature review. In: *European Conference on Games Based Learning*, vol. 1, p. 50. Academic Conferences International Limited (2014)
7. Chesani, F., Galassi, A., Mello, P., Trisolini, G.: A game-based competition as instrument for teaching artificial intelligence. In: *Conference of the Italian Association for Artificial Intelligence*, pp. 72–84. Springer (2017)



8. Chow, W.: A pedagogy that uses a kaggle competition for teaching machine learning: an experience sharing. In: 2019 IEEE International Conference on Engineering, Technology and Education (TALE), pp. 1–5. IEEE (2019)
9. Columbus, L.: LinkedIn's fastest growing jobs today are in data science and machine learning. Retrieved April 10, 2018 (2017)
10. Deterding, S., Dixon, D., Khaled, R., Nacke, L.: From game design elements to gamefulness: Defining "gamification". In: Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek '11, p. 9–15. Association for Computing Machinery, New York, NY, USA (2011)
11. Durán-Rosal, A., Fernández, J., Gutiérrez, P., Hervás-Martínez, C.: Detection and prediction of segments containing extreme significant wave heights. *Ocean Engineering* **142**, 268–279 (2017)
12. Garcia-Algarra, J.: Introductory machine learning for non stem students. In: European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, pp. 7–10. PMLR (2021)
13. Guijo-Rubio, D., Casanova-Mateo, C., Sanz-Justo, J., Gutierrez, P., Cornejo-Bueno, S., Hervás, C., Salcedo-Sanz, S.: Ordinal regression algorithms for the analysis of convective situations over madrid-barajas airport. *Atmospheric Research* **236**, 104798 (2020)
14. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. *ACM SIGKDD explorations newsletter* **11**(1), 10–18 (2009)
15. Hicks, S.C., Irizarry, R.A.: A guide to teaching data science. *The American Statistician* **72**(4), 382–391 (2018)
16. Jackson, A.E.: The 50 best jobs in america for 2018. Glassdoor (2018). URL <https://bit.ly/3vN5EKL>
17. Kross, S., Guo, P.J.: Practitioners teaching data science in industry and academia: Expectations, workflows, and challenges. In: Proceedings of the 2019 CHI conference on human factors in computing systems, pp. 1–14 (2019)
18. Lee, J.J., Hammer, J.: Gamification in education: What, how, why bother? *Academic exchange quarterly* **15**(2), 146 (2011)
19. Legaki, Z., Hamari, J.: Gamification in statistics education: A literature review. In: GamiFIN Conference 2020, CEUR workshop proceedings, pp. 41–51 (2020)
20. Miller, S., Hughes, D.: The quant crunch: How the demand for data science skills is disrupting the job market. Burning Glass Technologies (2017)
21. Milosz, M., Milosz, E.: Gamification in engineering education—a preliminary literature review. In: 2020 IEEE Global Engineering Education Conference (EDUCON), pp. 1975–1979. IEEE (2020)
22. National Academies of Sciences, Engineering, and Medicine and others: Data science for undergraduates: Opportunities and options. National Academies Press (2018)
23. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011)
24. Polak, J., Cook, D.: A study on student performance, engagement, and experience with kaggle inclass data challenges. *Journal of Statistics and Data Science Education* **29**(1), 63–70 (2021)
25. Ramamurthy, B.: A practical and sustainable model for learning and teaching data science. In: Proceedings of the 47th ACM Technical Symposium on Computing Science Education, pp. 169–174 (2016)
26. Sailer, M., Homner, L.: The gamification of learning: A meta-analysis. *Educational Psychology Review* **32**(1), 77–112 (2020)
27. Stack-Overflow, I.: Stack overflow developer survey 2021 (2021). URL <https://bit.ly/3y0Khbb>

# Hackathon in teaching: applying machine learning to Life Sciences tasks

David Guijo-Rubio, Víctor M. Vargas, Javier Barbero-Gómez, Jose V. Die, Pablo González-Moreno

**Abstract** Programming has traditionally been an engineering competence, but recently it is acquiring significant importance in several areas, such as Life Sciences, where it is considered to be essential for problem solving based on data analysis. Therefore, students in these areas need to improve their programming skills related to the data analysis process. Similarly, engineering students with proven technical ability may lack the biological background which is likewise fundamental for problem-solving. Using hackathon and teamwork-based tools, students from both disciplines were challenged with a series of problems in the area of Life Sciences. To solve these problems, we established work teams that were trained before the beginning of the competition. Their results were assessed in relation to their approach in obtaining the data, performing the analysis and finally interpreting and presenting the results to solve the challenges. The project succeeded, meaning students solved the proposed problems and achieved the goals of the activity. This would have been difficult to address with teams made from the same field of study. The hackathon succeeded in generating a shared learning and a multidisciplinary experience for their professional training, being highly rewarding for both students and faculty members.

**Key words:** professional experience; interdisciplinarity; artificial intelligence; professional profile

## 1 Introduction

Learning programming languages has become established in recent years in disciplines beyond the traditional ones related to Computer Sciences. These tools are used widely across all STEM (Science, Technology, Engineering and Mathematics)

---

David Guijo-Rubio

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: dguijo@uco.es

Víctor M. Vargas

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: vvargas@uco.es

Javier Barbero-Gómez

Dept. of Computer Science and Numerical Analysis, University of Córdoba, e-mail: jbarbero@uco.es

Jose V. Die

Dept. of Genetics, University of Córdoba, e-mail: jose.die@uco.es

Pablo González-Moreno

Dept. of Forest Engineering, University of Córdoba, e-mail: pablo.gonzalez@uco.es

areas in general and in the field of Life Sciences in particular. This last field has benefited from the advantages of solving current procedures without entailing an excessive workload [10, 11]. This new scenario requires universities to incorporate the learning of programming languages and the work in multidisciplinary teams as transversal competencies.

With this aim, a Teaching Innovation Project was granted by the University of Cordoba for the academic year 2020/2021. This Project made an approach to this new scenario including students and faculty from different areas of knowledge (forestry engineering, computer science, genetics and agronomy), applying innovative teaching tools such as the hackathon [9, 14]. The Project consisted of a series of work sessions where real Life Sciences problems were presented to teams built by students from both fields (Data Science and Life Sciences), sharing knowledge throughout the whole work process. This hackathon tool also contributed to developing other skills, such as teamwork between different areas, interdisciplinarity and learning first-hand about a professional profile. More specifically, the profile of the data analyst, with enormous importance today and great future projection, both in the area of Computer Science [4] and in those related to Life Sciences [3, 13, 8].

The Project was mainly based on the resolution of two challenges: 1) the visualisation task associated with a problem in the area of Biotechnology and Genetics, and 2) the predictive analysis task, in the area of Agroforestry Engineering. In both challenges, real problems were considered being of interest to students and the scientific community. After the different work sessions, the results obtained were highly satisfactory, achieving many of the objectives set by the project and encouraging the students to carry out complementary curricular university training.

## **2 Development of the innovation experience**

The Teaching Innovation Project was focused on the following subjects belonging to the University of Córdoba (Spain): 1) Introduction to Computational Modelling (4th year - BSc Computer Engineering), 2) Mathematical Foundations of Programming (1st year - MSc Remote Sensing and Spatial Models applied to Forest Management) and 3) Analysis and Interpretation of Genomes (1st year - MSc Biotechnology).

### **2.1 Objectives**

The main objectives associated with the implementation of the Teaching Innovation Project were the following:

1. To improve the acquisition of practical aspects related to machine learning.
2. To broaden knowledge of programming languages and the basic concepts and notions associated with the Life and Natural Sciences.
3. To use competition dynamics for teaching purposes to solve problems.
4. To create an atmosphere of cooperation and competitiveness at the same time among the students, with the aim of progressing and achieving the best results in the resolution of both problems (visualisation and massive data modelling).

5. To confront students with the real complexity of the two problems from a multi-disciplinary point of view.
6. To develop presentation skills in front of specialised and non-specialised audiences.

## 2.2 Materials and methods

The above objectives were carried out through the activities described below.:

**Activity 1. Communication system of the event.** For the broadcasting of the event and as a central hub for the event material and resources, a public website<sup>1</sup> was created with all the information related to the Project activities. A total of 20 students formalised their enrolment in the teaching activity.

**Activity 2. Training guides and presentation of the challenges.** Aiming to generate basic knowledge in the area not related to each participant's training, two "Data Science Guides" were produced, for the programming languages in which the students are most fluent, Python and R, using the Google Collaboratory platform. In these guides, the basic notions of data loading, preprocessing and modelling were presented for developing solutions for the following problems: 1) Information visualisation and molecular feature extraction. This problem, associated with the area of Biotechnology and Genetics, consisted on analysing from a graphical point of view the main features of gene ontology in the process of functional gene annotation. And 2) Obtaining Machine Learning and Data Mining models. This problem, related to the area of Agroforestry Engineering, consisted on carrying out a predictive analysis and study of the forest biodiversity of Andalusia (South Spain). In addition, explanatory videos were provided with the theoretical contents of each challenge. After the informative sessions, a collaborative and competitive dynamic was proposed to the students to solve both challenges.

**Activity 3. Challenge 1: pre-processing of databases and data visualisation in datasets with biological information.** The main objective of this activity was to become familiar with the pre-processing, one of the most important preliminary phases in Data Analysis. For this first challenge, a topic related to the area of Biotechnology and Genetics was selected. The problem was focused on the functional annotation of molecular stress signalling in *Arabidopsis* [2]. The aim of this activity was twofold. On the biological side, the aim was to determine whether there are quantitative/qualitative differences in transcriptional activation in putrescine- and thermospermin-treated plants. On the computational side, it was required to generate automatic functional annotation analysis tools based on the standard gene ontology vocabulary [1]. The second objective of this activity was to visualise the data by means of various plots. There are numerous techniques in this area, which have generated enormous interest due to the numerous advantages they provide, serving as an initial phase in the analysis of a specific problem. Among other techniques, students were shown how to carry out a visual analysis of the distribution of the data, finding out if there was any error in them or what type of distribution was present in

---

<sup>1</sup> <https://biodatauco.github.io/> (in Spanish).

the data. On the other hand, visualisation is also of great interest for the analysis of the results, so that we can analyse the results of the different models or how accurate the prediction of each model has been and its fit with reality. In this sense, they were presented with a proposal to address the challenge, consisting of two sections, a descriptive one, where a principal component analysis [7] or cluster analysis [6] could be used, and a functional one, where the functional annotation itself would be carried out.

**Activity 4. Challenge 2: generation and evaluation of Machine Learning and Data Mining models.** This activity was the second challenge presented to the students. The topic of this second problem was more focused on discovering the secret patterns of plant diversity in Andalusia (Spain). For this purpose, the students were provided with a database with the inventories of vegetation in natural areas of Andalusia as well as the number of woody plant species that exist [5]. This database is unique in Andalusia, containing approximately 50,000 vegetation inventories to describe all the environmental variability of Andalusia. The objective of this activity was also dual, on the side of Forestry Engineering, the aim was to carry out an analysis of the richness of woody plants existing in the region of Andalusia. On the Computer Engineering side, the aim was to obtain a predictive model [12] of plant diversity as accurately as possible. In the second phase of the predictive analytics challenge, students were encouraged to interpret the results obtained in the model evaluation phase. The main idea was, given a model and a prediction obtained, to understand the fundamentals behind the methodology, as well as to interpret the results and generate knowledge from them.

### 3 Results

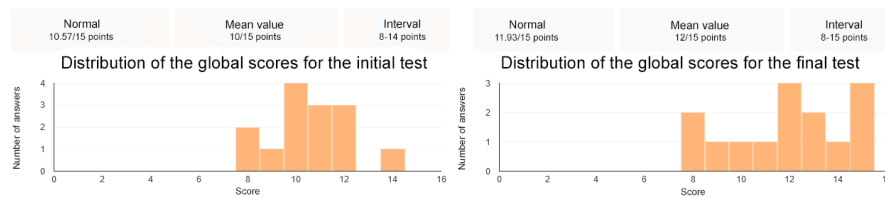
The results of the project were evaluated both quantitatively and qualitatively. On a quantitative level, questionnaires were carried out for students to self-evaluate their knowledge before and after the activities. A qualitative analysis of the results was also carried out, with special emphasis on the objectives achieved and future improvements.

#### 3.1 Knowledge assessment questionnaires

Two questionnaires were carried out using Google Forms. The first one was carried out prior to the start of the Project, with the aim of ascertaining the students' prior knowledge in the three areas of knowledge. For this purpose, a total of 15 questions were used (5 per area including data science, genetics and agroforestry) in which the students were asked about their basic knowledge. With this questionnaire, we were able to find out the starting point and which area needed to be emphasised the most. As expected, students in one discipline might have no prior knowledge of the other disciplines, so it was interesting to apply the questionnaire to see the level of knowledge in these disciplines. The results of the level questionnaire, carried out for a total of 14 students before the activity, showed that the average score obtained was

10.57 points out of 15 and the individual scores ranged from 8 to 14 points (graph on the left of Figure 1).

Similarly, the same questionnaire was done at the end of all the activities of the Project. It is important to note that after the completion of the first questionnaire the answers to any of the questions were not revealed, but only the basic concepts needed during the completion of the questions were explained. The main purpose of repeating the same questions was to see whether the students had consolidated the concepts they already had and improved those related to the other disciplines. In this case, we did expect the results of the questionnaire to improve. In particular, the average score rose to 11.93 out of 15 points, while the maximum of individual scores increased up to 15 points (graph on the right of Figure 1). Therefore, a noticeable improvement can be perceived with respect to the results obtained in the initial questionnaire.



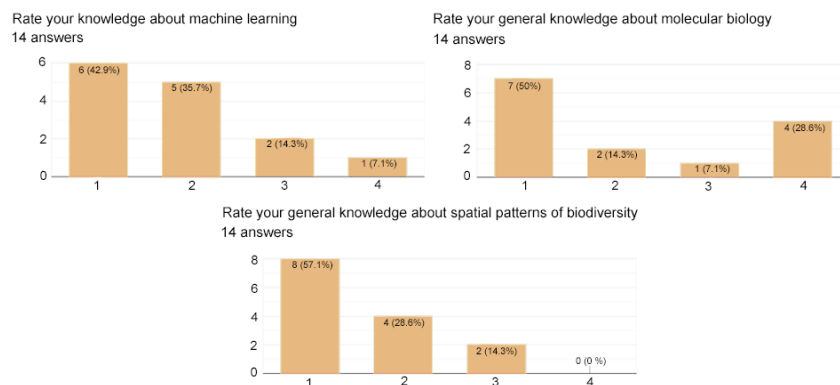
**Fig. 1** Results of the initial (left) and final (right) questionnaires.

### 3.2 Self-assessment

In order to self-assess their knowledge of the three disciplines, another questionnaire was carried out, also using Google Forms (4 questions on Data Science, 4 questions on genetics and 3 questions on agroforestry) before and after the teaching activity. The purpose of this self-assessment was to identify to what extent the students were aware of the knowledge they had from their perception. In these questionnaires there were not right or wrong answers, they simply evaluated their knowledge from their perspective. Figure 2 shows the results of the self-assessment forms.

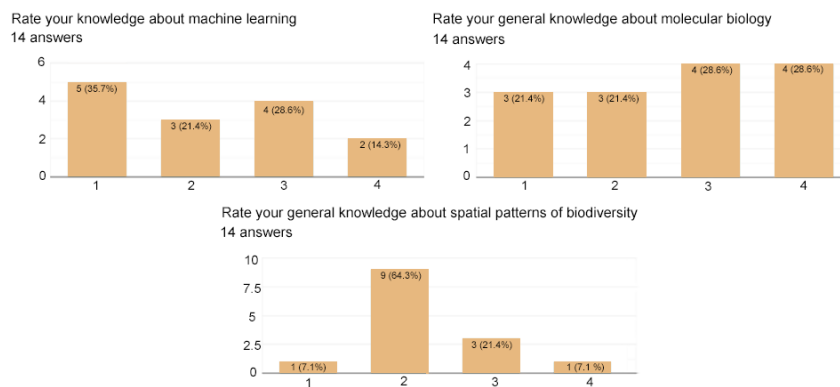
The students rated each of the above aspects with a value from 1 to 4, with 1 being the lowest level of knowledge or mastery of the topic and 4 being the highest. The initial results reflected that most students indicated that they did not have much overall mastery of the hackathon topics (Figure 2 shows one question of each area). For most of the questions, the results were concentrated in option 1. On the other hand, the number of results in option 4 was minimal in all questions.

At the end of all the Project activities, the self-assessment questionnaire was carried out again. Similarly, the questions were the same as in the previous self-assessment so that a comprehensive comparison of their vision could be made. The results of the final self-assessment were on average higher than those of the initial one (Figure 3 shows the same questions chosen in Figure 2). In the previous case, most of the values were concentrated at 1, whereas in this case they were more spread out and there were a few values at level 4, indicating that some students



**Fig. 2** Results of the initial self-assessment questionnaire.

felt that they gained some knowledge during the activity. In general, most of the values were concentrated in levels 2 and 3, indicating that most students did not consider themselves experts in all subjects after the activity (which is normal due to the duration of the activity), but they did recognise that they had a certain degree of knowledge about all the topics covered. Thus, the results were considered to be quite satisfactory.



**Fig. 3** Results of the final self-assessment.

### 3.3 Qualitative analysis

From a qualitative perspective, the teaching staff also carried out an evaluation of the Project through the following two activities.

#### 3.3.1 Presentation of the results obtained by the working groups

In these presentations, the students, divided into groups, made a presentation of the solutions proposed for each of the two tasks carried out. These presentations allowed

us to check in detail the work done by each group, the way they had analysed the problem and the method they had used to solve it. They were also of great interest because they allowed other students to see different solutions to the same problem and to detect elements that they had overlooked during their work.

### 3.3.2 Qualitative analysis of students' feelings

At the end of the last activity, a more informal survey was carried out using the Mentimeter tool<sup>2</sup>, with the aim of reflecting the students' feelings during their participation in the hackathon. Specifically, this Mentimeter asked three different questions (Figures 4 and 5). A total of 13 students participated in this final questionnaire.

The positive aspects on which most of the students agreed were collaboration and learning (left side of Figure 4). This is quite satisfactory as it reflects two of the objectives that we most wanted to reinforce in the Project. Moreover, the right side of Figure 4 shows the aspects that could be improved according to the hackathon participants. The main aspect pointed out is the lack of time to solve the challenges. In the future, an attempt will be made to spend less time on the introduction/presentation of the challenges so that the groups have more time to work.



**Fig. 4** Mentimeter: (left) What did you like most about the hackathon? and (right) What aspects would you improve? Bigger words were more repeated by the students in the questionnaire.

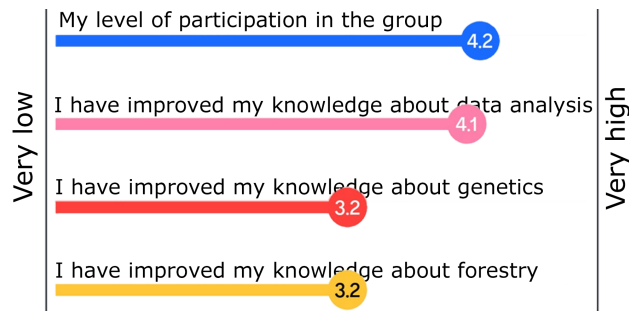
Finally, Figure 5 shows the average values of the self-evaluations made by each participant on their level of participation in the group and the improvement in their knowledge as a result of carrying out this activity. As can be seen, the level of participation of the participants within their group was very high, so we can confirm that the way in which the groups were created and the size of the groups was appropriate and correct.

### 3.4 Results of the teaching innovation experience

This section describes the results achieved during the realisation of the activity, as well as those that could not be fulfilled or were not achieved to the degree of

<sup>2</sup> <https://www.mentimeter.com/>





**Fig. 5** Mentimeter: Self-assessment of certain personal aspects such as their level of participation or the improvement they have seen in their knowledge thanks to the activity.

satisfaction expected. Firstly, the objectives (Section 2.1) that are considered to be achieved in view of the results obtained and described in Section 3 are presented:

- The students' ability to tackle real problems using machine learning techniques was improved, including an initial analysis of the data available to them, their visualisation, the necessary preprocessing required to work with them and the generation of models that allow predictions to be made on the basis of the data available to them.
- Theoretical and practical concepts related to genetics and forest diversity were acquired by solving problems related to these areas. In order to be able to undertake them, expert knowledge on the subject of the problem was necessary.
- Knowledge of data analysis and programming languages was improved through the guides and videos provided by the activity's teaching staff.
- Cooperation between members of different groups and disciplines was fostered, as well as healthy competition between groups in order to achieve the best results. Competitions for teaching purposes proved to motivate students, in this case, to try to find the best solution to the proposed problems.
- Students tackled real problems with the difficulty that they entail and, moreover, it was done in a limited time, so that multidisciplinary cooperation within each group was essential in order to solve the problems presented.
- The students developed their ability to explain the results of their work in public, so that people from different areas could understand the results they reached and how they achieved them. In addition, they were also able to explain the difficulties they encountered in arriving at a solution and the way in which they were able to solve them.

On the other hand, we describe the objectives we consider that have not been achieved during the implementation of this Project:

- The groups were designed seeking the presence of at least one profile from each discipline. However, in some cases, not all participants were able to attend all the sessions.

- In some cases the challenges presented were not fully met, mainly due to lack of time. In a way, this was expected as the challenges were of a certain complexity and the time given to solve them was quite limited.

### 3.5 Strengths and weaknesses of the project

This section indicates the main strengths of the project carried out, as well as the main weaknesses that will be addressed for future activities of a similar nature.

#### **Strengths:**

- The activity carried out was of great interest to the participating students for two reasons: 1) participants who were somewhat familiar with Data Science had the opportunity to work on real problems of some interest and complexity, and 2) the participants belonging to the disciplines to which the problems posed were related, have learned Data Science techniques to work with these data, analyse them and obtain models from them.
- The project team was made up of people from different disciplines, making it easier to work in a multidisciplinary activity.
- The students in the activity had extra motivation thanks to the competition for teaching purposes that was carried out and which encouraged them to study the problem in greater depth to find the best possible solution.

#### **Weaknesses:**

- Difficulty of coordination and group work due to the fact that it was carried out via videoconference rather than face-to-face due to the COVID-19 pandemic.
- Difficulty in establishing the initial level of knowledge, due to the fact that the participants belonged to different disciplines.

## 4 Conclusions

Having analysed the results obtained from the forms, the qualitative analysis and the teachers' perception, we believe that experiences such as the one promoted in the innovation teaching project presented here encourages the students' interest in the search for new knowledge. In relation to participation, we consider it was highly satisfactory, not only because of the number of students who decided to take part in a fully voluntary activity that had no direct academic benefit for them, but also because it was complementary to their curricular training.

It is remarkable that using current techniques and tools generates more interest in participation. In this case, we used the hackathon tool, very popular in Data Science, which allowed us to motivate the continuous learning and recycling of knowledge that students must undergo nowadays. The hackathon consisted of working on two different challenges that teams of students tried to solve by competition for teaching purposes. The activity managed to encourage cooperation between people from the same group with different degrees (collaboration was vital to achieve a good result in both challenges), and healthy competition between groups (it was very important to achieve the best solution).

As it is an activity with a certain research connotation, several participants were interested in this novel area, either in the completion of Final Degree Projects, or

in the application for Research Projects derived from the degrees in Life Sciences, where interest in Data Science has increased significantly in recent years.

**Acknowledgements** The teaching innovation project has been funded by the University of Córdoba with reference 2020-2-5003. This work has also been partially subsidised by the “Agencia Española de Investigación (España)” (grant reference: PID2020-115454GB-C22 / AEI / 10.13039 / 501100011033); the “Consejería de Salud y Familia (Junta de Andalucía)” (grant reference: PS-2020-780); and the “Consejería de Transformación Económica, Industria, Conocimiento y Universidades (Junta de Andalucía) y Programa Operativo FEDER 2014-2020” (grant references: UCO-1261651 and PY20\_00074). David Guijo-Rubio’s teaching was funded by the University of Córdoba through grants to Public Universities for the requalification of the Spanish university system from the Ministry of Universities funded by the European Union - NextGenerationEU (Ref. UCOR01MS). The teaching of Víctor M. Vargas was funded by the Programa Predoctoral de Formación al Profesorado Universitario (FPU) of the Ministry of Science, Innovation and Universities (Ref. FPU18/00358). The teaching of Javier Barbero-Gómez was funded by the Programa Predoctoral de Formación de Personal Investigador (FPI) of the Ministry of Science, Innovation and Universities (Ref. PRE2018-085659). The teaching of José V. Die was funded by the H2020-MSCA-IF-2018 programme (Ref. 844431) and the Ramón y Cajal program (Ref. RYC2019-028188-I / AEI / 10.13039/501100011033). The teaching of Pablo González-Moreno was funded by a Juan de la Cierva Incorporación contract (Ref. IJCI-2017-31733) and Plan Propio UCO 2020.

## References

1. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., et al.: Gene ontology: tool for the unification of biology. *Nature genetics* **25**(1), 25–29 (2000)
2. Berardini, T.Z., Mundodi, S., Reiser, L., Huala, E., Garcia-Hernandez, M., Zhang, P., Mueller, L.A., Yoon, J., Doyle, A., Lander, G., et al.: Functional annotation of the arabidopsis genome using controlled vocabularies. *Plant physiology* **135**(2), 745–755 (2004)
3. Brunner, R.J., Kim, E.J.: Teaching data science. *Procedia Computer Science* **80**, 1947–1956 (2016)
4. Bureau, E.: 11.5 mn job openings by 2026, sky-high salaries: Why data science is booming (2020). URL <https://economictimes.indiatimes.com/magazines/panache/11-5-mn-job-openings-by-2026-sky-high-salaries-why-data-science-is-booming/articleshow/74667347.cms>
5. CMA: Work reports of the vegetation mapping of andalusia’s forest mass (2019)
6. Duran, B.S., Odell, P.L.: Cluster analysis: a survey, vol. 100. Springer Science & Business Media (2013)
7. Everitt, B.S., Howell, D.C.: Encyclopedia of Statistics in Behavioral Science—Volume 2. John Wiley & Sons, Ltd (2021)
8. Hicks, S.C., Irizarry, R.A.: A guide to teaching data science. *The American Statistician* **72**(4), 382–391 (2018)
9. Kross, S., Guo, P.J.: Practitioners teaching data science in industry and academia: Expectations, workflows, and challenges. In: 2019 CHI conference on human factors in computing systems, pp. 1–14 (2019)
10. Markowitz, F.: All biology is computational biology. *PLoS biology* **15**(3), e2002050 (2017)
11. Perkel, J.M., et al.: Five reasons why researchers should learn to love the command line. *Nature* **590**(7844), 173–174 (2021)
12. Phyu, T.N.: Survey of classification techniques in data mining. In: International multicference of engineers and computer scientists, vol. 1 (2009)
13. Ramamurthy, B.: A practical and sustainable model for learning and teaching data science. In: 47th ACM Technical Symposium on Computing Science Education, pp. 169–174 (2016)
14. National Academies of Sciences, E., Medicine, et al.: Data science for undergraduates: Opportunities and options. National Academies Press (2018)

# Digital platforms for education. The case of e4you

Javier Parra-Domínguez, Sergio Manzano, Susana Herrero and Pablo Chamoso<sup>1</sup>

**Abstract** At present, there are various developments in Information and Communication Technologies (ICTs) and their importance for the advancement of society. One of the most representative developments has to do with the population's education. Their consonance gives the significance of ICTs with digital transformation, and it is in this sense, that digital platforms emerge as facilitators of education in digital environments. This paper aims to present the e4you digital platform and, through an in-depth study of it, provide insight into the benefits of working in education on this type of platform. It is demonstrated that the platform fulfils its functioning with flexibility and ubiquity, collaborative environments, efficiency and effectiveness of student learning, multifunctionality, motivation and personalisation.

**Keyword** Education; Digital Platforms; Digital Transformation; ICTs

---

Javier Parra-Domínguez,  
BISITE RESEARCH GROUP, edificio I+D+i – calle espejo S/N 37007 Salamanca (Spain)  
e-mail: javierparra@usal.es

Sergio Manzano  
BISITE RESEARCH GROUP, edificio I+D+i – calle espejo S/N 37007 Salamanca (Spain)  
e-mail: smanzano@usal.es

Susana Herrero  
IoT Digital Innovation Hub, edificio PCUVa Módulo 117-118 Pº. de Belén, 9ª,  
470011 Valladolid (Spain) e-mail: susanahc@usal.es

Pablo Chamoso  
BISITE RESEARCH GROUP, edificio I+D+i – calle espejo S/N 37007 Salamanca (Spain)  
e-mail: chamoso@usal.es

## 1 Introduction

The advance in Information and Communication Technologies is undeniable nowadays. The ways of relating in society and training have advanced in a clear and determined way thanks to ICTs [1].

According to the United Nations and according to the data published in the report "Digital 2021. Global Overview Report", the world population at the beginning of the year 2021 was 7.83 million inhabitants, of which some 4.66 billion declared themselves to be Internet users, representing 59.5% of the world's. The report also shows that the majority of the world's population is characterised by the constant use and availability of mobile phones and social networking environments. In addition, it is essential to add that users spend an average of seven hours per year using the internet in terms of the aggregate of their devices. Based on these figures, it can be seen how the field of ICTs and their interaction with human beings the subject of different disciplines is such as communication, philosophy, neuroscience, pedagogy and engineering [2, 3].

It is essential to point out that communication as a favourable element at a pedagogical level in the digital environment is, in the current times, within the 5.0 paradigm [4], where factors such as sensations or emotionality are primordial since this is how users' emotions are identified and categorised using different devices, products and services [5]. In the current era, the development of artificial intelligence in deepening sensory reality based on "one device in one" is also remarkable.

From all of the above, there is no doubt that the path is marked by society's need to communicate and be communicated through the empowerment of digital environments. The education practised in this type of environment and its charge through the promotion of communication is relevant. Many authors make it clear that the digital environment and its development in educational environments will accompany society for a long time [6, 7, 8], especially highlighting accelerators of the process such as the current syndemic consequence of the health crisis imposed by COVID-19. All of the above is refuted by the evidence that most schools and universities were forced to close their physical spaces and classrooms [9].

The advance in the use of ICTs in specific educational environments is not something new; it is a constantly evolving process that, in the last 20 years, has gone from an education based on web 1.0 where the student had little power of participation to a mere receiver of content [10] to participate in web 2.0 and 3.0 education in which students and teachers have an active and dynamic interaction with the use of virtual campuses, forums, blogs, [11] as well as digital platforms such as the case study included in this report [12]. Undoubtedly, the importance of the human factor and social interaction in developing ICT focused on education is evident.

Within the development of training on digital platforms, it is worth highlighting the configuration of these to encompass virtual teaching, online training, distance learning [13] (known as "eLearning"), hybrid blended learning [14] (known as "blended-learning" or "bLearning") and the use of mobile devices for learning that takes place both outside and inside the classroom [15] (known as "Mobile Learning" or "mLearning"). Concerning the three learning methodologies mentioned above, it should be noted that the basis of the training offered in the case study of this report, the e4you digital training platform, is to be found in the eLearning platform [14]:

- The training activity is carried out via an Internet connection ("eLearning")
- It is delivered through ICT. ("eLearning")
- This type of learning promotes learner-centered activities ("eLearning")
- Combination of virtual (asynchronous) and face-to-face (synchronous) ("bLearning", which will be implemented with face-to-face activities such as the Cybersecurity conference)
- Use of mobile devices ("mLearning")
- The benefits of the use of digital platforms for the development of ICTs are as follows:
  - Flexibility and ubiquity. Access to learning is possible at any time and from any place
  - Flexible and collaborative learning environment
  - It improves the effectiveness and efficiency of student learning
  - Multifunctionality. Multiple tasks can be performed in various contexts, interacting with others or creating and sharing content
  - Motivation
  - Personalisation

It should be noted that all of the above is linked to the fact that the success of the initiative in the form of a digital platform is given and determined by the fact that it can deepen the motivation of the student and the teacher and that they are receptive [16] as well as possessing positive and proactive attitudes towards this infrastructure, in addition to having knowledge and a clear command of new technologies [17].

It is important to note that in the digital educational environment, there is still a difference between the way of communicating in this environment and communication established in a context not mediated by digital devices [18]. At this point, it is essential to point out that the effect of showing oneself differently by individual student profiles to make a particular impression on others to achieve, for example, greater recognition is something that is outside the scope of the digital platform so as not to have biases, for instance, in terms of grades.

The work presented here presents the functioning of the e4you digital platform as a success story and the connotations that must be considered when implementing this system to comply with the precepts set out in this introduction. In addition to the introductory section, the development of the work will include a sub-section in the introduction itself where the platform will be described, and then

move on to a second section where the relevant milestones of interaction on the platform will be established. The third section will introduce the main results of educational interaction and will end with the fourth section of conclusions.

## 2 Materials and Methods

### 2.1 The e4you platform

The e4you platform was born as a project of the BISITE Research Group<sup>2</sup> of the University of Salamanca, the General Foundation of the University of Salamanca<sup>3</sup>, the University of Salamanca<sup>4</sup> itself and the IoT Digital Innovation Hub<sup>5</sup>. All of them, as partners and/or collaborators, develop this platform that is configured as an educational tool open to all where the primary motivation is the promotion and development, as well as research on the application of disruptive technologies such as Blockchain and its combination with others such as Artificial Intelligence to provide solutions to the challenges of the present and future education.

As shown in figure 1, the project facilitates educational processes through technology, particularly those related to personalisation or specific attention to the training needs of each person.



**Fig. 1** The figure is e4you platform homepage

When creating MOOCs with the e4you platform, there are three main verticals:

- Universities and other educational institutions - The focus is on solutions for universities or higher education institutions, but also perfect for schools, training centres or academies.

<sup>2</sup> <https://bisite.usal.es/en>

<sup>3</sup> <https://fundacion.usal.es/es/?jjj=1654077325281>

<sup>4</sup> <https://usal.es/>

<sup>5</sup> <https://innovationhub.es/>

- Enterprises - Where the focus is on the provision of specific skills or certified pathways to boost the capacity of teams.
- Governments - Where citizens are offered flexible training experiences tailored to people's needs.

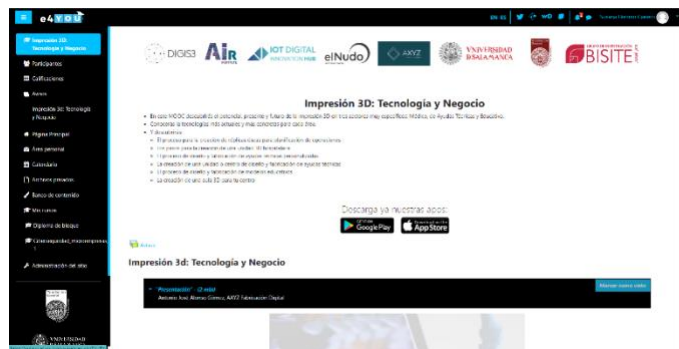
### 3 Teacher-student interaction. The environment

In this section, we introduce the content layout of the e4you digital platform as well as the training dynamics that allow the platform to be aligned with the precepts in the form of benefits of this type of platform; as we saw beforehand, these are none other than flexibility and ubiquity, creation of collaborative environments, efficiency and effectiveness of student learning, the existence of multi- functionality, motivation and personalisation.

Next, we incorporate figures that establish, firstly, the interaction of the teacher with the platform and, secondly, the student's interaction with the platform.

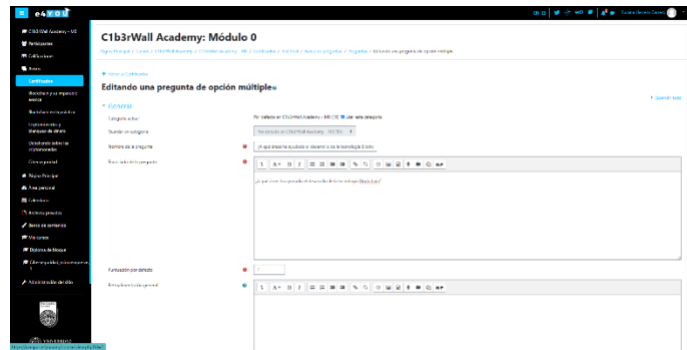
#### 3.1 Teacher-plataform interaction

As shown in figure 2, the teacher can interact with the collaborative space. In addition to this overview, it is essential to highlight the aspects related to the editing action by the teacher (figure 3) and the possible actions to be performed within the MOOC (figure 4) and, in an operational way, the creation of new categories, the management of courses and categories, the aggregation of questionnaires or the creation of a question bank for an exam.

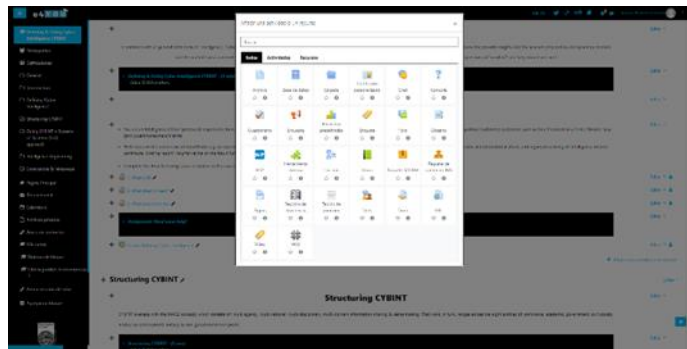


(2)





(3)



(4)

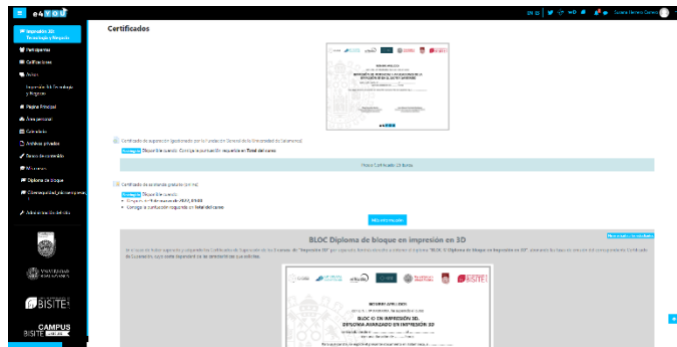
**Fig. 2** The figure is MOOC Interface

**Fig. 3** The figure is editing action by the teacher

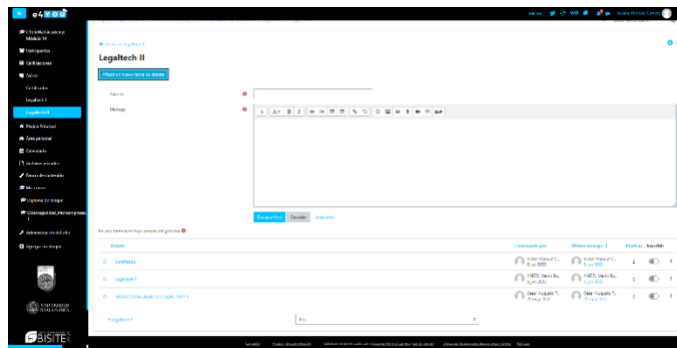
**Fig. 4** The figure is actions to be carried out within the MOOC. Resources

### 3.2 Learner-plataform interaction

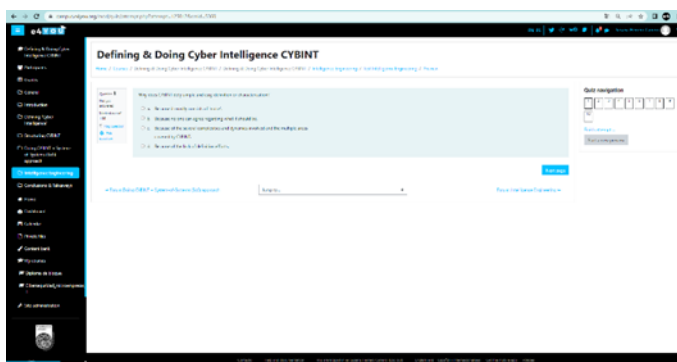
The second part that motivates the benefits that, at least as basic, a digital educational platform should have is related to the interaction between the learner and the platform. In aggregate terms, it is vitally important that the learner feels supported by the preliminary work to the course and the availability of the material at all times. On the other hand, the provision of a forum for participation and exams and the possibility of acquiring a certificate (Figures 5, 6 and 7) are also essential in operational terms.



(5)



(6)



(7)

**Fig. 5** The figure is forum

**Fig. 6** The figure is certificates

**Fig. 7** The figure is sample exam question

## 4 Main results of educacional interacion

As indicated in the introduction and convergence with the development of the e4you project, the primary motivation of digital platforms is the easy access to education and the provision of education in the sense of taking advantage of the necessary teacher-platform-student interactions, considering the new 5.0 paradigm related to the use of new sensory and emotional universes.

With the development of the e4you project, which is presented as a platform in this study, it is possible to meet the main benefits demanded by the educational paradigm that takes advantage of digital platforms.

One of the first benefits is flexibility and ubiquity, which is achieved by allowing alternative navigation at all times. On the other hand, the configuration as a collaborative environment is a maxim that is transferred with the most significant relevance offered to spaces such as the forum or even to interactions attended at all times through social networks. The importance of efficiency and effectiveness in learning is motivated by the concern for the platform's design, which, through the use of white backgrounds, for example, tries to avoid any hint of distraction, reducing the time students spend studying and preventing the emergence of slow time. The configuration as a multifunctional platform is a premise that every digital platform that promotes education must consider since the facilitation of different functions such as interaction with other users, with the teacher, or with other roles is vital to replicate the physical environment far as possible. Motivation, through care with the teacher-platform-student relationship, is evident in the continuous monitoring of processes by platform administrators and personalisation, which is key to making students feel unique and better attend to the lessons offered.

## 5 Conclusions

Today, the advance of ICTs is manifest and socially necessary and indisputable. While the first ten years of the century have generally focused on the advancement of ICTs, from 2010 to 2020, it has been characterised by digital transformation, a process that continues today. This characterisation has enabled the development of the 5.0 paradigm, whose access to emotionality and sensation has, to a large extent, allowed access to education from digital platforms.

In this paper, we present the e4you platform dedicated to training broadly. As can be seen, there are different processes that the platform can address to provide adequate training in digital environments. The experience has allowed and continues to enable us to attend to the benefits that digital media must-have, which are:

- Flexibility and ubiquity. Access to learning is possible at any time and from any place.
- Flexible and collaborative learning environment

- It improves the effectiveness and efficiency of student learning
- Multifunctionality. Multiple tasks can be performed in various contexts, interacting with others or creating and sharing content
- Motivation
- Personalisation
- Acknowledgements

It is essential to highlight that we have also managed to combine what is known as eLearning, bLearning and mLearning, considering the main characteristics without them being mutually exclusive.

## 6 References

1. Fernández-Portillo, A., Almodóvar-González, M., y Hernández-Mogollón, R. (2020). Impact of ICT development on economic growth. A study of OECD European union countries. *Technology in Society*, 63, 101420.
2. Al-Rahmi, W. M., Alzahrani, A. I., Yahaya, N., Alalwan, N., y Kamin, Y. B. (2020). Digital communication: Information and communication technology (ICT) usage for education sustainability. *Sustainability*, 12(12), 5052.
3. Ramirez, G. M., Collazos, C. A., y Moreira, F. (2018). All-Learning: The state of the art of the models and the methodologies educational with ICT. *Telematics and Informatics*, 35(4), 944-953.
4. Roblek, V., Meško, M., Bach, M. P., Thorpe, O., y Šprajc, P. (2020). The interaction between internet, sustainable development, and emergence of society 5.0. *Data*, 5(3), 80.
5. Tekdal, M., Saygıner, Ş., y Baz, F. Ç. (2018). Developments of web technologies and their reflections to education: A comparative study. *Journal of Educational and Instructional Studies in the World*, 8(1), 17-27.
6. Gourlay, L. (2021). There is no'virtual learning': The materiality of digital education. *Journal of New Approaches in Educational Research*, 10(1), 57-66.
7. Emejulu, A., y McGregor, C. (2019). Towards a radical digital citizenship in digital education. *Critical Studies in Education*, 60(1), 131-147.
8. Soroka, V. (2019). Digital Education in the International Pedagogical Discourse. *Comparative Professional Pedagogy*, 9(4), 74-81.
9. Tadesse, S., y Muluye, W. (2020). The impact of COVID-19 pandemic on education system in developing countries: a review. *Open Journal of Social Sciences*, 8(10), 159-170.
10. Huk, T. (2021). From education 1.0 to Education 4.0-Challenges for the Contemporary School. *The New Educational Review*, 66(4), 36-46.
11. Ohei, K. N., y Brink, R. (2019). Web 3.0 and web 2.0 technologies in higher educational institute: Methodological concept towards a framework development for adoption. *International journal for Infonomics (IJ)*, 12(1), 1841-1853.
12. Williamson, B. (2021). Making markets through digital platforms: Pearson, edu- business, and the (e) valuation of higher education. *Critical Studies in Education*, 62(1), 50-66.
13. de Souza Rodrigues, M. A., Chimenti, P., y Nogueira, A. R. R. (2021). An exploration of eLearning adoption in the educational ecosystem. *Education and Information Technologies*, 26(1), 585-615.
14. Sáiz Manzanares, M. C., García Osorio, C. I., y Díez Pastor, J. F. (2019). Differential efficacy of the resources used in B-learning environments. *Psicothema*.
15. Al-Emran, M., Mezhuyev, V., y Kamaludin, A. (2018). Technology Acceptance Model in M-learning context: A systematic review. *Computers & Education*, 125, 389-412.
16. Decuypere, M., Grimaldi, E., y Landri, P. (2021). Introduction: Critical studies of digital education platforms. *Critical Studies in Education*, 62(1), 1-16.

17. Vinogradova, M. V., Kulyamina, O. S., Larionova, A. A., Maloletko, A. N., y Kaurova, O. V. (2016). Digital technology in the field of educational services. *International Review of management and Marketing*, 6(2S), 281-287.
18. Blau, I., y Eshet-Alkalai, Y. (2017). The ethical dissonance in digital and non- digital learning environments: Does technology promotes cheating among middle school students? *Computers in Human Behavior*, 73, 629-637.
19. Kaplan, A. y Haenlein, M. (2016). Higher education and the digital revolution: About MOOCs, SPOCs, social media, and the Cookie Monster. *Business Horizons*, 59(4), 441-450.

# 3D Virtual Laboratory for Control Engineering using Blended Learning Methodology

Francisco Zayas-Gato, Álvaro Michelena, Esteban Jove, José-Luis Casteleiro-Roca, Héctor Quintián, Elena Arce and José Luis Calvo-Rolle

**Abstract** The combination of face-to-face experiences with online technology (Blended Learning) represents an effective approach in today's educational context. The Blended Learning methodology offers the possibility of maintaining face-to-face teaching, reducing the number of students in classroom taking advantage of both modalities. Thus, in terms of laboratory practice, a realistic 3D virtual lab experience as a complementary method to a physical lab can constitute an useful solution to overcome these challenges. The main objective of this proposal is the emulation of a physical level control plant from the laboratories of the Polytechnic School of Engineering of Ferrol (University of A Coruña). In this case, three software tools are combined to build a virtual laboratory, as an alternative to the real one. This modern simulation environment provides students an online working tool, suitable for applying control engineering concepts through a novel approach.

## 1 Introduction

The Blended Learning (BL) methodology, also termed as hybrid or mixed learning [23] involves the online-offline teaching-learning combination [9]. One of the advantages of this methodology is the flexibility provided to students in terms of time planning. Furthermore, BL fosters personalized training, meeting the needs of students learning [22]. The methodology provides students with an easy way to develop their work, focusing on the information and skills needed. BL also implies an interesting collaborative environment for students and instructors. However, the dependence on Information Technologies (IT) represents a disadvantage of this

---

Francisco Zayas-Gato, Álvaro Michelena, Esteban Jove, José-Luis Casteleiro-Roca, Héctor Quintián, Elena Arce and José Luis Calvo-Rolle  
University of A Coruña, CTC, CITIC, Department of Industrial Engineering, Ferrol, A Coruña, Spain; e-mail: {f.zayas.gato, alvaro.michelena, esteban.jove, jose.luis.casteleiro, hector.quintian, elena.arce, jlcalvo}@udc.es

method. The lack of knowledge on technology may result in a waste of resources [21, 14, 16].

On the other hand, laboratory practice in engineering education complements theory lessons in most subjects. This approach promotes collaborative and project-based learning [12]. In this sense, scientific experiments provide students valuable experience in several techniques and helps to develop important skills for their future career [25]. Generally, laboratories are equipped with sophisticated equipment [8, 11, 10]. However, in some cases, students must organize themselves into working groups in order to practice with the available material. Thus, an interesting alternative to real labs is the design and implementation of virtual labs [7].

3D models can bring users an immersive experience close to conventional real experiments [24, 18, 17]. In this sense, the simulation of laboratory experiments represents an attractive alternative [15]. Simulations are not only feasible involving a large number of simultaneous users but also can be implemented with minimal resources.

Contributions like [13] propose the development of schematic diagram-based 2D virtual hydraulic circuits and 3D virtual hydraulic equipment. In [19], an experience supported by a virtual turning machine is proposed as a platform to improve students' understanding of machine behaviour and increase their competence level in this topic. Other works like [5], use CyclePad as a virtual laboratory for designing and analyzing thermodynamics cycles in several educational contexts.

Latest technological advances in terms of computing and digitalization simplify the implementation of virtual scenarios [25, 20] and also help to create BL experiences [21]. In this sense, this work proposes the combination of three modern software tools: Factory I/O, KEPServerEX<sup>®</sup> and Matlab<sup>®</sup>/Simulink, for creating a virtual plant similar to a physical one from the Laboratories of the Politechnic School of Engineering (EPEF), at the University of A Coruña (UDC). Then, this scenario is proposed as a platform for a BL experience involving Control Engineering students.

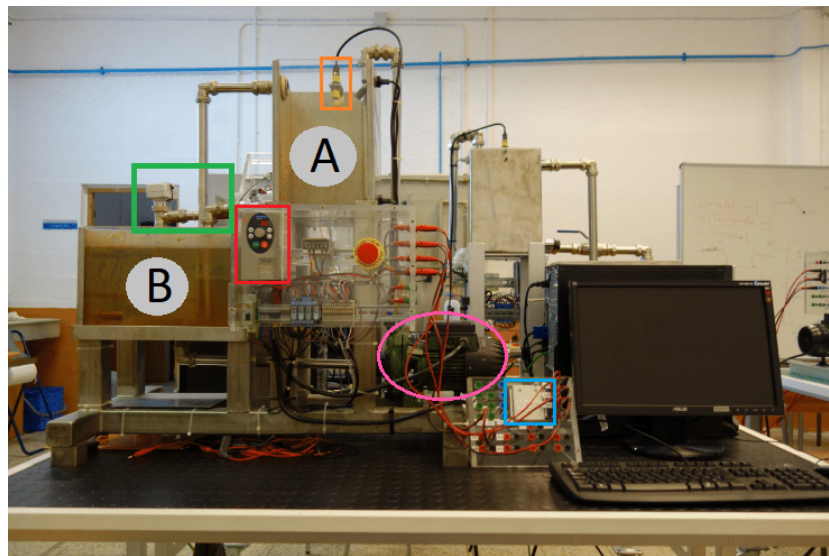
The paper is structured as follows: after the present introduction, the materials and methods of this work are described. Section 3 exposes the experiments and results. Finally, the conclusions and future works are listed.

## 2 Materials and Methods

### 2.1 Background

The EPEF laboratories host several real industrial scale plants emulating industry processes such as speed, pitch or level control, among others. These platforms are mainly used in control and automation subjects. Thus, they are useful for teaching several disciplines within Electronics and Automation Engineering degrees context. In this sense, students can benefit from these plants to support them in their future careers.

One of the laboratory plants is the level control plant. The most common experiment performed with this plant is the liquid level control in one of its tanks. As shown in Figure 1, the level plant is built from two tanks. Tank A in which water level is maintained and tank B that feeds tank A. Tank A water level is measured through an ultrasonic sensor (orange rectangle) installed on top of the tank. To protect the pump and to prevent tank B from running out of water, tank B tank is slightly larger than tank A. A pump is used to fill tank A (pink oval). The speed of the pump is controlled by a Variable Frequency Drive (VFD, red rectangle). The upper tank can be emptied in three ways. Through the upper pipe, if liquid level is equal to or higher than tank's capacity, by using manual or solenoid valve (green rectangle) or by the pump itself.



**Fig. 1** Real Level Control Plant

On the other hand, a control system is implemented on a desktop PC to develop level control experiments. These control experiments based on Proportional Integral Derivative (PID) controllers are performed using Matlab<sup>®</sup>/Simulink software. In this sense, the signal exchange between plant and control system is done through a data acquisition card (blue rectangle). The exchanged signals are:

- Ultrasonic sensor measured level: plant to control system.
- Level Set Point: control system to plant.
- Control Signal sent to VFD: control system to plant.
- Control Signal to solenoid drain valve: control system to plant.

It should be noted that the number of plants is limited. In addition, it should be taken into account that some students follow the master's degree online and cannot



attend the laboratory physically. Hence, it is necessary an alternative to these real plants to propose a BL approach.

## 2.2 Virtual Plant with Factory I/O

An interesting solution to meet the needs of both teaching modes (face-to-face and online) is the use of two level control plants. On the one hand the real plant and on the other hand a virtual plant that emulates the behavior of the real one. For this purpose, the software Factory I/O represents a very useful resource.

Factory I/O is capable to emulate large-scale industrial applications to be used at the classroom [1]. In this sense, students can take advantage from a vast library of industrial parts to configure their own training scenario. Hence, users can turn their computer into a PLC training kit avoiding the risk of damaging any physical equipment. Moreover, it works with almost any Programmable Logic Controller (PLC) equipped with industrial communication protocols such as Modbus or OPC, among others. Not only PLCs but any software application loaded with such communication drivers, can be used as the main platform to control Factory I/O scenes.

## 2.3 OPC Server with KEPServerEX

KEPServerEX<sup>®</sup> is an OPC UA server based on OPC technology for connecting multiple devices and applications, from plant control systems to business management systems [2]. KEPServerEX<sup>®</sup> has been designed to quickly and easily establish communication with almost any device, regardless of the driver used. In addition, it centralizes all the acquired data in a single application and serves all the information from the same access point, minimizing the effort required to work with it.

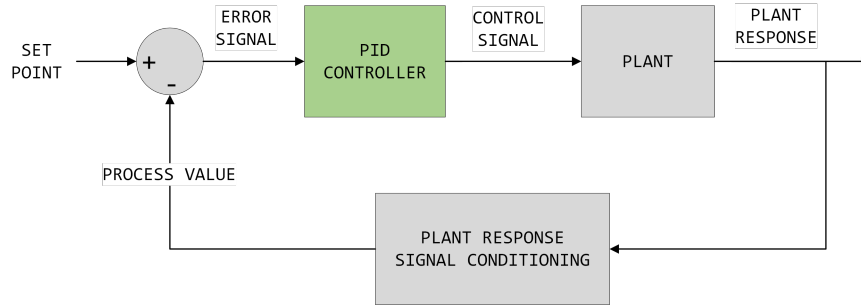
In this case, an effective approach can be applied by using KEPServerEX<sup>®</sup> as a data exchange gateway between Factory I/O and another software tool. This provides flexibility for choosing the best control tool for the scenes depending on the degree's subject being taught.

## 2.4 PID Controller with Matlab/Simulink

Matlab<sup>®</sup> is a well known programming platform for numerical computing widely used in science and industry [3]. Some of Matlab<sup>®</sup> features include data analytics, algorithm and application development, or hardware connection, among others. In this sense, it provides toolboxes with proven functionality as well as an independent module called Simulink for design and simulation.

Simulink is a block diagram programming environment commonly used to create, simulate and deploy multi-domain systems [4]. More specifically, it also provides control engineering tools, from plant modelling to control algorithm design and tuning. As with the real level control plant, Simulink is proposed as the main platform to design and execute a closed-loop simulation based on a PID controller.

The fundamental PID control law is still present on most continuous process controllers [26]. Furthermore, The PID algorithm has been employed in many contexts, achieving very good results due to its simplicity [6]. Figure 2, shows the basic scheme of a closed-loop control system with PID control. It has three main blocks: PID, plant and plant response signal conditioning. Each of them with its own input and output signals. The set point represents the desired water level for tank A and the process value is the water level measured with the ultrasonic sensor. The error signal is the subtraction of the process value and the set point, and then sent to the PID input. Finally, the control signal is computed by the PID controller and sent to the plant. The plant response signal conditioning block receives the plant response and outputs the process value back to the summing junction.



**Fig. 2** Closed-loop control system scheme

On the other hand, given this configuration, the control law from a mathematical point of view can be described through Equation 1. Where:

$$u(t) = K \left[ e(t) + \frac{1}{T_i} \cdot \int_0^t e(t) dt + T_d \cdot \frac{d}{dt} e(t) \right] \quad (1)$$

- $u(t)$ : control signal (PID output).
- $e(t)$ : error signal (PID input).
- $K$ : proportional gain.
- $T_i$ : integral time.
- $T_d$ : derivative time.

In this case, the closed-loop configuration, involving the PID controller, can be implemented by using specific Simulink blocks. Therefore, this is one of the proposed goals for this BL experience.

## 2.5 Proposal for students

An experimental practice is proposed for Industrial Engineering master's degree students of the University of A Coruña. This practice is planned for groups of two or three students, so active participation is ensured. In addition, students within the group assume different roles depending on the practice stage in which they are involved.

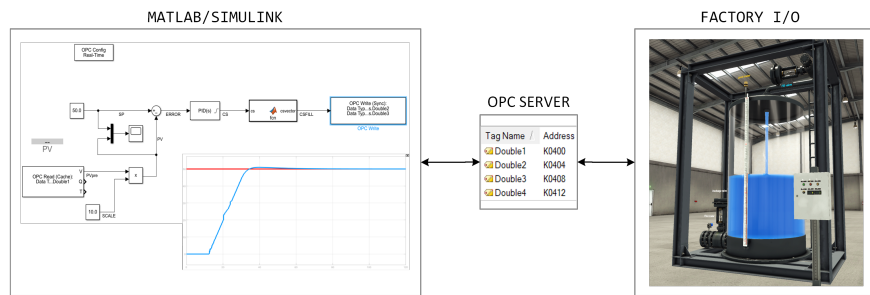
The main goals of this practice are the PID tuning in a closed-loop configuration applying different strategies and a comparative analysis of the obtained results. More specifically, the minimum requirements to successfully complete the practice are:

- Virtual scenario creation with Factory I/O through the existing level control scene, modifying the required parameters.
- OPC server configuration and deployment.
- Closed-loop control system design using Simulink.
- Interface configuration between applications.
- Empirical tuning of the PID controller.
- Proposal of different PID tuning parameters.
- Test the control system applying three different set point levels.
- Comparative analysis of the specifications: peak time, settling time, response time and overshoot time.
- Justification report of practice's results.

## 3 Experiments and Results

### 3.1 Virtual Lab implementation

The process of building the virtual laboratory by interconnecting the proposed applications can be summarized through Figure 3.



**Fig. 3** Virtual lab implementation through app interconnection

On the left side, it can be noticed the Simulink block diagram implementing the main closed-loop blocks: PID controller, plant response signal conditioning and error signal management. In addition, the necessary OPC communication blocks (configuration, read and write) for the OPC server connection with Simulink are incorporated. Moreover, a real time scope for plant's time response visualization is integrated.

The central part illustrates the data exchange carried out by the OPC Server. Once configured, the OPC Server is completely transparent to the user, allowing students to focus on the main goals of the practice.

The right side, shows the virtual plant built with Factory I/O, emulating the real level control plant's behaviour. In this sense, students can benefit from the realistic 3D experience with real sound emulating the filling and discharging of the tank. This scene also includes some manual controls to force the filling and discharging valves, as well as monitoring the current liquid level measured by the virtual instrument.

### 3.2 Surveys

After the report submission deadline, a feedback survey (Table 1) was provided to the students.

ID	Statement
Q1	How likely would you recommend the teacher to continue proposing the Factory I/O and Matlab®/Simulink methodology to a friend or colleague?
Q2	In which aspects should practices with Factory I/O and Matlab®/Simulink be improved?
Q3	If you feel the project instructions were insufficient, how would you improve them?
Q4	If you consider that this work methodology is not suitable for the Control Systems Technology subject, which method do you think is better?

**Table 1** Feedback survey for the virtual lab experience

In this case, twenty students participated in this survey by expressing their opinion. Regarding Q1, students were classified according to their answers as promoters, passives and detractors. It should be noted that the survey has achieved good results (Table 2).

Q1 Results	
Promoters	11
Passives	11
Detractors	6

**Table 2** Q1 Results

On the other hand, the following shows some examples for Q2 answers:

- *"I found the classes quite good, maybe they could go a little deeper into Factory I/O's environment but I have no objection with the teaching methodology".*
- *"I would appreciate further explanation on how KEPServerEX<sup>®</sup> works and the features the other software provide, rather than following a guide to configure them".*
- *"It would be better to teach Simulink contents from the beginning of the academic program. There are students who have never worked with Simulink, thus, everyone could do it by himself as subject's intensity increases instead of working alone from scratch".*

Regarding Q3, listed below are some of students' answers:

- *"There are some aspects left to student's choice, perhaps it would be better to clarify them to avoid confusion".*
- *"I think it would be helpful to provide a short video to understand how the control system blocks interact with each other before getting into practice".*

Finally, some of Q4 answers are shown below:

- *"It is a good methodology, it makes it interesting in comparison to other subjects".*
- *"More face-to-face workload, since some people are more disinterested to learn concepts following the online mode".*

## Conclusions and Future Works

In this paper, a BL experience based on a realistic 3D industrial scenario is proposed. In this sense, the integration of three modern software tools has achieved good results and stands as a powerful tool to overcome the lack of physical laboratory plants. Moreover, it allows to combine face-to-face and online teaching, improving students' experience.

Future works could propose different virtual plants such as production line automation or speed control with virtual motors. In addition, this methodology can be applied in other subjects like Embedded Systems or Industrial Automation, among others.

Finally, the survey model can be applied in future academic programs to perform a comparative or statistical analysis of the results. Furthermore, surveys could be improved and/or modified considering obtained results and expected performance.

## References

1. Factory i/o: Next-gen plc training. 3d factory simulation website (2022). URL <https://factoryio.com/>. Online, accessed June 06, 2022

2. Kepserverex<sup>®</sup> opc ua server website (2022). URL <https://www.kepserverexopc.com/kepware-kepsserverex-features/>. Online, accessed June 06, 2022
3. Matlab<sup>®</sup> website (2022). URL <https://es.mathworks.com/products/matlab.html>. Online, accessed June 06, 2022
4. Simulink<sup>®</sup> website (2022). URL <https://es.mathworks.com/products/simulink.html>. Online, accessed June 07, 2022
5. Baher, J.: Articulate virtual labs in thermodynamics education: A multiple case study. *Journal of Engineering Education* **88**(4), 429–434 (1999)
6. Bahón, C.A., Giner, C.R.: *Tecnología de sistemas de control*, vol. 163. Univ. Politèc. de Catalunya (2004)
7. Balamuralithara, B., Woods, P.C.: Virtual laboratories in engineering education: The simulation lab and remote lab. *Computer Applications in Engineering Education* **17**(1), 108–118 (2009)
8. Casado-Vara, R., Sittón-Candanedo, I., De la Prieta, F., Rodríguez, S., Calvo-Rolle, J.L., Venayagamoorthy, G.K., Vega, P., Prieto, J.: Edge computing and adaptive fault-tolerant tracking control algorithm for smart buildings: A case study. *Cybernetics and Systems* **51**(7), 685–697 (2020)
9. Deschacht, N., Goeman, K.: The effect of blended learning on course persistence and performance of adult learners: A difference-in-differences analysis. *Computers & Education* **87**, 83–89 (2015)
10. Fernandez-Serantes, L.A., Casteleiro-Roca, J.L., Berger, H., Calvo-Rolle, J.L.: Hybrid intelligent system for a synchronous rectifier converter control and soft switching ensurement. *Engineering Science and Technology, an International Journal* p. 101189 (2022)
11. Fernandez-Serantes, L.A., Casteleiro-Roca, J.L., Calvo-Rolle, J.L.: Hybrid intelligent system for a half-bridge converter control and soft switching ensurement. *Revista Iberoamericana de Automática e Informática industrial* (2022). DOI <https://doi.org/10.4995/riai.2022.16656>
12. Ferreira, V.G., Canedo, E.D.: Design sprint in classroom: exploring new active learning tools for project-based learning approach. *Journal of Ambient Intelligence and Humanized Computing* **11**(3), 1191–1212 (2020)
13. Gao, Z., Liu, S., Ji, M., Liang, L.: Virtual hydraulic experiments in courseware: 2d virtual circuits and 3d virtual equipments. *Computer Applications in Engineering Education* **19**(2), 315–326 (2011)
14. García-Ordás, M.T., Alaiz-Moretón, H., Casteleiro-Roca, J.L., Jove, E., Benítez-Andrades, J.A., García-Rodríguez, I., Quintián, H., Calvo-Rolle, J.L.: Clustering techniques selection for a hybrid regression model: A case study based on a solar thermal system. *Cybernetics and Systems* pp. 1–20 (2022)
15. Gonzalez-Cava, J.M., Arnay, R., Mendez-Perez, J.A., León, A., Martín, M., Reboso, J.A., Jove-Perez, E., Calvo-Rolle, J.L.: Machine learning techniques for computer-based decision systems in the operating theatre: application to analgesia delivery. *Logic Journal of the IGPL* **29**(2), 236–250 (2021)
16. Jove, E., Casteleiro-Roca, J.L., Casado-Vara, R., Quintián, H., Pérez, J.A.M., Mohamad, M.S., Luis Calvo-Rolle, J.: Comparative study of one-class based anomaly detection techniques for a bicomponent mixing machine monitoring. *Cybernetics and Systems* **51**(7), 649–667 (2020)
17. Jove, E., Casteleiro-Roca, J.L., Quintián, H., Méndez-Pérez, J.A., Calvo-Rolle, J.L.: A new method for anomaly detection based on non-convex boundaries with random two-dimensional projections. *Information Fusion* **65**, 50–57 (2021)
18. Jove, E., Gonzalez-Cava, J.M., Casteleiro-Roca, J.L., Quintián, H., Méndez Pérez, J.A., Vega Vega, R., Zayas-Gato, F., de Cos Juez, F.J., León, A., Martín, M., et al.: Hybrid intelligent model to predict the remifentanyl infusion rate in patients under general anesthesia. *Logic Journal of the IGPL* **29**(2), 193–206 (2021)
19. Koh, C., Tan, H.S., Tan, K.C., Fang, L., Fong, F.M., Kan, D., Lye, S.L., Wee, M.L.: Investigating the effect of 3d simulation based learning on the motivation and performance of engineering students. *Journal of engineering education* **99**(3), 237–251 (2010)
20. Leira, A., Jove, E., Gonzalez-Cava, J.M., Casteleiro-Roca, J.L., Quintián, H., Zayas-Gato, F., Álvarez, S.T., Simić, S., Méndez-Pérez, J.A., Luis Calvo-Rolle, J.: One-class-based intelligent

- classifier for detecting anomalous situations during the anesthetic process. *Logic Journal of the IGPL* **30**(2), 326–341 (2022)
21. Míguez-Álvarez, C., Crespo, B., Arce, E., Cuevas, M., Regueiro, A.: Blending learning as an approach in teaching sustainability. *Interactive Learning Environments* pp. 1–16 (2020)
  22. Porter, W.W., Graham, C.R., Spring, K.A., Welch, K.R.: Blended learning in higher education: Institutional adoption and implementation. *Computers & Education* **75**, 185–195 (2014)
  23. Tayebinik, M., Puteh, M.: Blended learning or e-learning? Tayebinik, M., & Puteh, M.(2012). *Blended learning or E-learning* pp. 103–110 (2013)
  24. Valdez, M.T., Ferreira, C.M., Barbosa, F.M.: 3d virtual laboratory for teaching circuit theory—a virtual learning environment (vle). In: 2016 51st International Universities Power Engineering Conference (UPEC), pp. 1–4. IEEE (2016)
  25. Vasiliadou, R.: Virtual laboratories during coronavirus (covid-19) pandemic. *Biochemistry and Molecular Biology Education* **48**(5), 482–483 (2020)
  26. Zayas-Gato, F., Quintián, H., Jove, E., Casteleiro-Roca, J.L., Calvo-Rolle, J.L.: Diseño de controladores PID. Universidade da Coruña, Servizo de Publicacións (2020)