وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR ANNABA-UNIVERSITY
Faculty of Technology
Department of Computer Science

جامعة باجي مختار – عنابة
كلية التكنولوجيا
قسم الاعلام الألي

Domain: Mathematics and Computer Science

Branch: Computer Science

Specialty: Embedded Computing

Testing Collective Cyber-Physical Systems and Embedded Multi-Agent Systems

A Dissertation Submitted to the Department of Computer Science in Partial Fulfillment of the Requirements for the LMD Doctorate Degree in Computer Science.

Submitted by
Oudina Zina

Supervised by
Pr. Derdour Makhlouf

Board of Examiners

Quality	Name	University Degree	Affilitation
Chairman	Ghanemi Salim	Professor	University of Badji Mokhtar Annaba
Supervisor	Derdour Makhlouf	Professor	University of Oum El Bouaghi
Examiner	Khtatba Morad	MCA	University of Badji Mokhtar Annaba
Examiner	Benmounah Zakaria	MCA	University of Constantine 2
Examiner	Sahraoui Abdellatif	MCA	University of Echahid Cheikh Larbi Tebessi - Tebessa

Academic Year: 2023/2024

Acknowledgements

"Praise be to God and thanks be to God"

I would like to thank Pr. Rachid Boudour, who proposed this subject and supported me during many years. I particularly thank him for the competence, availability, critical outlook, and confidence with which he guided me throughout this work.

I would also like to thank Pr. Derdour Makhlouf, professor at the University of Oum El Bouaghi, who supervised and directed this thesis, for his patience, motivation, and guidance over the past four years.

My profound and grateful thanks to Professor M.S, who is the reason for every success in my life.

My beloved sister Samira, brother Khaled, and Abdel Rezzaq, you have all of my love and gratitude for your unwavering support for me. Thanks also to my sister's husband, Abdel Hamid.

Without forgetting: my dear nieces, Rayan and Kawthar. My dear nephew, Taha. My beloved niece, Mayasin, and my beloved nephew, Mohammed Talid (BEZA), my beloved Soujoud .

My Dear mother and father, you have all my love, thanks, and appreciation for your great support and patience. With gratitude and tremendous love, I dedicate this work to you.

Zina Oudina , zina.oudina@univ-annaba.org .

Abstract

Designing a high-confidence cyber-physical system (CPS) is required to guarantee the effectiveness and security of the system. The variability of the system makes the engineering requirements for CPS complex. Additionally, it is necessary to test CPS robustness because system errors and malfunctions can occur. The verification determines whether a system meets a set of requirements. Such cyber-physical systems are used in various essential fields, including air traffic, transportation, energy, and oil and gas. Verification is critical for developing secure systems with high assurance levels. Defining a test strategy is critical to reducing costs and maximizing the efficiency of system development and manufacturing organizations.

This study highlights how CPS, which is a complex and heterogeneous systems, is well modeled with the most recommended multifaceted approach for the design of security and confidence that is the model based system engineering (MBSE). The use of artificial intelligence with a combination of modern modeling tools was provided. A multifaceted test approach is the result of this merging. An embedded multi-agent system is used with verification technique-based rules to design a test model for trust CPS. To improve the efficiency and assurance of the proposed TEST EMAS model, we also provided a reputation test for each agent test within our test model, which enables testing trust in an embedded multi agent system.

This study achieved the intended theoretical and practical targets and proposed a models and approach for modeling and analyzing trust cyber-physical systems, with an emphasis on verification. It distills modern conception and design with emerging AI approaches, such as embedded multi-agent systems, resulting in the development of comprehensive scale modeling and verification techniques for trust in cyber-physical systems.

Keywords: Cyber physical system (CPS), Embedded Multi Agent systems (EMAS), Trust, Model Based System Engineering (MBSE), Verification, EMAS TEST.

ملخص

هناك حاجة إلى تصميم نظام فيزيائي سيبراني عالي الثقة لضمان فعالية النظام وأمنه. إن تنوع النظام يجعل الاحتياجات الهندسية معقدة للغاية. بالإضافة إلى ذلك، من الضروري إختبار قوة النظام السيبراني الفزيائي لأنه من الممكن أن تحدث أخطاء في النظام وأعطال. يحدد التحقق ما إذا كان النظام يلبي مجموعة من المتطلبات. تُستخدم هذه الأنظمة الفيزيائية السيبرانية في مجموعة متنوعة من المجالات الأساسية، بما في ذلك الحركة الجوية والنقل والطاقة والنفط والغاز. يعد التحقق أمرًا بالغ الأهمية لتطوير أنظمة آمنة ذات مستويات عالية من الضمان. يعد تحديد استراتيجية الاختبار أمرًا بالغ الأهمية لتقليل التكاليف وزيادة كفاءة تطوير النظام ومؤسسات التصنيع.

تسلط هذه الدراسة الضوء على كيفية تصميم النظام الفيزيائي السيبراني ، وهو نظام معقد وغير متجانس بشكل جيد .باستعمال النهج متعدد الأوجه الموصى به لتصميم الأمان والثقة وهو هندسة النظم القائمة على النموذج و استخدام الذكاء الاصطناعي مع مجموعة من أدوات النمذجة الحديثة. نهج الاختبار متعدد الأوجه هو نتيجة لهذا الدمج اذ يتم استخدام نظام متعدد الوكلاء مدمج مع قواعد تعتمد على تقنية التحقق لتصميم نموذج اختبار للثقة للنظام السيبراني الفيزيائي. من أجل المضي قدمًا نحو تحقيق الكفاءة وضمان اختبار نموذج الاختبار المقترح، قمنا أيضًا بتوفير اختبار سمعة لكل وكيل مسؤول عن اختبار ضمن نموذج الاختبار الخاص بنا والذي يتيح اختبار الثقة في النظام المتعدد الوكلاء المدمج.

حققت هذه الدراسة الأهداف النظرية والعملية المقصودة وتقترح نماذج ونهجًا لنمذجة وتحليل الأنظمة الفيزيائية السيبرانية الموثوقة، مع التركيز على التحقق والاختبار. إنه يبرز المفهوم والتصميم الحديث مع مناهج الذكاء الاصطناعي الناشئة مثل الأنظمة المدمجة متعددة الوكلاء، مما يؤدي إلى تطوير نماذج شاملة وتقنيات التحقق من أجل الثقة في الأنظمة المادية السيبرانية.

الكلمات المفتاحية: النظام الفيزيائي السيبراني ، الأنظمة المدمجة متعددة الوكلاء ، الثقة ، هندسة النظم القائمة على النموذج، الاختبار، نموذج الاختبار المبني على الانظمة المدمجة متعددة الوكلاء.

Résumé

La conception d'un système cyber-physique (CPS) de haute confiance est nécessaire pour garantir l'efficacité et la sécurité du système. La variabilité du système rend les besoins techniques du CPS assez complexes. De plus, il est nécessaire de tester la robustesse du CPS, car des erreurs et des dysfonctionnements du système peuvent survenir. La vérification détermine si un système répond à un ensemble d'exigences. De tels systèmes cyber-physiques sont utilisés dans divers domaines essentiels, notamment le trafic aérien, les transports, l'énergie, ainsi que le pétrole et le gaz. La vérification est essentielle au développement de systèmes sécurisés avec des niveaux d'assurance élevés. La définition d'une stratégie de test est essentielle pour réduire les coûts et maximiser l'efficacité des organisations de développement de systèmes et de fabrication.

Cette étude met en évidence comment le système cyber-physique, ces systèmes complexes et hétérogènes, est bien modélisé avec l'approche multifacette la plus recommandée pour la conception de la sécurité et de la confiance, à savoir l'ingénierie des systèmes basée sur les modèles. L'utilisation de l'intelligence artificielle avec une combinaison d'outils de modélisation modernes est proposée. Une approche de test à multiples facettes est le résultat de cette fusion. Un système multi-agent intégré est utilisé avec des règles basées sur des techniques de vérification pour concevoir un modèle de test pour le CPS de confiance. Pour améliorer l'efficacité et l'assurance du TEST EMAS proposé, nous avons également fourni un test de réputation pour chaque agent de test au sein de notre modèle de test, ce qui permet de tester la confiance dans le système multi-agent intégré.

Cette étude a atteint les objectifs théoriques et pratiques visés et propose des modèles et une approche pour modéliser et analyser la confiance dans les systèmes cyber-physiques, mettant l'accent sur la vérification. Il distille une conception et un design modernes avec des approches d'IA émergentes telles que les systèmes multi-agents embarqué, aboutissant au développement de techniques complètes de modélisation et de vérification à l'échelle pour la confiance dans les systèmes cyber-physiques.

Mots clés : Système Cyber physique (CPS), Systèmes Multi Agents Embarqués (SMAE), Confiance, Ingénierie de système basée sur des modèles (ISBM), Vérification, EMAS TEST.

Contents

List of Figures

List of Tables

Abbreviations

CPS	Cyber-Physical system
CPPS	Cyber-Physical Production System
MAS	Multi- Agent System
EMAS	Embedded Multi Agent system
MBSE	Model Based System Engineering
SE	System Engineering
SysML	Systems Modeling Language
UML	Unified Modeling Language
ICT	Information and Communication Technology
BDI	Belief–desire–Intention Software Model
MMSA	Multimedia Software Architecture
FMI	Financial Market Infrastructures
SSP	System Structure and Parameterization
AoMA	Application-oriented Middleware Architecture
AGR	Model (Agents/ Group /Role)
AGRS	Model (Agents / Group / Role/ Service)
MaDKit	Multi-Agent Development Kit
JADE	Java Agent Development
TDD	Test-Driven Development
SCADA	Supervisory Control and Data Acquisition
ICS	Industrial Control System
PLC	Programmable Logic Controller
RTU	Remote Terminal Unit
HMI	Human-Machine Interface
MDS	Microwave Digital System
BNDC	Bi-simulation-based no Deducibility on Compositions
AMS	Agent Management System
DF	Directory Facilitator
FIPA	Foundation for Intelligent Physical Agents
GSPN	Generalized Stochastic Petri Nets
C3PS	Command and control cyber-physical system.
FNDN	Flying Named Data Networking
DT	Digital Twin
TDD	Test Driven Development
RTA	Real-Time Agent
RMA	Remote monitoring Agent
ACL	Agent Communication Language
UAV	Unmanned Aerial Vehicle
FCB	Flight Control Board
ESC	Electronic Speed Controller
TCU	Transceiver Control Unit
PMS	Power Management System
DSP	Defensive Security Plan

CHAPTER 1

GENERAL INTRODUCTION

"Every science begins as a philosophy and ends as an art."

Will Durant (1981)

1.1 Introduction

Cyber-physical systems have become the backbone of the systems running in the major production and digitization fields that supervise heavy and precision industries, such as the petroleum and electronic industries, diesel industry, smart vehicles, aviation industry, and space industries.

CPS integrates computational and physical operations that are supervised and managed via networks and embedded computers. A CPS connects many pieces of equipment via wireless networks. Communication was established using groups of sensors and actuators.

In conjunction with the same importance and widespread use, embedded multi-agent systems have become an important approach for modeling, simulating, and testing complex intelligent systems. Multi-agent systems are used in various fields, including sociology, video games, animation, and finance, for expertise and trade. Thus, they can imitate social models, enable robotics research, and be tailored to specific problems, such as energy management.

Trust in critical systems, such as CPS and EMAS, may be defined as a quality that meets many engineering requirements, such as safety, security, reliability, and other sub-qualities. These attributes are related to many actors such as systems and environments, stakeholders, and users. Trust may be derived from the concept of reputation, which means that the system is reputed and how others perceive its reputation.

When confidence is neglected, there may be fatalities, long-term environmental effects, damage to vital infrastructure, or other disastrous outcomes, including loss of equipment, exposure to private information, financial loss, and damage to one's reputation. Risks and unfavorable effects increase with the degree of industry networking and integration.

The challenge of novel technology and industry 4.0 triggers the need for computational tools and methods to verify trust in these complex systems and the quality of how those systems deliver services and ensure user satisfaction.

This introductory chapter outlines how the thesis is developed; it introduces the motivation and presents the context that relates to the verification of cyber-physical systems and

embedded multi-agent systems. This chapter also showcases the contribution and addresses the aims and structure of the thesis.

1.2 Motivation

Developing systems with high levels of confidence and security is recommended, with the increase of risk and safety dangers, even in small, large, and complex industries.

Numerous risks and dangers exist for cyber-physical systems that have the potential to disrupt their operations, endanger human safety, harm the environment, or even cause a financial meltdown necessitating the establishment of trustworthiness and security in lifecycle development and integrating verification. Developing a test methodology is essential for reducing expenses and optimizing the productivity of manufacturing and product development companies.

The previous points motivated us to look for a test technique that ensures decentralized verification for each trust property and quality level and throughout the entire cycle life of the CPS. The advantage of this test technique is that it performs verification early in the design process, ensuring that any errors in any step are fixed, and that any design problems are addressed on time. Not waiting until the conclusion of the modeling and development cycle life and then using CPS to find defects and bugs at the first level of conception. A delay in mistake detection leads to system failure, hazard harm, risks, and high costs.

1.3 Problem Statement

The challenge of novel technology and industry 4.0 triggers the need for computational tools and methods to verify trust in complex systems and the quality of how those systems deliver services and ensure user satisfaction.

Significant engineering challenges regarding security and reliability limit the enormous promise of CPS and necessitate the discovery of new solutions [1]. When a CPS is developed with proper dependability, V&V costs are typically reduced, and expensive redesigns are avoided [2].

On the one hand, we identified an issue with a lack of comprehensive analysis of trust quality in CPS, as well as the challenge of modeling trust quality in cyber-physical systems during the design process. On the other hand, there is a dearth of studies in the domain of trust testing, as well as an acceptable approach for testing trustworthiness with all of its qualities within a complex and heterogeneous system, such as CPS. We considered the possibility of unifying these different areas of study by representing them through the following sub-issues:

1.3.1 Statement one: Analysis and evaluation of Trust for CPS

There is no comprehensive taxonomy of trustworthiness and trust measurement types in the literature, nor is there a precise definition of all types of trustworthiness metrics. There are no established criteria for trustworthiness and no guidelines for evaluating CPS reliability.

1.3.2 Statement two: Trust Modeling for CPS

Some methods for representing trust in systems have been proposed in the literature. However, it does not offer suitable instructions for a methodical approach or recognized modeling language to allow trust-based analysis. Establishing trust by design in CPS and methodically integrating trust engineering into system development from the start of the system lifecycle are the two most important difficulties.

1.3.3 Statement three: Testing trust in CPS and Embedded multi agent systems

The authors of [3] stated that while there are many different testing methodologies for cyber-physical systems, more thorough approaches are needed to guarantee trustworthiness in safety-critical scenarios. Testing and validating the accuracy of CPS components is arduous. Little research has been conducted on trust in CPS testing procedures [4]. Numerous approaches for testing CPS exist , such as trust based verification and digital twins, vetting-based verification of standards, and runtime monitoring of sensor streams [5], treat trust from many points of view and targets, but do not address trust as established by design in components. The literature on multi agent systems not widely addressed the trustworthiness. Computational tools and techniques are required to confirm the trustworthiness of multi-agent systems and system components concerning task fulfillment [6].

1.4 Context

The complexity of Cyber-Physical Systems provides unparalleled testing issues in terms of functionality, non-functionality, and quality. In this study, we will test the quality. Consequently As a result, we require basic research to understand the quality that must be tested and how to deal with the challenges that testing entails. Given the breadth and depth of the topic, we seek to present an analysis of trustworthiness in terms of concern, requirements, and attributes by defining the test's target, which is CPS trust quality. The second question is how to model this quality in the CPS during the early design phase. How to verify this quality within the CPS life cycle model using an appropriate and intelligent approach, such as embedded multi-agent systems.

1.5 Objectives

The objectives targeted by this thesis subject are theoretical and practical.

a) For theoretical objectives, the expected results are as follows:

- ✓ Analysis of Trust from the Perspective of System Quality. We define the set of trust requirements for the CPS, as well as the set of trust properties. Provide a quantitative evaluation of trust.
- ✓ Addresses the challenges of trust CPS modeling and provides a suitable model that is a means of designing such complex systems and complex qualities such as trustworthiness by combining many approaches such as systems engineering, requirement engineering, modeling tools, and integrated verification for the all-cycle development of trust CPS.
- ✓ Use Artificial intelligence methods, such as embedded multi-agent systems for testing trust in cyber physical systems and design a test model based on embedded multi-agent systems.
- ✓ Deepening the underlying test approach, taking into account collective functionalities, emerging behaviors, and communication by testing the reputation of agents in the test model.

b) For practical objectives, the expected results are as follows:

- ✓ Integration of the testing approach into the Jade platform.
- ✓ Experimental validation and design of a model for cyber physical systems in the oil and gas industry. Apply the test approach to petroleum CPS (ICS and SCADA).

1.6 Thesis Contribution

This thesis proposes a formalization of analysis, modeling, and verification based on an analysis framework, design model, and test model. We consider the particularities of the CPS as a complex system, and the particularities of the quality (trust) to be tested. Our proposed solution is built on three pillars: the first is to analyze the trust quality of CPS and present a comprehensive set of attributes and requirements within the trust degree framework, as well as classify trust concerns for cyber-physical systems. The second pillar discusses how to model trustworthiness in CPS at the early design phase on the basis of the ranked trust degree in phase one and how verification is implemented in the model. The final step is to verify the trust quality of the CPS with an embedded multi-agent system. For the experimental part of this thesis, we chose CPS in the field of oil and gas. For this topic of petroleum CPS, we contribute model-based system engineering for trust SCADA

and ICS. We address the risk of petroleum CPS and mitigation as well as the trust concern for CPS in the oil and gas industry. Risk assessment and mitigation availability are important for the verification of trust in petroleum CPS and will be incorporated into the verification rules of the test model (EMAS TEST). The contributions of this study are as follows.

1) Part 1: Trust degree framework

We suggest a flexible and well-reasoned framework based on trust ranking to assess trust in cyber-physical systems (CPSs). Three trust degrees were established: degree one refers to a secured CPS, degree two refers to a trustworthy CPS, and degree three refers to a trusted CPS. Subsets pertaining to each degree were created from a defined set of trust attributes and requirements. We suggest categorizing attributes according to commitments, judgments, and system functionality. A simple mathematical formula is suggested to calculate CPS trust quantitatively.

2) Part 2: Trust concern for CPS

We offered an analysis of trust-related issues and detailed the entire set of trust-related requirements and apprehensions that propel the development of the intended trust quality in CPS from the beginning of the design phase. Each fear class consists of a collection of properties that can be associated with any one or all of the following: trustworthy, human, business, or functional. The attribute set definition for each fear type shows how trust-related issues are interrelated. It also demonstrates how recognizing issues could aid in lowering risks.

3) Part 3: Modeling trust CPS

We proposed an effective and practical MBSE method to establish confidence in CPS. The cornerstone of MBSE is system engineering, and the MBSE technique for the trust cyber physical system is built upon a number of disciplines, including requirement definition, modeling, and verification.

4) Part 4: EMAS TEST Model for testing trust in CPS

In our proposed MBSE method, a verification phase is considered and presented step by step (element verification, requirement verification, attribute verification, and quality verification). This testing mechanism is combined with embedded multi-agents, which target the verification of trust quality in CPS. The methodological level, decentralized nature of EMAS, and analysis help to address the complexity of trustworthiness quality. We proposed an EMAS TEST model and detailed the verification steps, agent behavior, and characteristics.

In addition, we proposed algorithms, verification rules, and defined a task of the agent within the EMAS TEST, as well as defined embedded constraints.

5) Part 5: MBSE for trust ICS and SCADA in oil and gas industry

We covered the design principles of a modern ICS and SCADA system for the oil and gas industry, as well as Model-Based Systems Engineering (MBSE) methodology.

6) Part 6: Comprehensive Risk Classification and Mitigation in the Petroleum CPS

We offer a risk classification for the oil and gas industry as a whole. We classified the risks based on the knowledge that the cyber-physical system is the most crucial element of the industry and that the risks are either physical, cyber, or related to permission and authorization for oil and gas companies. The well-known risks to the oil and gas industry's SCADA, ICS, petroleum CPS, and system structure are presented. In addition, a security strategy that mitigates the effects of threats in oil and gas zones is proposed.

Figure 1.1 depicts the contribution phases.

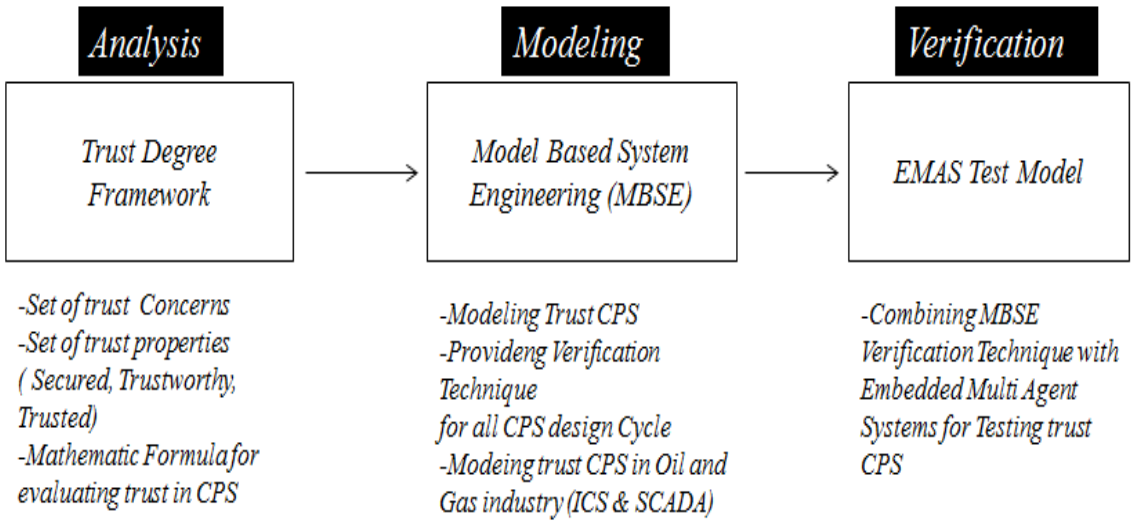

Figure 1.1 Contribution phases

1.7 Publication

The contribution parts are published via journal and conference papers as presented in Table 1.1.

Table 1.1Publication for contribution parts

Publication Type	Paper	Contribution part
Journal	Oudina, Z., Derdour, M., Boudour, R., Dib, A., & Yakoubi, M. A. (2023). Trust cyber physical systems: Trust degree framework and evaluation. Journal homepage: http://iieta. org/journals/ijsse, 13(2), 213-225.	1
Journal	Oudina, Z., Derdour, M., Dib, A., Yaakoubi, M.A. (2024). Identifying and addressing trust concerns in cyber-physical systems for the oil and gas industry. Ingénierie des Systèmes d'Information, Vol. 29, No. 2, pp. 469-478. https://doi.org/10.18280/isi.290208	2
Journal	Oudina, Z., & Derdour, M. (2023). Toward Modeling Trust Cyber-Physical Systems: A Model-based System Engineering Method. International Journal of Advanced Computer Science and Applications, 14(7).	3
Journal	Oudina, Z., Dib, A., Yakoubi, M. A., & Derdour, M. (2024). Comprehensive Risk Classification and Mitigation in the Petroleum Cyber-Physical Systems of the Oil and Gas Industry. International Journal of Safety & Security Engineering, 14(1).	6
Conference	Oudina, Z., Derdour, M., Dib, A., & Tachouche, A. M. A. (2023, October). Model Based System Engineering for trust SCADA and ICS Systems in Oil & Gas Industry. In 2023 5th International Conference on Pattern Analysis and Intelligent Systems (PAIS) (pp. 1-8). IEEE.	5
Conference	Oudina, Z., Derdour, M., & Bouhamed, M. M. (2022, October). Testing cyber-physical production system: Test methods categorization and dataset. In 2022 4th International Conference on Pattern Analysis and Intelligent Systems (PAIS) (pp. 1-8). IEEE.	State of the art
Conference	Z. Oudina, M. Derdour, A. Dib and M. M. Bouhamed, "Empirical Analysis of the Security Threats and Risks that Drones Face, Represent, and Mitigation," 2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS), EL OUED, Algeria, 2024, pp. 1-8, doi: 10.1109/PAIS62114.2024.10541193.	Preparation for future work

1.8 Thesis Structure

This thesis is divided into three parts, as illustrated in Figure 1.2. The study begins with an introductory chapter describing the context and motivation. The theoretical and practical goals are presented. The contribution of this thesis is thorough, and it highlights its publication. Part one contains chapters two and three, and presents the state of the art that relates to the context of the thesis: cyber physical systems and embedded multi agent systems, trust CPS, and trust verification. Part two presents our contribution, which contains chapter four and chapter five.

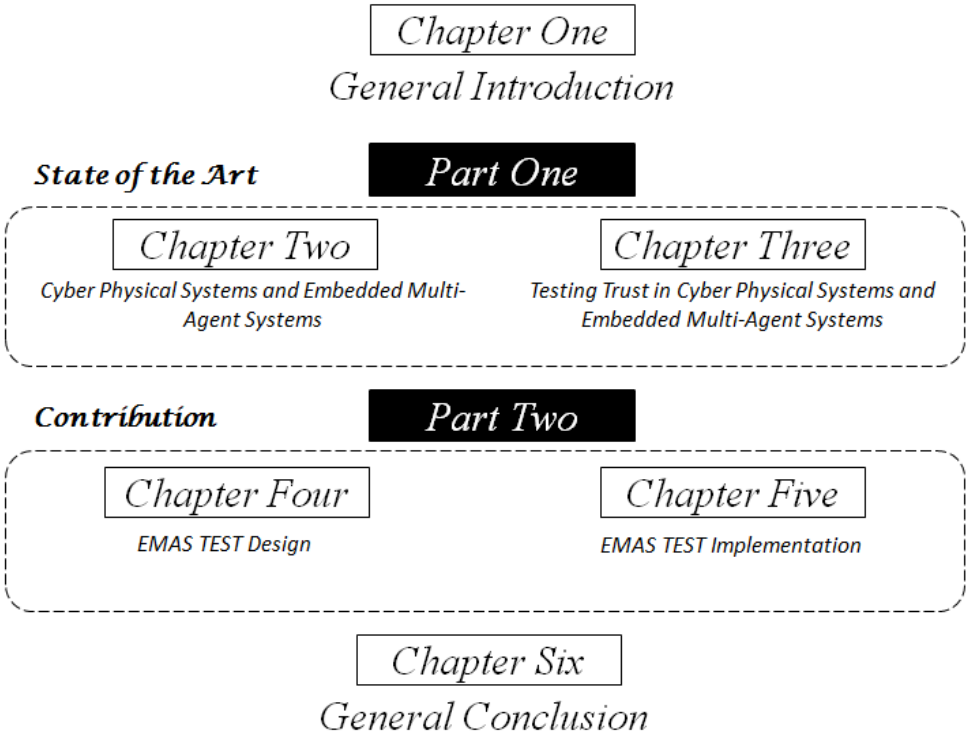

Figure 1.2 Thesis structure

Chapter two defines CPS, its characteristics, and its sphere of use. The embedded multi-agent systems are also defined, along with their properties, kinds, organizational models, and application domain.

Chapter three explores how trust is vital in the establishment of a new CPS and how researchers vary their conceptualization and verification techniques. Challenges in testing CPS are highlighted. Additionally, trust and its verification in embedded multi-agent systems are addressed.

The design of the proposed solution is presented in Chapter 4; a technique for testing trust in CPS includes a study of trust quality in CPS as well as a presentation on modeling trust in CPS. A description of the EMAS TEST model is presented, including the verification methods, agent behavior, and characteristics. In addition to the required algorithms, verification rule definitions, and agent tasks inside EMAS TEST. Chapter five describes the experiments with the proposed models and techniques.

The thesis concludes with a general conclusion that highlights the contributions made and viewpoints that might be pursued further, opening up new areas for future research.

CHAPTER 2

CYBER PHYSICAL SYSTEMS

AND EMBEDDED MULTI-AGENT SYSTEMS

"We are artistic enough that I can draw freely in my imagination. Imagination is more valuable than knowledge. Knowledge is limited. Imagination encircles the world."

Albert Einstein

2.1 Introduction

More than 15 years ago, cyber-physical systems were introduced. The term cyber refers to the use of digital power to benefit the physical environment. These systems rely on board sensors and controls to identify, maneuver, listen to, and communicate with humans. The Cyber-Physical System (CPS) is the most important intelligent system in the new generation of digital systems, consisting of complicated components. What are the definitions of cyber physical systems, as well as their important characteristics and fields of application, modeling these systems, are the subject of this chapter.

We begin with definition and descripion followed by comprehensive definitions. Integration, scalability, cooperation, and other CPS characteristics are also highlighted. The CPS architecture and modeling approach are described.

2.2 Foundations of Cyber-Physical Systems

2.2.1 Definitions and description of Cyber-Physical Systems (CPS)

A Cyber-Physical System (CPS) is the most essential intelligent system in the new generation of digital systems, consisting of complex components and pieces that are not always homogeneous. It has computational and physical capabilities that allow it to interact with humans in ways that have never been seen before. It is regarded as a network of variables comprising of both physical inputs and outputs. The development and application of sophisticated CPS will create novel opportunities for economic growth.

A cyber-physical system necessitates embedded systems as well as a blend of methods, mechatronics, and design. Thus, cyber physical systems are a sophisticated blend of physical and computational aspects that can be found in a variety of industries, such as manufacturing, aerospace, automotive, and transportation. There are numerous definitions of cyber systems, including generic definitions and definitions focused on the proprieties of the system or the industry in which the system is used.

A cyber-physical system (CPS) is a system in which computer elements collaborate to control and command physical entities. CPS is a collection of computer elements that work together to control physical entities. The ultimate goal of CPS is to create smart spaces such as smart transportation, smart health, smart grids, etc. Complex and large-scale CPSs include smart electrical networks. A smart grid is an electricity and information transmission network that allows energy to be collected and distributed in a controlled, reliable, and secure manner, ensuring optimal network functioning and meeting the needs of the stakeholders [7].

The primary function of CPS is to interact with the physical world. This is a distinct feature that distinguishes them, especially when compared to "traditional" computer systems that only perform data processing. CPS has a broader mission. They can control energy flows (electrical networks), material flows (oil wells and pipelines), and signals (air traffic control).

Physical processes in a cyber physical system (CPS) are monitored and controlled by embedded computers and networks, generally through feedback loops in which calculations are affected by physical processes and vice versa [8]. Wireless networks are utilized in CPS to connect disparate pieces of equipment. Sets of sensors and actuators were used to establish communication. CPS has been used in a variety of fields, including aerospace, transportation, essential infrastructure, and industrial manufacturing.

The term "cyber physical systems," or CPS, was coined by Professor Edward Lee of the University of California, Berkeley more than a decade ago. The cyber component refers to their digital power, whereas the physical part refers to their impact on the real world. It is critical to have sensors and actuators to monitor the system and intervene at a physical level [9]. Another significant concept is the use of distributed embedded systems that are no longer isolated from one another but communicate with one another. Finally, the entire system is self-contained with a digital command that constantly feeds information from the sensors to control the actuators. A major breakthrough is in artificial intelligence.

According to the European Commission, this is an autonomous industrial IoT. Valeo's self-driving automobile is an example of CPS in action. However, we can name intelligent SCADA, these industrial control-command systems, if they are linked, autonomous robots, dialysis equipment that operates on its own, or even a connected plane, even if the pilot constantly retains his hand on the controls to change the direction.

2.2.2 Cyber physical System Characteristics

Cyber-physical systems (CPSs) are embedded systems that increase the autonomy, interdependence, connectivity, and collaboration of all of their constituent parts. Permit the enhancement or offer new functionality or a new service.

a) Integration*:* Another essential part of Cyber Physical Systems is the integration of Wireless Sensor Networks with the cloud. The benefits of CPS include fault tolerance, control over network transaction scheduling, middleware and software that facilitate network coordination, media access control techniques, and their effects on system dynamics [10].
b) Flexibility: CPS provides more services than cloud computing and WSN put together.
c) Scalability: CPS, which is a part of cloud computing, has the potential to provide users with resources according to their requirements.
d) Managing Uncertainty: The technique of proving certainty involves offering proof that a design is trustworthy and legitimate. Cyber-physical systems can adapt and function in unsteady environments.
e) Human-system interaction: Situational awareness, or how people perceive a system and its surroundings, is a critical component of decision-making and should be modeled and measured.
f) Performance: CPS can provide better feedback and autonomous redesign owing to the intimate relationship between sensors and cyber infrastructure.
g) Response time: CPS will improve the response time and assist early failure detection, as well as proper resource utilization, such as bandwidth.
h) Cooperation: Electronic components working together to influence physical components.

Figure 2.1 depicts other important characteristics of cyber-physical systems.

2.2.3 Cyber physical System Architecture and Design

2.2.3.1 CPS architecture

The study [11] investigated a wide range of CPS architectures, which vary depending on the application domains and system needs. These architectures include layered, cloud-based, multi-tier, SOA-based, and 5C-based designs. The five C's (connection, conversion, cyber, cognition, and configuration) constitute the majority of designs.

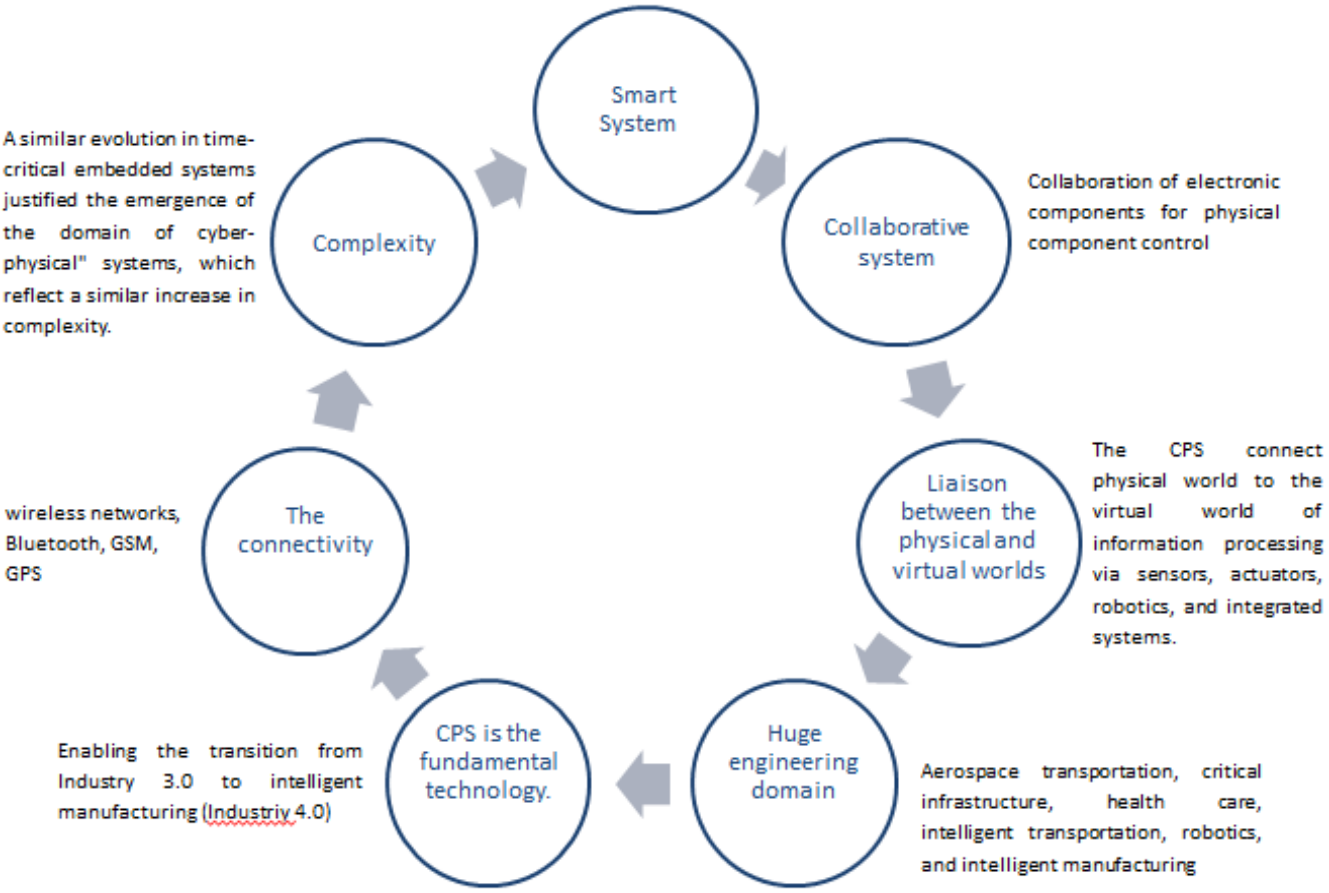

Figure 2.1 CPS characteristics

- The connection level is the point at which self-connection and self-sensing are made possible by the device's behavior and design.
- Conversion level: Machines can utilize self-aware knowledge to predict future issues by measuring the data from connected devices and sensors.
- Cyber level: Every machine generates and creates its own "twin" by using instrumented features.
- Level of cognition: A complete understanding of the monitoring system is required. It is possible to choose which jobs to prioritize and to maintain the smooth operation of the machine, as well as its current state. Collaboration is a part of diagnostic and decision-making processes, sharing knowledge acquired with consumers.
- Configuration level: The machine or production system can be set up based on the priority.

2.2.3.2 CPS Design

The design and modeling of CPS are extremely difficult because of their complexity and widespread application. Researchers have employed a variety of methodologies and techniques, model-based approaches, and tools to model cyber-physical systems, as clarified in [11].

Many architectural tools are available for modeling CPS and are used to describe complete systems, incorporating graphical tools that are helpful for examining the structure of CPS and the ways in which its component parts communicate and exchange data, such as SysML [12] and UML [13]. The two most popular modeling techniques are discrete-event and continuous-time modeling approaches.

Differential equations [14] were used to model the dynamic behavior of the system. Continuous-time methods [15] and hybrid discrete-event and continuous simulations [16] were used to describe the physical processes and analog circuitry. A reward-based model was used for the experimentation and simulation of CPS[17]. The Multi-agent model can be used for modeling CPS [18]. Multi-view Meta model based on UML [19].

2.2.4 Application fields

2.2.4.1 Manufacturing

CPS can be utilized in the manufacturing business to optimize processes by automating the entire process, resulting in a single decentralized platform for all plants. Manufacturing automation reduces labor and material costs while also shortening the production time.

Cyber-Physical Production Systems (CPPSs) are CPSs that are used in a production environment. They consist of digitally transformed industrial facilities and intelligent machinery driven by information and communication technology (ICT). They hint at increasingly inventive and linked industrialization processes [20].

Advancements in Information and Communication Technology (ICT) have enabled the growth of manufacturing processes. ICTs attempt to meet current market demands, boost responsiveness and agility, and ensure product quality [21]. In [22], they implemented a CPS on the work floor to address common CPS issues and enable intelligent production.

The Cyber-Physical Manufacturing Model [23] proposed at cloud to bridge the gap between CPS, cloud computing, and manufacturing.

2.2.4.2 Healthcare

CPS can be used remotely and in real-time to track the status and physical condition of patients. Additionally, by placing smart sensors in homes that identify accidents and immediately alert the system, CPS can be used to assist patients with aging in place. Table 2. 1 shows examples of CPS in healthcare.

Table 2.1 Some examples of CPS in healthcare

CPS used in Healthcare	Ref
CPS architecture as a layered cloud using big data was validated by a testbed based on robot technology	[24]
CPD is designed with layered centralized architecture	[25]
CPS on model-based architecture used for controlling the experimentation of a clinical scenario	[26]
CPS is designed on layers and validated by field tests, application scenarios, and use cases.	[27]

2.2.4.3 Smart grid

The primary application of cyber-physical systems is in electrical power grids. CPS must be utilized for the research, creation, assessment, and validation of power system security controls and algorithms [28, 29]. Information gathering, evaluation, decision making, and management are all essential components of a smart grid ecosystem. Both the operational facets of electricity generating and network communication are under their purview. To link end users to the smart grid, the CPS is responsible for maintaining and monitoring the transmission and distribution networks [28].

2.2.4.4 Smart transportation

Cyber physical systems are utilized to improve the performance of traffic control systems. A large number of cutting-edge electronic equipment and information systems are placed along the route to implement traffic control systems. In order to improve safety and efficiency, Cyber Physical Systems coordinate with the transportation process and integrate information into it [30].

2.2.4.5 Smart City

A smart city that uses ICT to deliver efficient services and improve the lives of its residents is powered primarily by a Cyber-Physical System which is integrated with modern software platforms and demands mobility, security, safety, privacy, and data-processing criteria [31].

2.2.4.6 Robots

Humanoid robots can be utilized to care for the elderly at home as well as for scientific research into underwater environments, space settings, and vital infrastructure protection. Sometimes used for personal reasons. It can also be used in agriculture. Cyber physical systems are widely employed in collaborative robots and their testbeds. Numerous open robotics platforms are available for researchers, such as Robotarium [32] and IRIS [33], which facilitate collaborative robotics experimentation with generic swarm algorithm development and evaluation. The Umbrella project [34] utilized and provided various testbeds for Gazebo robots.

2.2.4.7 Oil and gas industry

Exploration, production, and administration of petroleum can be aided by the use of petroleum cyber-physical systems (CPS) in various oil sector operations through the application of optimization techniques [35]. Using a metric of field data that is remotely collected at production wells, the CPS in the petroleum industry can accurately replicate fluid movement over the entire reservoir [36].

Industry 4.0, which also employs actuators and smart sensors to remotely maintain Supervisory Control and Data Acquisition (SCADA) systems and monitor operations in real-time, makes the integration of diverse industrial technologies with ICT conceivable [37, 38].

2.3 Foundations of embedded multi-agent systems

2.3.1 Definition of embedded multi-agent systems (EMAS)

Agent: A computational system with the ability to see its surroundings and take appropriate action, depending on its beliefs and motives. An agent is a physical or virtual entity that exists in an environment in which it has only a partial representation and can act. An agent is a real or virtual entity that exhibits autonomous behavior and evolves in an environment in which it can see, act, and interact with other agents.

MAS: A multi-agent system is a collection of autonomous agents that have some connectivity and operate in the same environment.

Embedded MAS

Definition1: An embedded mixed hardware system (MAS) is one in which the agents are software and hardware. Physical communication, physical acts, and perceptions in the actual world are examples of such interactions.

Definition2: It's an agent-based system that provides physical gadget social intelligence, autonomy, and proactivity.

2.3.2 Characteristics of EMAS

a) The agent is defined by the following characteristics:

Autonomy: The capability of an agent to operate without human intervention and make judgments on its own behavior and internal circumstances is referred to as autonomy.

Social: an agent collaborates with people or other agents to accomplish a task.

Reactivity: The ability of an agent to observe its surroundings and react quickly to changes that occur in them.

Proactivity: By taking the initiative, an agent can demonstrate goal-directed behavior.

b) An EMAS adhere to certain requirements derived from embedded systems [39].

Complexity: Developing a global model for networked embedded systems is frequently unfeasible because of their extreme sophistication.

Liveness: Early program termination may not be accepted by the system. It is important to avoid deadlocks. The system must be able to modify its behavior even if it provides a partial or unsatisfactory response.

Mobility: The movement of nodes within a Networked Embedded System must be considered in wireless communication.

Self-organization: To accomplish a system's overall objective, Networked Embedded Systems usually need to modify their behavior based on their interactions and local duties.

Integrity: The main way to keep a networked embedded system intact is to ensure that fault tolerance measures are in place to ensure functional integrity.

Aggregation of heterogeneous data: In embedded systems, computation is inherently characterized by heterogeneity. Similarly, there are other heterogeneities, such as synchronous versus asynchronous event processing or continuous versus discrete time control methods.

2.3.3 Agent classification

According to [40], agents can be divided into three primary groups: reactive, cognitive, and hybrid agents.

- Reactive agents: Reactive agents lack a long-term perspective of their surroundings. Consequently, they do not schedule their next activities in accordance with environmental change forecasts. However, because they provide redundant behaviors based on the responses to threshold effects, they are strong and dependable.
- Cognitive agent: This uses objective-based internal reasoning along with a representation of its surroundings and other agents to guide its decisions. The cognitive agent then remembers specific prior occurrences. As a result, he was able to reason about his perception of his surroundings, the effects that his actions would have, and the satisfaction that these effects would provide him. On the basis of these justifications, he made his decision. Consequently, he was able to memorize circumstances and how his surroundings responded to him.
- Hybrid Agent: combination of the reactive and cognitive agents.

2.3.4 Organization model

A multi-agent system is a type of agent society. There are two models: the AGR model (agents/group/role) and AGRS model (agents/group/role/service).

2.3.4.1 AGR model (Agents/Group/ Role)

From an organization-centric perspective, the Agent Group Role (AGR) agent architecture specifies the functions that agents play inside groups, such as a framework for creating multi-agent systems [41]. According to [42], an organization can be defined as a framework that describes how its members relate to one another and cooperate to achieve a common goal. The model was separated into three components:

a) An agent is an entity that actively participates in one or more groups.

b) A group is a collection of agents that share common traits or perform the same set of activities. Agents are limited to interacting with other agents in this group (but they can also interact with agents from other groups).
c) Role: This is a group-specific duty that must be performed by one or more agents within the same group; any agent in this group can assign this role to themselves.

2.3.4.2 AGRS model (Agents/Group/Role/Service)

The AGRS model aims to extend the AGR model to address the open difficulties in conventional multi-agent systems. Ferber and Mansour observed that the development of multi-agent systems frequently employs approaches centered on the agents or the social side of the agents, resulting in the creation of a closed and adapted system for a single specific application [43].

The concept of service is developed and used within an organization to clarify who can do what and how. In the AGRS model, the service is abstract and encompasses all functionalities that an agent might request or deliver.

2.3.5 Applications and domains

In accordance with [44], ambient intelligence, grid computing, electronic commerce, semantic web, bioinformatics and computational biology, space, education, military, and manufacturing applications, among other areas, are the primary application domains of multi-agent systems. In addition, MAS applications include information system governance, risk management, compliance, distance education, and remote control.

MAS are utilized in the real world for graphics applications, such as computer games, and have been used in industry [45]. Movies have also used agent systems. It is extensively recommended for use in mobile and networking technologies to provide high scalability, self-healing networks, and dynamic and automatic load-balancing. They are employed in coordinated defense systems. Transportation [46], logistics, power systems [47], smart grids[48].

2.4 Integration of multi-agent systems into cyber-physical systems

Industry 4.0, in accordance with [49], is centered on the development of intelligent objects and manufacturing methods as well as the incorporation of CPSs into industrial production, logistics, and services that have already been provided [50]. Intelligent systems such as (industrial) robotics, integrated cloud services, and big data analytics are made possible by the abundance of sensors that produce data. There are

numerous ways to integrate MAS in CPS, including in the field of robotics (which involves people, services, and robot-forming agent organizations), where these groups of agents must collaborate to share information, goods, and services. By employing these MAS to model and implement cooperation between higher-level organizations, it is possible to increase the autonomy and robustness of collaboration.

The authors of [51] noted that MAS and CPS operate based on very similar principles. They also suggested that a CPS's cyber portion should be designed with an agent to improve how the cyber portion exchanges data with the physical part and uses that data to perform specific functions.

Other studies, such as [52], characterized CPS as physical and engineering systems, with operations examined, organized, monitored, and coordinated by an intelligent core.

To implement an intelligent Demand-Side Management system (DSM), the authors presented in [53] a Cyber-Physical System (CPS) based on Multi Agent System (MAS) technology. The goal of this system is to maximize the power usage of public buildings powered by solar energy. A sensor network under node supervision is comprised of CPS. Every physical node in the building is placed in a room and paired with an intelligent agent in a virtual environment where the MAS are located.

Consequently, the suggested approach may help consumers optimize the sustainability and financial viability of their own energy plants. To study cross layer coupling and performance interdependencies for CPS, they developed a cross layer system model in [54] and addressed two system synchronization problems, where one hostile agent aims to manipulate the behavior of the entire system by interacting with physical layers. Reactions: In the context of a multi-agent system, distributed control is solved using the Nash equilibrium. A multi-level multi-agent method based on recursion was presented by the authors of [55] for the supervision and observation of large-scale artificial complex systems. They created a framework that can be applied to a wireless sensor network supervision system and allows a genuinely physically decentralized MAS to communicate with abstract multi-agent layers.

2.5 Design and Development Methods for CPS with EMAS

2.5.1Design methodologies for CPS with EMAS

In the literature, various methods have been proposed for developing CPS with EMAS. They introduced a two-layer architecture modeling approach in [56] to allow for runtime

intelligence across the entire system and real-time adaptation at the device level. The high-level cyber module, designed as a multi-agent computing model, and the low-level physical module, designed as an agent-embedded IEC 61499 function block model, comprise the first layer. Tests on the multi-agent simulation model, including NetLogo, and testbeds built on Jetson Nano and Raspberry Pi platforms were used to assess the proposed model.

They presented a general agent-based strategy for industrial automation system self-management in [57] using the design of a passenger elevator as an example of self-healing. The architecture of the approach is split into three levels: automation system connection level, functioning degree of self-management, and level of control and supervision. They introduced an application-oriented middleware architecture (AoMA) in [58] that enables decentralized smart sensing and management of industrial wireless sensor networks using multi-agent and IEC 61499 function block modeling.

The authors of [59] suggest a multi-agent system based on modeling that mimics a manufacturing workshop and is regarded as a Cyber-Physical System (CPS). To depict the workshop and its elements, as well as its application to transport tasks in flexible production workplaces, the model is a multi-agent fractal architecture.

2.5.2 Tools and languages for designing CPS and EMAS

The prospects are visible, multi-agent cyber-physical systems (CPSs), which are common in contemporary infrastructure systems, such as the upcoming smart grid, transportation networks, and public health systems.

2.5.2.1 Tools and languages for designing CPS

There are numerous modeling languages and techniques for CPS and their quality in the literature. The combination of models that record program and cognitive activity within the physical environment is a difficult problem, in addition to the difficulties of merging discrete-event and continuous-time modeling paradigms to improve the capacity to develop dependable CPSs in the future [7, 60].

Tools for architectural modeling of CPS are widely used to represent entire systems, such as graphical notations such as SysML and UML, which are helpful for thinking about the structure of the CPS and how its component parts communicate and exchange information. The two most well-known modeling approaches are discrete-event and continuous-time modeling paradigms. Continuous-time behaviors can be represented by modeling techniques that use mathematical notations. To create a physical model that can accurately predict how a system will interact with its physical surroundings, continuous-time modeling

is necessary. It uses differential equations and iterative integration techniques to capture the dynamic behavior of a system [61]. It has the advantage of continuous-time approaches for analog circuits and physical processes [62].

Hybrid discrete-event and continuous simulations [63], inductive constraint logic programming, hybrid timed automata [64], ontologies [65], information schema [66], UML [67], and SysML [68] are a few systems modeling approaches and tools for CPS design, analysis, and information dynamics modeling [69], the security meta model for software architecture (SMSA) [70], and multimedia software architecture (MMSA) meta model, which permits the description of software architectures and reliable cooperation [71].

Languages such as Stateow/Simulink, Modelica [72], hybrid CSP [73], and HyVisual [74] are specifically designed to model holistic embedded systems and CPSs. The goal of co-modeling, also known as collaborative modeling, involves building system models composed of independent models [75]. Crescendo is an approach to a co-simulation engine [76].

FMI is one way in which models written in one language can be coupled with models written in another language. The C language (behavior) and XML (interface) are used by the open standards FMI [77] and SSP [78] to create a model format for pre compiled models that may be shared between tools and/or merged to be co-simulated. Bloqqi [79] is a control software research language and tool that facilitates the creation of C code for deployment. The language was specifically designed to investigate the reusability of the control model. Modelica, a well-developed industrial modeling language [80], has several proprietary and open-source implementations. The goal of Modelica is to simulate and model cyber-physical systems. A research language called cumen [81] was used to model and simulate physical systems, with a focus on investigating rigorous simulations.

2.5.2.2 Languages and platforms for MAS

Although multi-agent systems can be implemented on a variety of platforms, Java is the language of choice for many agent and multi-agent development tools, which greatly increases the versatility of the multi-agent systems that can be produced. We discovered JACK, JADE, JADEX, JAgent, Jason and MaDKit.

a) Agent Builder

An integrated toolkit for creating intelligent software agents is Agent Builder. Based on the Agent0 [82, 83] and Placa [84] BDI models, it was created by Reticular Systems, Inc. The commercial product, AgentBuilder, is a closed source. Only tutorial agents accessed the free

evaluation version. There are three versions available: AgentBuilder Enterprise, AgentBuilder Pro, and AgentBuilder Lite.

Agent Builder is a comprehensive development environment for multi-agent systems written in Java that operates on it. It consists of two parts: the toolbox used to design the agents and execution environment. The development environment contains tools for managing agent-based software development, analyzing agent activity domains, specifying agent behaviors, designing inter agent communication networks, and testing and debugging software agents. Agents produced with Agent Builder communicate using KQML and are of the BDI type, which is implemented in the AgentO language. Software tools, which create connections between stages and practically cover every aspect such a frame reduces the adaptability of the tool. Only the Agent Builder agent model is used. Agents that interact with the environment or utilize a different model cannot be easily integrated. Learning AgentBuilder is challenging owing to its length.

b) MadKit

Gutknecht and Ferber developed the (Multi-Agent Development KIT), a scalable and modular multi-agent development platform. Likewise, the software on the platform was free. MaDKit is incredibly adaptable and permits the development of multi-agent systems with a large number of degrees of freedom in the agents' basic structure (which makes it simple to create reactive and cognitive agents) and communication protocols. MadKit uses an agent micro kernel and is mostly an MAS runtime engine. Aalaadin is the name of the underlying organizational model [85]. This platform is focused on organizing multi-agent systems, and the latest versions of both the AGRS and AGR model are natively integrated.

c) JADE

The Java platform JADE (Java Agent Development) is used for the development of agents and multi-agent systems. The software on this platform was free. The FIPA-ACL standard was supported by JADE. JADE fully supports the debugging and deployment of the developed systems. The platform consists of three components required to generate a multi-agent system: an agent management service, a service that facilitates agent communication, and an agent directory service.

d) JADEX

JADEX, an online resource, is an extension of JADE that is currently fully autonomous and created in Java. This platform, created by the University of Hamburg, adds a modular dimension to the system design, especially with the Jadex Agents, Jadex Processes, Jadex Rules, and Jadex XML modules. It can perform all the tasks that JADE can perform in terms

of applying, testing, and deploying agents and multi-agent systems. The fact that the JADEX permits the use of BDI agents should not be overlooked.

e) Zeus

Zeus offers a comprehensive setting for rapidly developing applications involving collaborating agents. It was developed by the Agent Research Program of the British Telecom Intelligent System Research Laboratory. Zeus' primary unique feature is the seamless integration of every phase from design to deployment. It employs real software engineering approaches (design patterns and UML) and offers theoretical and practical tools. Zeus is modular; however, it only supports an agent model, which restricts the variety of multi-agent system designs that may be developed. Likewise, Zeus is lengthy and challenging to learn.

f) Jack

Jack is a component-based framework for creating, executing, and merging commercial Java-based multi-agent systems. The Australian Artificial Intelligence Institute (AAII) developed the BDI model dMARS, which serves as its foundation [86]. Jack concentrated mostly on the stage of development. The Jack Development Environment (JDE), a graphical project management tool; the Jack Agent Language (JAL) compiler, which converts JAL applications into pure Java programs; and the Jack Agent Kernel, a library of auxiliary classes, make up the toolkit.

2.6 Conclusion

A sophisticated, multidisciplinary, physically aware paradigm known as a cyber-physical system (CPS) combines embedded computing technology (the cyber component) with the physical world (the physical portion). This chapter defines and summarizes CPS, including its characteristics and sphere of use. Embedded multi-agent systems are also specified, along with their properties, types, organizational models, and application domains.

The dynamic behaviors of the real world make CPS unpredictable because of the interaction required with the physical world. Therefore, a CPS must justify these modifications, modify its behavior accordingly, and ensure certain crucial elements such as safety, security, and confidence. In addition, a CPS can work with EMAS. New techniques and strategies must be developed to handle CPS difficulties. The use of intelligent agents to make these problems smarter and formally model them is one approach for solving them. Many studies in the literature have addressed the use of MAS in modeling CPS, and we offered a summary of these works in this chapter.

Failure to outline the CPS's precise characteristics and the cyber-physical system's functions. Additionally, the actors and participants in the CPS—users, developers, and potential clients—do not have their goals clearly defined. Failing to set goals, especially in the business world. The absence of safety and security safeguards and the unpredictability of system component behavior are the primary barriers to developing cyber-physical systems, particularly during the modeling stage. The next chapter will discuss the trust quality in CPS and EMAS and the methods currently used to test this confidence.

CHAPTER 3

TESTING TRUST IN CYBER PHYSICAL SYSTEMS AND EMBEDDED MULTI AGENT SYSTEMS

"Everything bears a second chance, except honesty and trust. When it collapses, it will not come back, even if you are given a thousand chances."

Wasini Al-Araj

3.1 Introduction

The fundamental idea of trust underpins all human relationships and influences social, political, and commercial rules and practices. These social rules for in-person encounters have taken thousands of years to be widely accepted.

From Wikipedia A trusted system depends on the enforcement of security policies and procedures. In the setting of computers, "trust" describes the basis on which a user sends data across a means of interaction.

The emergence of Cyber-Physical Systems (CPS), which are utilized in smart homes, healthcare, business, and other domains, has an effect on people's daily lives. A cyber-physical system is considered a trust system if it guarantees the principles of confidentiality, integrity, availability, security, and safety. The creation of CPS necessitates clear assessment methodology specifications and metric uniformity, measurement standards, computation techniques, and requirements governance.

Implementing trusted system technology can help a system defend against intruders and harmful malware. The value of trust systems has evolved to the point where they are now strategic initiatives. Recently, the research community has been tasked with ensuring the successful implementation, enhancement, and verification of trust in complex systems, such as CPS.

This chapter explores how trust is vital in the establishment of a new CPS and how researchers vary their conceptualization and verification techniques. Challenges in testing CPS are highlighted. Additionally, trust and its verification in embedded multi-agent systems are addressed.

3.2 Trust in Cyber-Physical Systems

Numerous study approaches have been used to assess the reliability of CPS; however, these approaches lack systematicity and long-term objectives. No work has tackled the problem of confidence in CPS overall. While some efforts have focused on the security viewpoint,

others have addressed the trustworthiness of CPS components, such as software, without addressing trust computation. Certain studies have been conducted in sequence and across a project. For example, in the study [87], they provided a foundation for methods that require engineering and design that consider reliability when designing the CPS. They offered architectures, design strategies, and thorough service requirements for systems that strike a compromise between reliability and trust. They offered a theoretical representation of the design time from start to finish trustworthiness assessment employing the trustworthiness report, process converter, formula builder, and trustworthiness profile builder of the beginning-to-end trustworthiness calculator (E2E TW calculator).

The lengthy calculation method in [87] is tied to metric values provided by software providers and should only be applied to certified products. Furthermore, few metrics are available.

The study [88] presents a layer-based security methodology that targets the security of each layer using OSI and PRM models. In [89], they presented trustworthy solutions for integrated manufacturing cyber physical, merging cyber-security and reliability criteria, and designing the cyber-security component with a resilient system framework.

Two tiers of trust are proposed in the trust-based secure cyber physical systems approach [90]: a) External Trust, which stands for the CPS's actual physical surroundings; and b) Internal Trust, which consists of trustworthy internal components including sensors, actuators, and communication networks.

The issue of information-sharing quality among interconnected critical infrastructure is described in [91], where they suggest using the policy-based management paradigm in conjunction with Trust and Reputation management. They suggested that the application of policies and trust indicators can improve the accuracy of risk indicators, aid in establishing trust in system management, and assist CI operators in assessing the relationship between their CI and their peers.

The author of [92] offered a blockchain-based approach to problems with privacy, security, and trust. The advantages of blockchain technology for CPS have been explored, demonstrating how blockchain facilitates secure CPS, and the process of integrating blockchain technology into various CPS components.

Comparison of existing trust framework

Table 3.1 presents a comparison of some existing frameworks for evaluating trust CPS in the literature. The points of comparison are related to the methods used, type of evaluation, and advantages and disadvantages.

Table 3.1 Comparison of some existing framework for trust CPS

Framework	Methods	Requirement/ Metrics	Trust Evaluation	Advantages	Disadvantages & Limits	Application Domain
Framework for requirements engineering and design [87]	Framework of requirements engineering	-Elicitation of trust requirements and Trust concern. - Conflicts on the goal level between actors.- Resource(human or non-human).-. incorporating the need for trust into an organization's procedure - Set of trustworthiness properties	- Modeled trustworthiness requirements in BPMN. - End-to- end trustworthiness evaluation.- Evaluated each attribute based partial trustworthiness measurements of each asset.- System topology, or structural compliance, is achieved by combining trustworthiness data at different granularities.	- needs from various actors involved in the early stages. Evaluate trustworthiness-related risks after description of describe trustworthiness requirements.- The activity is completed while the system is being designed.	- Used for certified product only -Computation process is so long. - set of metrics is very limited - Metrics not classified.	General CPS
P2P overlay for intrusion detection [93]	-JXTA framework. -Emulated network (test bed network using VMWare 4.0)	-Presented on the set of services for network organization' management, and communication between peers. - Metrics: Security ,integrity , authenticity, peers' trustworthiness	-Based on event's evaluation - The correlated alert is computed by combining the received confidence with the trust.	-Detect real Internet worm attack -Target Internet attacks and increase the detection and protection capabilities	-Trust based on peers ,incident generation . Evaluation target not specified clearly -Lack of well-defined evaluation process. -Target only the network.-Lack of metrics of trustwort-hiness and trust	Network
Integrated Data-Driven Framework [94]	- secure correlation-based encryption/ decryption -Trustworthiness judgment approach - reliable countermeasures - High design degree of freedom (DoF)	Requirement of security or trustworthiness not mentioned - Metrics : safety, performance, Security, robustness	-Tests on a simulated two-area power grid validate the theor-etical outcomes.- Not clear evaluation of security / trustworthiness, It employed performance assessors that included mean absolute errors (MAE), root mean squared errors (RMSEs), decision-making logics, false alarm rates (FARs), and miss detection rates (MDRs).	-Target secure transmission and attack detection -target hardware and sensor network data - Target eavesdropping attacks	- The primary goal is CPS security, not rustworthiness. -Lack of metrics of trustworthiness and trust.- Trustworthiness Evaluation target not specified	Applied to control power grid system
Trust handling framework for Industry 4.0. [95]	Multidimensional trust models - Based on analysis of related work	-Set of social Attributes and various trust factors .-Set of IT and Computer Science attributes.-Set of attributes for Ad-hoc and Distributed Sensing Systems.	Survey and summary of existing evaluate method for trust.	-Explained Credibility in trust handling framework. -Analyze trust concept via existing wok in literature	Identifying only trust models in research areas related to Industry 4.0	Social sciences, Information systems, Distributed ad-hoc networks
TRUST.IO Framework [96]	- Establishing a reliable connection between CPS and the user's equipment. -A trusted remote Client Device.-Embedded ARM platform.- Using Advanced Encryption Standard (AES)	- Requirement of security or trustworthiness not mentioned.-Not clear definition of metrics only security is figured	-Protecting input/output (GPIO) interfaces on CPS.-Protecting CPS from attacks and malicious software.- Not clear evaluation of security / trustworthiness -Not target a quantitative evaluation	-Automatic. - Transparent -Minimal.-runtime overhead.-Minimal code modifications	- Target only the security for physical interface of CPS - Target only the communication and control between client and hardware of CPS	Applied to existing CPSs

3.3 Trust evaluation in cyber physical systems

Trust has no exact definition or mathematical calculation. Probability or statistics are employed to assess trust, particularly in dynamic networks with rapidly evolving topologies. The literature addresses trust evaluation in various ways and techniques.

They integrated several methods in [87], including end-to-end trustworthiness assessment. and assesses each feature according to the partial trustworthiness ratings of each asset. Integrating reliability scores according to granularity levels aligns with the topology or structure of the system. Evaluations of events also make use of these.

In [97], generalized stochastic Petri nets (GSPNs) were used to analyze and assess the trustworthiness of command and control cyber-physical systems (C3PSs). Three metrics were used to assess the trustworthiness of CPSs: availability, security, and reliability. For GSPNs, modeling power was applied, and trustworthiness was thoroughly assessed quantitatively. A stochastic Petri net model was presented in [98] to assess the security and resilience of CPSs against malicious attacks.

The trust-induced MAPE-loop was presented in [99], which is a workable technique for determining trustworthiness in computing systems by measuring fidelity. used the gathered data to evaluate the drifting's features in fidelity, extracted the machine's and user's dynamic qualities, verified the system's reliability, and implemented safety assurance measures.

In [100], the authors expanded the assessment of system trust by using psychophysical parameters. They acquired skin conductivity data using a commercially available Shimmer Sensing (GSR) sensor, considering factors such as the state of the operator who operates the UAV and standard metrics gained through interviews. Barka et al. [101] presented a new connectivity architecture for Flying Named Data Networking (FNDN), based on trust monitors.

According to the authors of [102], trust in UAVs is provided both within their missions and among their nodes. They proposed a trust model that estimates the direct trust levels by optimizing the weights of multiple parameters using a genetic algorithm. Their final host trust value was determined. Three reliability attributes and one performance attribute are used to assess a node's trustworthiness. Node reliability is reflected in the packet delivery ratio, node energy, and signal strength.

3.4 Methods and Techniques for Testing Trust in Cyber-Physical Systems

3.4.1 Testing cyber physical systems (CPS)

To fully grasp the issue of testing cyber-physical systems, three key concepts must be articulated: cyber-physical systems, application domain, and systems testing. The testing step determines whether the system fits the intended requirements. The requirements were previously digested by (client, system, developer), followed by analysis, then the design of the solution itself, followed by real coding of the program, testing to ensure that it works, and finally putting it into production. This technique is not only the foundation of many key software breakthroughs, but also has aspects similar to those of the 5C model for manufacturing. Requirement engineering is enforced as the first phase of system development and includes requirements for analysis, specification, and validation [103].

Requirement engineering for cyber-physical systems is extremely challenging because of the massive system sizes, component heterogeneity, involvement of multidisciplinary stakeholders and machines, and continuous change [104].

3.4.2 Testing trust in cyber physical systems

Trust in cyber-physical systems is covered from several sides in the literature; security and cyber security are the main focus and considered the pillar of trust in the majority of studies, while other aspects of trust, such as system quality, are disregarded. The current approach for assessing and testing trust in CPS is based on this approach. Many tools for testing trust from the cyber security angle are widely used, such as Tesbed.

3.4.2.1 Testbed for testing cyber physical systems

Many research papers in the literature propose a testbed as an effective tool for modeling and testing cyber-physical systems in a variety of sectors, including electrical energy, oil and gas, and spatial industries, as well as for a variety of test targets. The testbed simulates the entire CPS and its virtual counterparts by combining the advantages of virtual and physical testing [105]. The testbed was primarily used to demonstrate data acquisition for production and maintenance. It consists of a semi-automatic production line, data analysis capabilities, and commercial software for system monitoring [106]. A dataset of testing methods for CPS is presented in [107].Table 3.2 presents some of the surveyed tesbeds for cyber security.

Table 3.2 A sampling of surveyed testbed works

Testbed Name	Domain	Objective	Content /Soft & Hard	Ref
Power-Cyber	Smart Grid	Cyber Security	ISEAGE, RTDS, Sie ,DIgSILENT, Power Factory, Power TG EMS, SICAM RTU	[108] [109]
SGDRIL	Smart Grid	Cyber Security	RSCAD, RTS, Relay/PMU tester, SEL Relays, PDC, SVP and RTAC	[110] [111]
CCADS	Smart Grid	Cyber Security	IDS,GLA,TFPG,FCM	[112]
-	Smart Grid	Cyber Security	PMU tester, PDC	[113]
-	Smart Grid	Cyber Security	RTM, HIL, RTDS, DeterLab	[114]
SGDRIL	Smart Grid	Cyber Security	HIL,RTS,DGI	[115]
	Smart Grid	Cyber Security	WAMPAC- SCADA	[116]
SGDRIL	Smart Grid	Cyber Scurity	NIDS,SASs	[117]

3.4.2.2 Formal methods for testing CPS

Formal methods are rigorous mathematical approaches used for creating and testing hardware and software systems. The objective was to ensure that the systems operated correctly. Several formal verification techniques have been documented in the literature, the most significant of which is model checking [118]. Certain studies have concentrated on automatic verification techniques; for example, in [119], system reliability and security were verified using formal modeling, static language checking, code generation, optimization, and visual depiction of counterexamples. It employed timed discrete event systems and timed controller models in conjunction with timed plant models and closed-loop as a formal technique to target timing needs [120]. Software modules built in several languages were converted to timed automata in [121]. It adopted the PLCopen XML format and applied function blocks and UPPAAL timed automata models. Other studies focused on model-based testing. Model-checking BNDC was provided in [118], with a focus on system security and the use of process algebra SPA and BNDC (bi-simulation-based no deducibility on compositions). The authors of [122] target safety and timing criteria through the application of semi-formal requirements engineering and systematic mapping.

3.4.2.3 Simulation

Simulation is a crucial tool for confirming and testing both novel and established methods. For complicated issues that are dynamic and cannot be resolved with basic (mathematical) models, there are two methods for simulating and modeling complicated systems. The first uses tool coupling to perform a co-simulation and is based on well-known simulation tools from the relevant domain. The second strategy is to represent the entire system using a single appropriate language, such as Modelica [123]. CPS simulations depict the system in terms of software execution and network topology for computation nodes, routers, switches, and network data packets [124]. Depending on the

system levels or processes, assembly lines, and logistics, numerous types of manufacturing simulators exist. Offered a multidisciplinary framework for co-simulation and simulation in [125], which examined the behavior of the system and the interactions between its components in many engineering disciplines. Simulation-based prototyping is being increasingly used in CPSs to assess early designs that are sufficiently detailed in their modeling using integrated open source tools, standard languages, C/C++, SystemC, and the Open Dynamics Engine. This approach has recently been adopted in manufacturing to expedite the product development process and support early evaluations [126].

3.4.2.4 Digital twin for testing CPS

A digital twin (DT) model for CPS verification was proposed by Choi et al. [127]. The integrated platform enables data analysis and fault identification, and uses digital 3D modeling to replicate the system's operations and data. In the simulation by Amini et al. [128], failures within a physics-based digital twin of wind turbines were identified using a neural-network-based classifier, and the test failure rationale was inferred using a black-box technique. This approach functions without relying on a digital twin or cyber-physical system. In several studies, the most popular testing method based on creating digital twins is the use of neural networks. To replicate the right behavior of the system, existing approaches build a digital twin utilizing a historical data collection of the correct behavior of the physical asset [129, 130, 131, 132, 133].

3.5 Challenges in Testing Trust in Cyber-Physical Systems

Engineers and testers often face challenges while working with cyber-physical systems. Greater independence, accompanied by scale and complexity increases [134,135]. The particularities of CPS as a complex system and the particularities of trust as a complex quality to be modeled and tested propose many issues. The research community has addressed different challenges in Modeling CPS and its reliability. Before discussing testing trust in CPS, we must first check the design of CPS and its functionalities, and then its qualities. In this context, we can mention the following challenges for complex CPSs with strict requirements for robustness, security, and reliability:

3.5.1 Challenges to the trust modeling in CPS

Numerous approaches have been proposed to model trust in a system . However, it does not offer appropriate guidelines for a methodical approach or modeling language that is suited for supporting this kind of trust-based study. Achieving trust by design in CPS.

and methodically integrating trust engineering into system development from the beginning of the system lifecycle are the two biggest issues.

The particular difficulties and constraints of the present study include defining trust needs and concerns that might be relevant to several parties, focusing exclusively on security quality, and perceiving trust as a means of achieving security and security services without considering other trust attributes. The selection of modeling approaches capable of handling the intricacy of trust in CPS is another challenge.

Linking discrete-event and continuous-time modeling methodologies to enhance the development of reliable CPSs is a difficult task that is made more difficult by the inability to combine models that depict program and cognitive activity with physical surroundings [136].

3.5.2 Challenges that face the verification of trust

a) Uncertainty: Uncertainty permeates every aspect of the CPS because of the intricate, time-varying relationships between physical systems, network hardware, and computational infrastructure, uncertainty permeates every aspect of CPS. One of the most difficult tasks to complete is determining these uncertainties, particularly in the creation, implementation, and analysis of online tests. [137]

b) Real-time CPS testing: Achieving high real-time CPS testing is a crucial, yet challenging task for future CPS research. Communication delays, intricate interdependencies between various components, and the impact of fault propagation influence an entire CPS [138].

c) How testing functionality of CPS with non-functional properties: Testing the functional aspect and quality of CPS, as trustworthiness and security differ in techniques and tools. Therefore, combining these tools is challenging [139].

d) Automation: Automated testing is a critical topic for test improvement.

e) Multi-objective: No test can fully evaluate a CPS's resilience, robustness, fragility, or dependability. Future studies will focus on creating a test for effective multi objective testing, which will include simultaneous functional and nonfunctional testing.

f) Reliability, continuity, and reproducibility: CPS testing must be continuous and repeatable to obtain consistent results.

3.6 Trust in embedded multi agent systems

Numerous concepts, such as the interaction between an agent's capacity and knowledge, are used to treat trust in multi-agent systems [140]. In the research area, there are three main approaches to building trust in MAS: Trust can be based on the agent's cognitive state, as in [141], or it can be reputation-based, as in [142], a method using bidirectional relations that are provided at the design phase [143]. A model for checking quantitative trust within a group was investigated in [143].

A dynamic logic approach is presented in [144]. A trust quantification paradigm based on subjective logic was suggested in [145], wherein agents' activities are interpreted and a trustworthiness value is assigned.

The authors of [146] presented an information security mechanism in which onboard sensor devices are used to help robot agents build trust levels with one another based on circumstance analysis developed at a specific stage of an iterative process. They also proposed a method to determine an agent's reputation as a gauge of the sentiment that the public has developed over time regarding the characteristics of robots falling under the category of "saboteur" within a group of genuine robot agents. It was demonstrated that in multi-agent systems, inter-cluster distance can be used as a metric to assess the strength of trust models.

3.7 Methods for testing trust in embedded multi agent systems

3.7.1 Test driven development (TDD) paradigm

The test-driven development (TDD) paradigm and a framework with test scenarios that evaluate each agent separately were employed by the authors of [147]. This framework is based on the idea of creating mock agents and evaluating typical patterns for agent interactions. To create stereotype implementation for matching test cases, they examined the most prevalent agent interaction patterns, such as pair and mediation patterns. The authors of [148] suggested a test-driven method for building MASs that facilitates incremental and iterative MAS constructions. Additionally, they unveiled SUnit, a testing framework that builds on JUnit and enables the creation of automated tests for agent behaviors and interactions. It also supplies the mock agents required to replicate the organizational features of the MAS.

3.7.2 Trust based solution

The study [149] established the asymmetric convergence of their trust-based solution for the discrete-time multi-agent problem and proposed RoboTrust, a trust algorithm that determines an agent's trustworthiness based on statistical inferences and observations drawn from a variety of historical viewpoints.

3.7.3 Trust based feedback satisfaction

The trust establishment model developed by the authors of [150] focuses on trust evaluation. The model assesses trustees' degree of satisfaction by measuring and analyzing their demands using a multicriteria approach, taking into account their stated preferences and values. Trustees try to change their behavior based on their comments to increase their confidence levels. The trustworthiness test is based on satisfaction reports from trusters and involves simulations.

.3.7.4 Trust based reputation

A new trust reputation model known as TRR was presented in [151]. It considers the interdependence of all trust measures generated in a system from a mathematical perspective. The model calculates a parameter that expresses the significance of reliability in relation to reputation. An agent's capacity to recognize and assess the skills of others, as well as the knowledge it possesses, determines how much trust it can place on other agents [152]. The prevalent approach for evaluating trust in MAS is reputation based [153]. In the same context, evaluation of concept-based reputation is led by assessing other agents' trustworthiness by drawing on a variety of data sources, including firsthand experience and witness accounts. In [154], they introduced the well-known model of Trust and Reputation in the Context of Inaccurate Information Sources (TRAVOS), which mostly falls under the evaluation component.

3.8 Challenges of testing trust in EMAS

Because software agents are unique in that they are autonomous, distributed, intelligent, and have varied communication protocols, testing software agents is seen as a difficult task. Additionally, testing the trustworthiness of an agent is challenging. Agents interact in a concurrent, asynchronous, and decentralized manner, making MAS a complex system. Additionally, because it is challenging to predict ahead of time every contact an agent will have while executing a task, agent behaviors are nondeterministic. Software agents are therefore challenging to test and debug.

Another issue is that if we treat every agent in a multi-agent system as equally trustworthy, we have an impasse. However, in certain scenarios, such as military applications involving sensor fusion or multi robot formation control, assuming that every agent is reliable could lead to compromised network security or subpar cooperative performance.

Real-time testing of EMAS is a challenging task. Communication delays, intricate interdependencies between various agents, and a large number of sent and received messages.

3.9 Conclusion

Numerous studies and methodologies have addressed the testing and assessment of reliability in CPS, which are discussed in this chapter. However, they differ in terms of the trust perspective and test aims. Numerous challenges are raised by the particulars of trust as a complex attribute that needs to be modeled and assessed, as well as the particularities of CPS as a complex system. The research community tackled various issues related to CPS modeling and its trustworthiness. This chapter highlighted some test methods for trust in CPS and the challenges related to trust modeling and verification in CPS. Different trust concepts for embedded multi-agent systems are presented, as well as some test methodologies and challenges.

The existing approach in the literature does not offer suitable standards for a methodical approach or a recognized modeling language to enable trust-based analysis when it comes to modeling trust in CPS. The two most important issues are systematically integrating trust engineering and testing tools into system development from the outset of the system life cycle and building trust by design in the CPS. The next chapter addresses these challenges and proposes a multidisciplinary approach to modeling and testing trust in CPS from the early design phase.

CHAPTER 4

EMAS TEST DESIGN

"The core of scientific progress is the spirit of inquiry and the courage to challenge the status quo."

Elizabeth Blackwell

4.1 Introduction

Through the state of the art [Part I], we have detailed the field of cyber physical systems. We distinguished specific aspects of CPS, as well as their main characteristics and architectures. For embedded multi-agent systems, we have also identified the characteristics of agents and EMAS, and then presented different agent models and existing multi-agent platforms. The test methods for CPS and EMAS are also presented and discussed. For the second part, we propose a formalization of analysis, modeling, and verification based on an analysis framework, a design model, and a test model. We consider the particularities of CPS as a complex system and the particularities of the quality to be tested. We show how these needs were specified and led to the formalization of a testing approach for cyber-physical systems.

Collective cyber-physical systems and cyber-physical systems are considered complex artificial systems. The research community is interested in the possible contributions of embedded multi-agent systems in the development and testing of CPS. Requirement engineering is the initial phase of system development and includes analysis, specification, and validation. The verification aims to establish whether a system meets a set of requirements.

The aim of this study is to test trust in cyber-physical systems. Trust, as a system quality, comprises several sub-properties that are related to the system itself, the environment, industry, human interaction, and the use of essential systems. The question is how to develop trust in CPS in the early design phase, how to ensure that this quality exists throughout the CPS's life cycle, and how to guarantee that the qualified system meets all of the recommended intentions and requirements of stakeholders and users. How to verify this quality within CPS efficiently, while covering all the proprieties of trust and requirements. This chapter introduces our proposed solution to the aforementioned challenges. We begin by providing an overview of the suggested technique and objectives, as well as details of each phase and the advantages of the proposed test model.

4.2 Proposed Solution: Overview and objectives

4.2.1 Overview

Our proposed solution is built on three pillars: the first is to analyze the trust quality of CPS and present a comprehensive set of attributes and requirements within the trust degree framework, as well as classify trust concerns for cyber-physical systems. The second pillar discusses how to model trustworthiness in CPS at the early design phase on the basis of the ranked trust degree in phase one and how verification is implemented in the model. The final step is to verify the trust quality of the CPS with an embedded multi-agent system.

4.2.2 Objectives of the proposed methods

General objectives

1) Analyze the trust quality of crucial structures such as CPS and establish all relevant trust boundaries in terms of concern, requirements, and attributes for CPS in its environment.

2) Facilitate the evaluation and verification of trust in the CPS.

3) Our test approach aims to improve tests with analytical proofs, optimize tests through early design, and prevent untestable or non-deterministic failures.

4) Change the paradigm of how the current V&V security and safety critical systems work.

Special Objectives of EMAS TEST Approach

1) Testing the trust of cyber physical systems with embedded multi-agent systems.
2) Testing the trust of embedded agents in the EMAS test model. Consider the trust of an agent as reputation. How an agent is present, how it accomplishes its task, and how it interacts with other test agents in the EMAS TEST model. Figure 4.1 shows the proposed solution.

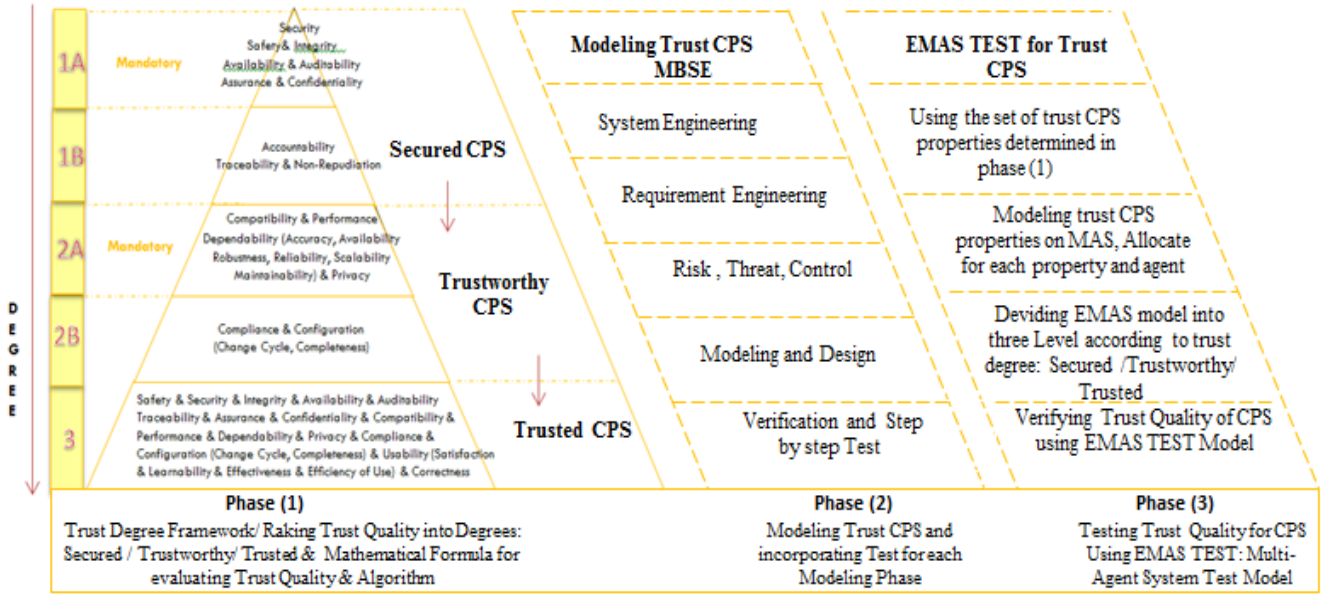

Figure 4.1 Proposed framework, models, and approaches for testing trust quality in CPS

4.3 Proposed Solution Phases

4.3.1 Phase one: Trust degree framework

We proposed a flexible framework for identifying and categorizing trust degrees according to a set of requirements and obligated properties [155]. Our approach separates the

collection of trust qualities into three subsets, each of which is defined as a minimum criterion for each degree that the system must satisfy to be considered a trust system. As can be seen in Figure 4.1 for phase one, each degree has a specific set of traits ranging from the lowest degree one to the greatest degrees two and three. The qualities of the first degree are coupled to those of degree two, and so on, up to degree three, as we proceed with the degree scale.

4.3.1.1 Degrees Definitions

a) Degree One: Secured CPS

Secured CPS focus on addressing security challenges and implementing security measures within the system. Contains the obligatory and essential qualities for a secured CPS defined by the organization or user, which are typically safety and security.

b) Degree Two: Trustworthy CPS

A cyber-physical system is considered trustworthy if it has a sufficiently high degree of safety and security, a set of other attributes, and a commitment to fulfilling its users' expectations of trust includes the necessary and basic qualities for trustworthiness that are adjusted by the user or organization.

c) Degree Three: Trust CPS

When end users have faith in a CPS system, CPS may be trusted. A trust CPS is a system that satisfies mandatory attributes and properties such as security, safety, and trustworthy attributes, in addition to a set of other properties to guarantee the CPS's trust quality.

4.3.1.2 Trust requirements

To properly define the trust concept and analyze the criteria, we divided the collection into a portion of the secured criteria that form the basis for addressing the vital security and safety requirements. Together with other needs linked to trustworthy characteristics, the aforementioned demands are included in the category of trustworthy aspects. The portion of CPS trustworthy requirements consists of extra trust-related criteria in addition to a portion of trustworthy requirements.

a) Secured requirements

The security requirements are based on security issues and safety considerations. Security can be divided into three types.

- Confidentiality: data and procedures are protected and kept confidential to prevent unauthorized disclosure.
- Integrity: protection against unauthorized changes to data and procedures.

- Availability refers to the means of service activity and data protection against the denial of service to authorized users.

For safety, according to ISO 26262, safety involves the prevention of accidents and defense against flaws, mistakes, and failures.

b) Trustworthy requirements

A cyber-physical system is considered trustworthy if it provides services that may be justifiably trusted, meeting both security requirements and system dependability standards.

- Privacy: Issues with privacy related to the CPS's capacity to prevent entities (people, computers) from accessing data that have been stored or transferred across several components. Users can access and control their personal data.
- The performance is the rate at which a service operates. Extent to which a system or component satisfies its intended purposes while adhering to predetermined constraints, including speed, accuracy, and memory usage.
- Dependability: Dependability is a set of sub-properties that can be justified by the service that it offers, including accuracy, robustness, reliability, scalability, and maintainability. The execution proceeded as planned, correct, and predictable.
- Compatibility: The capability of software and hardware from many sources to function together without modification.

c) Trusted requirement

The requirements are concerned with how people interact with CPS and within. Its foundation is a) the human element: apprehension about the features of CPS in relation to human usage. b) Usability: concerned with the degree to which a CPS can be utilized to successfully accomplish its functional goals and the content of its users.

4.3.1.3 Metrics

The literature does not categorize trustworthiness and trust attributes according to functioning or obligation. Our study [155] addresses this issue. Most characteristics are defined using the glossary terminology [156].

During the requirements engineering stage of CPS design, the description of the necessary attributes for trusting the CPS is completed.

We provided an attribute classification and a set of judging criteria in [155] based on system functionality and duty. As shown in Table 4.1, certain required attributes are not governed by rules or standards; instead, they rely on the precision and responsiveness of stakeholders' or users.

TABLE 4.1 MANDATORY PROPERTIES FOR TRUST CPS.

Properties	Definition	Sub-properties
Safety [157][158]	The capacity to function without endangering the environment of the system or putting people in danger.	Fault tolerance, robustness
security [159][160][161]	Security is the capacity of a hardware and software system to safeguard resource access and thwart attacks and misconduct.	Accessibility, responsibility, auditability, guarantee, tracking, authenticity, privacy, and non-refusal
Compatibility [162]	The capacity of hardware and software from many sources to cooperate without requiring changes.	Openness
Performance [162][163]	Throughput is the number of event replies processed in a given amount of time. Reaction Time: The duration of time required by the service to finish one transaction.	throughput, Response Time
Dependability [161][162]	The capacity to deliver a service and the planned execution were accurate and consistent.	accuracy, availability, robustness, reliability, scalability, maintainability
Privacy [157][163]	The capacity and features of the system that let users control how their personal data or information is used,	Nobody else is privy to or can use their personal data.
Usability [ISO 9241-210][163]	Is a set of requirements that a special product or system can meet simultaneously. It relates to how simple it is for a user to pick up the skills necessary to use, set up input for, and decipher the output of the service. favorable opinions on using the service	contentment, teachability, efficacy, and usage efficiency
Correctness [162]	The term "correctness" describes how well a system responds to user needs, particularly those pertaining to requirements for reliability and expectations of trust.	Check if the behavior of a system satisfies the user's criteria.

4.3.1.4 Trust concern for CPS

In [164], we presented an analysis of trust concerns and described the entire set of trust-related demands and anxieties that drive the creation of the desired trust quality in CPS from the beginning of the design process. Every fear class is a grouping of attributes that can be associated with one or more of the following: functional, human, business, or trustworthy. The interconnectedness of the classified concerns about trust is shown by the specification of the attribute set for each type of concern. In addition, it demonstrates how risk mitigation may be aided by concern identification. Many functional and nonfunctional characteristics and evaluations form the basis of trust as a CPS feature. The requirements are an interpretation of the fears, and are prevented and decreased by meeting the necessary attributes of the system. Our classification of trust fear is based on a combination of multiple factors.

a) Functional concerns

Concerns regarding the functionality of CPS pertain to a number of different elements and parts, such as devices, sensing, controlling, and communication.

- Functionality: The services offered by a CPS are a subject of concern. Certain functionalities are limited to the CPS physical characteristics.
- Information sharing between CPS and other organizations is the subject of communication issues.

- Controlling: The ability of a CPS to control a physical part's properties is the primary concern.
- How a CPS can influence changes in the physical world is addressed by actuators and the issues that surround them.
- Sensing: Sensing fears pertain to how CPS uses the physical data gathered by sensors to create a link between the physical and cyberspace.
- Interaction between the industry environment and the realm of CPS capabilities.

b) Human concerns

Humans are crucial players who interact with CPS and attest to their reliability. The concern is that the user will become aware of any differences between the system's development and their needs. The user can eventually reject the system as a result of losing faith. Human concern encompasses the following qualities: satisfaction, usability, and interactions.

c) Business concerns

Business must has to account for the basic elements and costs necessary for the development of CPS, in addition to the expenses associated with staff training. To guarantee its regulation, the cyber physical system must be accredited. Because it is dependent on the availability and deployment of CPSs, the time to market is another commercial challenge.

d) Trust concerns (trust as a quality)

Our trust degree framework [155] divides the collection of trust concerns into three groups: security, trustworthiness, and trust.

- CPS security concerns: Security and safety considerations are related to each other. The former focuses on ensuring that the CPS functions without endangering human life or the environment. Secrecy, integrity, and availability are security sub-properties that pertain to the ability of the CPS to guarantee that none of its systems, procedures, or products are vulnerable to unauthorized or unintentional access, change, loss, damage, or use.
- CPS trustworthy concerns: Concerns regarding performance, dependability, interoperability, and privacy are related.
- CPS Trust concerns refer to concerns about usability or the capacity of CPS to be used in a way that effectively satisfies both user needs and operational goals. Accuracy comprises system behavior that conforms to user specifications.

4.3.1.5 Proposed Trust Evaluation

As mentioned in [155], our goal was to strengthen CPS security and control. We employed a technique inspired by a differential-equation-based continuous modeling approach.

frequently employed in controls to represent a range of physical processes [165]. A group of discrete units and shared variables performs stability, security, and theoretical research, as in the Modelica language [166].

a) Proposed CPS trust equation

In our study, the assessment of trust in CPS was suggested [155]. When there were shared variables for trustworthy measurement and trustworthy degree (Y), trust degree and trusted measurements (Z), and secured degree and secured measurements (X) in the metrics conjunction, we used a simple equation.

$$X = \sum_{i=0}^{n} Mi \qquad (1)$$

- **M**: Secured metrics (can be evaluated by value, or presented by percentage)
- **n**: Number of secured metrics
- **L**: Number of mandatory metrics (a set of safety and security attributes)
- **L=2** (Mandatory metrics are safety and security)

$$Y = \sum_{k=0}^{m} Mk \qquad (2)$$

- **M:** Trustworthy metrics (can be evaluated by value or percentage
- **m**: number of trustworthy metrics
- **S:** Number of mandatory metrics (a set of trustworthy)
- **S=4** (Mandatory metrics are: privacy, dependability, performance, compatibility)

$$Z = \sum_{j=0}^{p} Mj \qquad (3)$$

- **M:** Trustworthy metrics (can be evaluated by value or percentage)
- **p:** number of trusted metrics
- **Q:** Number of mandatory metrics (a set of trusted metrics)
- **Q= 2** (Mandatory metrics are: Usability, Correctness)

$$T = \sum_{i=0}^{n} Mi + \sum_{k=0}^{m} Mk + \sum_{j=0}^{p} Mj \qquad (4)$$

T: Presents the final computation of trust.

b) Determine the threshold values

There is no formal definition of trust, no mathematical formula. Calculating trust involves using statistics, or probabilities, notably in dynamic networks with rapidly changing topologies.

• X presents the secured metrics (a set of safety and security properties). L: number of mandatory metrics (set of safety and security attributes or metrics) When X = L, that means the mandatory metrics of Degree one (secured CPS) are verified. If X > L, the mandatory

metrics are verified along with the expected or preferred metrics. In this case, the system is secure.

• At that point, the previous degree is considered verified, and the system has achieved degree 1 (the system is secure). Y presents the trustworthy metrics (a set of trustworthy metrics). S: Number of mandatory metrics (set of trustworthy metrics) If Y=S, that means the mandatory metrics of Degree2 are verified. If Y > S, it means that the mandatory metrics have been verified, as well as the expected or desired metrics. In this case, the system is trustworthy.

• At that point, we consider the previous degree verified and the system has achieved degree 2 (trustworthiness). Z presents the trusted metrics (a set of trusted metrics or judgment metrics). Q: Number of mandatory metrics (set of trusted metrics) If Z = Q, that means the mandatory metrics of degree 3 are verified. If Z > Q, it means that the mandatory metrics, as well as any expected or desired metrics, have been verified. In this case, the system is the trust system.

• On the assurance of security, trustworthiness, and trust qualities, we formulate a quantitative view. A number with a range of possible values; also known as a confidence interval. For that reason, we consider:

a. To reduce uncertainty, each mandatory attribute will be evaluated as 1 if verified, and as 0 if not verified.

b. No mandatory attribute, if verified, will get values in the range 0-1; it can get [0, 0.25, 0.50, 0.75, 1].

c. If mandatory attributes of each degree are verified, we can consider a system trust CPS.

d. The verification of sub-properties is considered part of its properties 'verification. Only properties are taken into account.

4.3.1.6 Advantages of our Trust Degree Framework

In a comparison of our Trust degree framework and existing frameworks in the literature, our Trust degree framework advantages are:

- Our framework is highly flexible and advantageous; it allows us to compute the trust of CPS components separately, which may target hardware, software, or a network. It is easier to quantitatively calculate trust.
- Several considerations were made in the composition of the formula to determine the degree of trust. The first degree addresses overall system safety and security, whereas the second degree focuses on system trustworthiness.
- Promote a buildup of confidence in the CPS from the beginning of design.
- Applied to all CPS domains and not linked to the proprietary measurement tools.

- Configurable: Depending on user, organizational, or stakeholder demands, the necessary characteristics can be added or removed.

4.3.2 Phase Two: Modeling trust in CPS

When the cyber-physical system meets the requirements of availability, confidentiality, integrity, security, and safety, it is considered a trust system. Nonetheless, an adequate analysis of these systems is lacking. The current research offers insufficient direction for a methodical process or modeling language to facilitate this type of trust-based examination. The two most pressing issues are achieving trust by default in CPS and systematically incorporating confidence modeling into the creation of systems from the beginning of the system life cycle. Model-based system engineering (MBSE) techniques can be used to solve problems in system trustworthiness design.

In [167], we suggested an effective and workable MBSE technique for creating trust in CPS. It is advised that businesses employ MBSE and must implement their methods. Model-Based System Engineering (MBSE) is a defined approach that uses modeling to support system requirements, analysis, design, validation, and verification (V&V) operations. It begins at the initial design stage and continues through the implementation and future lifecycle phases. The MBSE technique must be used because of the complex, multidisciplinary, and multi-domain nature of the CPS process.

A common use of MBSE is to: a) capture and manage a system's requirements, architecture, design, and identification of its environment. MBSE has emerged as a crucial component in the design of complex cyber-physical systems [168,169,170]. b) Participate in and offer perspectives for various reasons to facilitate communication among numerous stakeholders.

According to [171], system validation and verification during the first phases of system design is an advantage of MBSE activities. Architecture models must incorporate stakeholder needs to comply with MBSE. Because designing a trust CPS technically requires a variety of disciplines, as illustrated in Figure 4.2, which details the phases, activities in phases, and employed methods, our suggested MBSE method for the trust cyber physical system is based on several disciplines, including requirement definition, modeling, and verification. System engineering serves as the basis for the MBSE.

4.3.3 Phase Three: EMAS Test Model

In phase one, trust quality was ranked on three levels: secured, trustworthy, and trusted. Based on these three levels, we organized the test model, which is divided into three levels. All set properties are defined in phase one. Degree one contains mandatory and optional properties; degree two also contains mandatory and optional attributes. The third degree contains the mandatory attributes of degrees one and two in addition to the other properties of trust related to degree three.

We are faced with numerous properties and sub-properties of CPS, some of which are mandatory and others are not. Some of these relate to safety and others to security, usability, and other attributes. How to verify and collect information and conduct analysis of each property, as well as how to evaluate each degree, test all properties, and determine whether the CPS is secure, trustworthy, or trusted. In this context, distributed AI approaches, such as those based on embedded multi-agent systems (EMAS), are essential for handling the analysis and verification of CPS properties. Based on this, this study uses an EMAS approach to design cyber-physical system properties that can embed different data analysis capabilities, supporting the verification of CPS trustworthiness. These concepts were applied to CPS properties for multi-degree trust evaluation and testing, where different kinds of data analysis were performed in property verification and cooperative agents disposed along secured CPS, trustworthy, and trusted levels, and for each mandatory property is allocated an agent, and for each test degree or level, an agent test is associated with each level in the model, as presented in Figure 4.3.

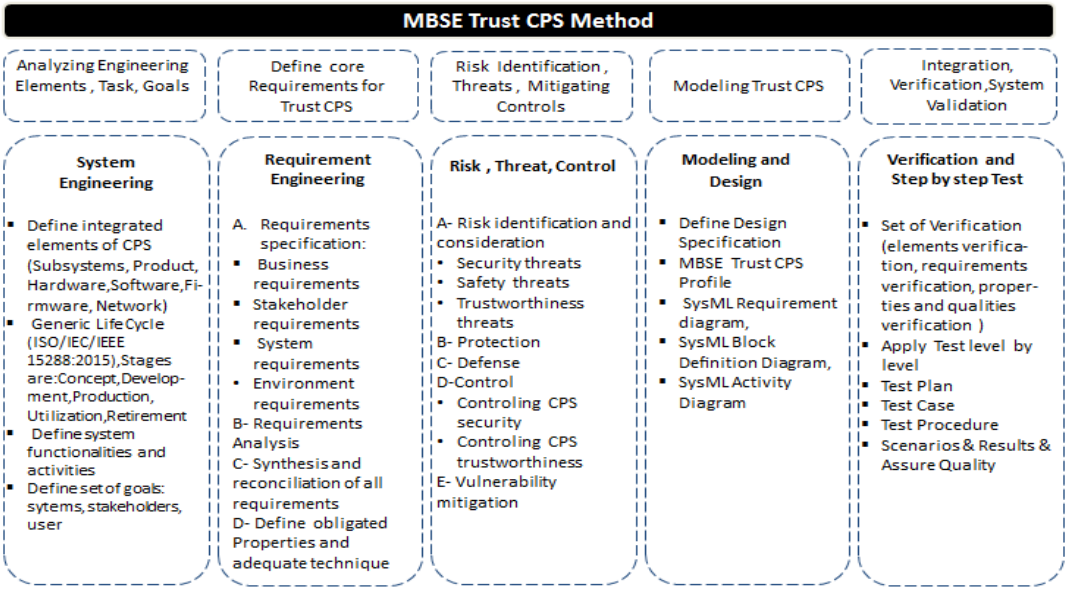

Figure 4.2 Model-based system engineering (MBSE) phases, activities within phases, and methods employed [167]

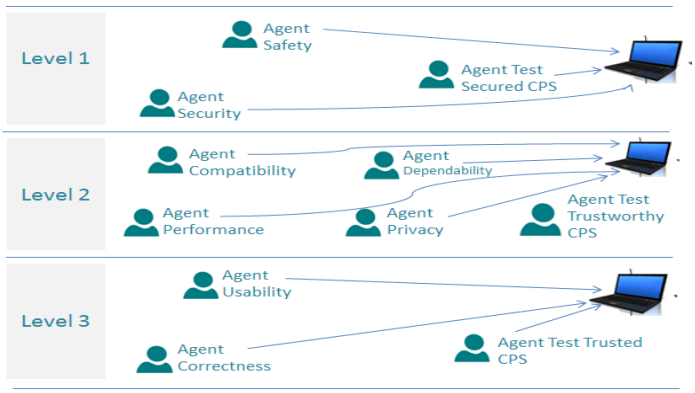

Figure 4.3 EMAS TEST Model Levels

4.3.3.1 Embedded multi-agent system (EMAS)

An embedded multi-agent system (EMAS) is an embedded system that has a collection of agents that can be software or hardware. EMAS interacts with its real environment through physical communication or actions. The organizational model of the embedded agent is shown in Figure 4.4.

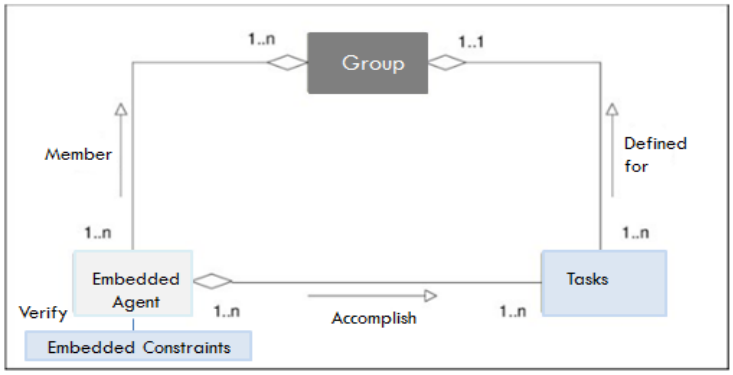

Figure 4.4 An embedded agent's organizational model

Embedded agents are the components of our testing system. Agents can work independently, allowing maximum flexibility and extensibility. A test task can be divided into several smaller tasks that can be completed directly by an agent. Agents are also responsible for decomposing testing work. Multiple agents may have the same functionality, but can be tailored to deal with different information types. Testing activities in the EMAS focus on

1) *Testing the trust in CPS with agent property.*
2) *Testing the trust level and the agent's reputation by testing the functionality of agent property with the agent test level.*

The verification of each trust property is provided by an agent test for the property. This test is for the quality of the CPS. The verification of trust in an agent within the EMAS test is about testing the reputation of the agent property: is this agent trusted by its agent supervisor, which is the agent test level or no? An agent's capacity to recognize and assess the skills of others, as well as the knowledge it possesses, determines how much trust it can place on other agents [172]. The prevalent approach to evaluate trust in MAS is reputation based [173]. In the same context, evaluation of concept-based reputation involves assessing other agents' trustworthiness by drawing on a variety of data sources, including firsthand experience and witness accounts. In [174], they introduced the well-known model of Trust and Reputation in the Context of Inaccurate Information Sources (TRAVOS), which mostly falls under the evaluation component. The set of agents is combined into three levels and mentioned by mandatory properties, such as those determined in the trust degree framework [155] as follows:

Level One:

- ***Agent safety**: Agent allocated to safety property for evaluation and testing.*
- ***Agent security:** Agent allocated to security property for evaluation and testing.*
- ***Agent Test Secured CPS:** Agents are allocated to test level 1, which is secured CPS. In addition to testing the functionality of agent safety and agent security.*

Level Two:

- ***Agent compatibility:** Agent allocated to compatibility property for evaluation and testing.*
- ***Agent performance:** Agent allocated to performance property for evaluation and testing.*
- ***Agent dependability:** Agent allocated to dependability property for evaluation and testing.*
- ***Agent privacy:** Agent allocated to privacy property for evaluation and testing*
- ***Agent Test Trustworthy CPS:** Agent allocated to test level two, which is trustworthy CPS. In addition to testing the functionality of the agent (compatibility, performance, dependability, and privacy).*

Level Three:

- ***Agent usability:** Agent allocated to usability property for evaluation and testing.*
- ***Agent correctness:** Agent allocated to correctness property for evaluation and testing.*
- ***Agent test Trusted CPS:** Agent allocated to test level three, which is trusted CPS. In addition to testing the functionality of agent usability and agent correctness.*

4.3.3.2 Description of the test with the EMAS TEST

In phase two of the MBSE model for trust CPS [175], a testing strategy is defined as the verification phase in the MBSE model. This phase presents a series of verifications (element verification, requirement verification, attribute verification, and quality verification) that should be completed and executed concurrently with the previous stages of the MBASE model. As shown in Figure 4.5, a test is conducted step-by-step to guarantee the quality of modeling and level-by-level verification to guarantee the design of the entire system. The main goal is to carry out an additional analysis to determine whether the system meets the trust criteria and enables prompt feedback on requirements and design decisions.

The advantage of this test technique is that it performs verification early in the design process, ensuring that any errors in any step are fixed, and that any design problems are addressed on time. Not waiting until the conclusion of the modeling and development cycle and then using CPS to find defects and bugs at the first level of conception. Delays in mistake detection lead to system failure, hazardous harm, and other risks.

We used this test technique in conjunction with an embedded multi-agent system, as shown in Figure 4.5. The EMAS TEST model considers the verification stages and translates them into tasks and rules for verification, as listed in Table 2.

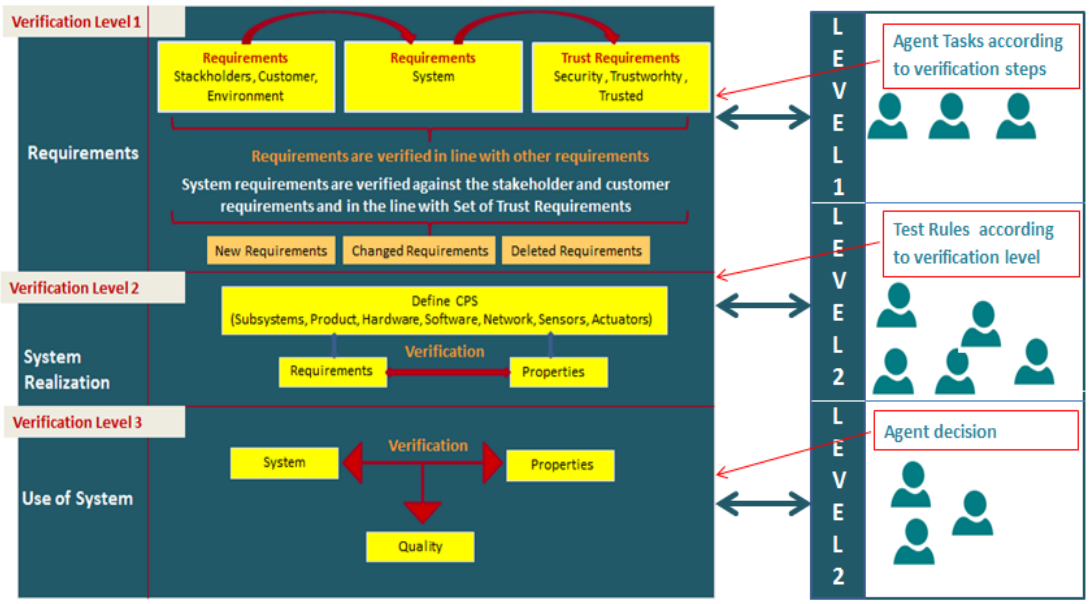

Figure 4.5 Description of the Test with EMAS

1- Verification Level One

This is the first verification pertaining to the early phases of CPS conception. Stakeholders define their requirements for CPS trustworthiness, as well as customer and CPS developer, environmental, and system requirements. The verification will carry out all the requirements from the actors and test them one by one until it results in a set of requirements that are appropriate for all actors. This step may result in a change or deletion of requirements.

Each agent in the EMAS TEST model will perform this initial requirement verification during the conceptual phase. This is the first task for each agent.

2- Verification Level Two

This second verification pertains to the realization phases of the CPS. The verification will carry out a test of the requirements designed in conception with the quality of the realized system, including the hardware, software, and network.

Each agent in the EMAS TEST model will perform this verification during or after the realization phase of CPS. This is the second task for each agent.

3- Verification Level Three

The third stage of verification focuses on the system usage. A combination of test plans, test cases, and scenarios is used. Assurance and certification can be provided to the tested CPS.

Each agent in the EMAS TEST model will perform this verification during or after the system usage. This is the third task for each agent.

In addition to the preceding verification level, another verification is associated with the third phase of MBSE for trust CPS [167].

4- Identification and Assessment of Risks

Threats that could jeopardize CPS operations should be controlled. This stage involves the identification of a group of resources, risks, or controls. Industries define rules for risk management and some businesses have their own protocols and requirements.

Each agent in the EMAS TEST model will perform this verification.

5- Verification of the availability of mitigation and protection

This verification specifically focuses on safety and security properties. Controlling CPS security and safety requires the following defense mechanisms: a) prevention, b) detection to identify the presence and source of an attack, and c) isolation of corrupted subsystems and rapid restoration to a normal state.

Agent Safety and security in the EMAS TEST model will perform this verification.

4.3.3.3 Agents tasks and verifications rules

Each agent in the EMAS TEST model assigns some responsibilities to ensure the verification of the CPS's trustworthiness. A set of rules ensures that verification occurs at all the stages and levels. Table 4.2 shows the tasks assigned to each agent as well as the verification rules for the previously described verification.

Table 4.2 The tasks assigned to each agent and the verification rules

Verification Objectives	Agent Task	Verification Rules
Requirement	**T1**: the Agent verifies the compliance of all actors requirement and announce that requirement are verified and decide that **R1** is **TRUE else FALSE**	**R1**:If all requirements are identified and filtered from all actors, agent will inform that the set of requirements for initial conception is verified
CPS Realization	**T2**: the Agent verifies if the requirements are met and announce that requirement are met in CPS realization and decide that **R2** is **TRUE** else **FALSE**	**R2**:If the requirement are verified during realization of CPS or after, the agent will announce that the requirement are meet
CPS Usage , Certification	**T3**:the Agent verifies if the CPS is used by user or standardized or certified and announce that CPS is used or certified for this property and decide that **R3** is **TRUE** else **FALSE**	**R3**:If the CPS is used and tried within many scenarios, integrate standards, or certified, the agent will announce that CPS is standardized , or certified in this property.
Identification and Assessment of Risks	**T4**:the Agent verifies if the risk are assessed and announce that risks are assessed and decide that **R4** is **TRUE** else **FALSE**	**R4**:If the risks and threats surrounding CPS are assessed, the agent will announce that risks in CPS are assessed
Verification of availability of Mitigation, Protection	**T5**: the Agent verifies the existing of defensive strategy or tools integrate within CPS and announce that CPS is mitigated and decide that **R5** is **TRUE** else **FALSE.-**The agent verifies all the rules, if **R1,R2,R3,R4,R5** are VRAI, and send message to Agent test level that the propriety Pi is verified positively, Pi=1 else Pi is verified negatively and Pi=0.	**R5**:If a mitigation tools or defensive methods integrated within CPS, the agent will announce that CPS is protected
Verification decision	**T6:-** the agent test level verify the functionality of related agent property in level.**-**the Agent test level one send message to Agent test level two that contain **D1** decision if CPS is secured or not. Same task : the Agent test level two send message **D2** decision if CPS is trustworthy to Agent test level three	

The set of rules specifies that an agent acts and completes his tasks according to a verified rule. An example is agent security in his first task, which verifies the set of requirements for the security attribute to determine if it meets the needs of all actors or not. The rule is that if all requirements are identified and filtered by all actors, agent security will decide that the set of requirements for the initial conception is verified.

The proposed list of verification rules relates to the necessary steps of the test designed in our proposed MBSE for trust CPS and ensures the test over the entire cycle development of CPS. The proposed test model can be used for all the phases of the CPS development cycle. If the test model is used for a realized CPS and there is no take-hand test before conception, the verification rules related to the conception phase R1 will be considered verified and true. The same is true for the verification rule R2, which is considered verified and true.

4.3.3.4 Trust verification process

The verification process was divided into three degrees of trust: secured, trustworthy, and trusted. The verification process for degree one is presented in Figure 4.6. The verification procedure for this degree focused on two properties: safety and security. The agent's safety qualities were tested by completing a series of tasks (T1, T2, T3, T4, and T5) and confirming a set of rules (R1, R2, R3, R4, and R5). The same processes for security properties.

Rule checking was performed using an embedded checking algorithm for agent safety. Each agent's property (safety, security, compatibility, privacy, etc.) contains the same embedded checking algorithm for verifying the property, which can be evaluated and sent here as the value of the verified property to the agent test level. This aids in checking the set of properties related to the level and determining the degree of trust. The agent test level (agent test-secured CPS, Agent test-trustworthy CPS, and Agent test-trusted CPS) also embeds a checking algorithm specified for the test level.

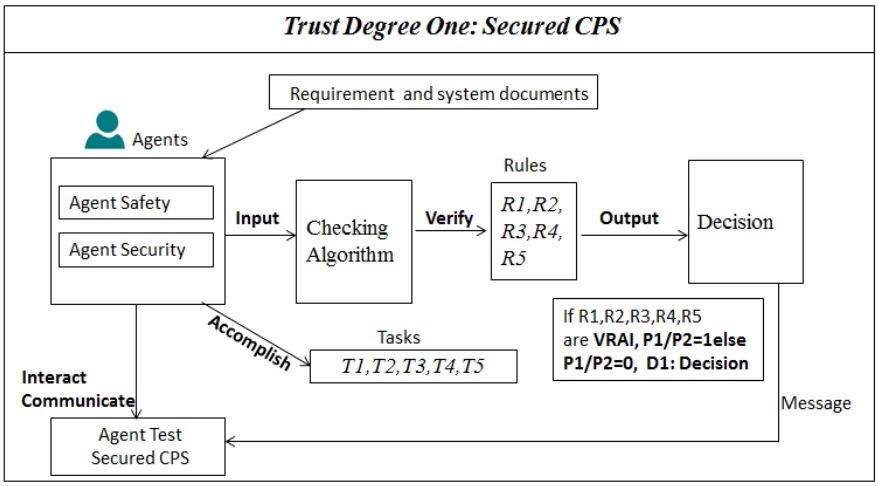

Figure 4.6 Verification process for degree one

The verification process for degree two begins after degree one has been completed and the decision that CPS is secured. The verification procedure for the trustworthiness degree focuses on four properties: compatibility, performance, dependability, and privacy. The verification process was the same as that defined for degree one. Figure 4.7 presents the verification process for degree two.

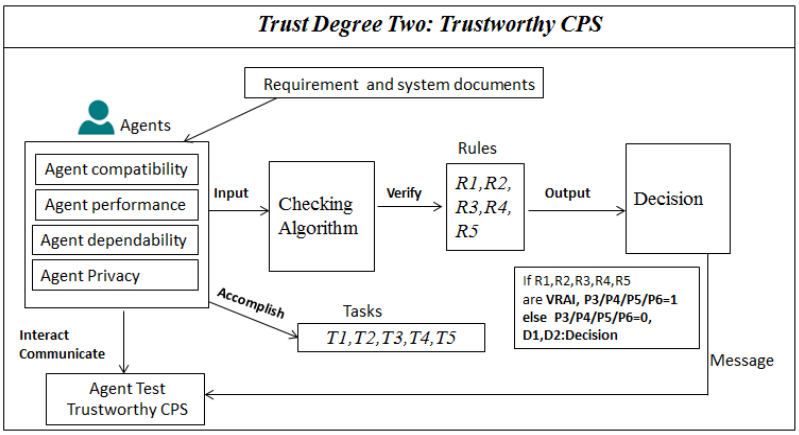

Figure 4.7 Verification process for degree two

The verification process for degree three begins after degree two has been completed, and the decision is that the CPS is trustworthy. The verification procedure for the trusted degree focuses on two properties: usability and correctness. The verification process was the same as that defined for degrees one and two. Figure 4.8 presents the verification process for degree three.

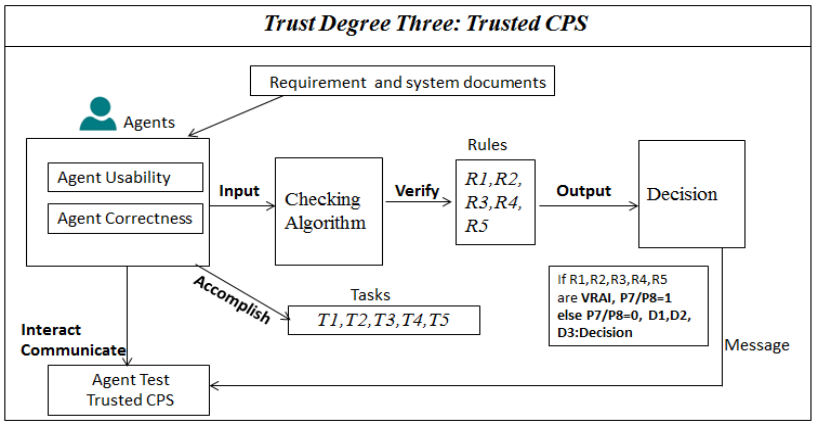

Figure 4.8 Verification process for degree three

4.3.3.5 Embedded constraints in EMAS TEST Model

The most difficult challenges to address and that require explicit representation of time are: 1) Responsiveness, 2) Termination (liveness), and 3) Resilience. Punctuality (timeliness). This aspect is relatively little covered, probably because applications using embedded EMAS are still too few in number.

***Reactivity**: Embedded systems must continuously respond to their surroundings.*

***Liveness:** The agent needs to avoid deadlock. The system must be able to adapt its behavior even in the event that it provides a partial or unsatisfactory response while still taking timeliness into consideration.*

***Timeliness:** Time control is essential for embedded systems, and timely responses are required.*

Reactivity and punctuality (timeliness) are the most challenging and hard to treat. Elmenreich [176] argued that real-time capabilities cannot be guaranteed reliably owing to the weak coupling of agents, which are generally asynchronous. The possibility of using MAS in the case of embedded systems that are highly constrained by time is challenging. However, very early on, Occello and Demazeau [177] evaluated the impact of real-time aspects on the design of agents and showed that they must be considered for each agent's abilities and for each level of the design. Therefore, we consider these points in the development of the EMAS Test.

The embedded agent in the EMAS TEST Model should accomplish its tasks within a constrained time with respect to the completion of tasks with the other agent. Therefore, it is necessary to consider the constraints of the embedded MAS. In our case, we will take into consider only real-time constraints, which are accounted in the accomplishment of tasks by agents and responses in time and in interaction and communication with other agents. The real-time constraint consideration in the EMAS Test Model is shown in Figure 4.9. The real-time constraints for interaction in the EMAS TEST process are presented in Table 4.3. The embedded agent in the EMAS TEST can be considered a real-time agent, and according to [178], a real-time agent (RTA) is an agent whose responsibilities are temporally constrained. An ATR (real-time agent) is an agent comprising a sequence of tasks, some of which have time constraints. For these agents, it is also necessary to consider the temporal precision, which is expressed by a set of temporal constraints imposed by the environment. For the tasks of the agent in the EMAS test, the communication is strictly real-time. Other tasks related to planning and software are flexible in real-time, as shown in Figure 4.9.

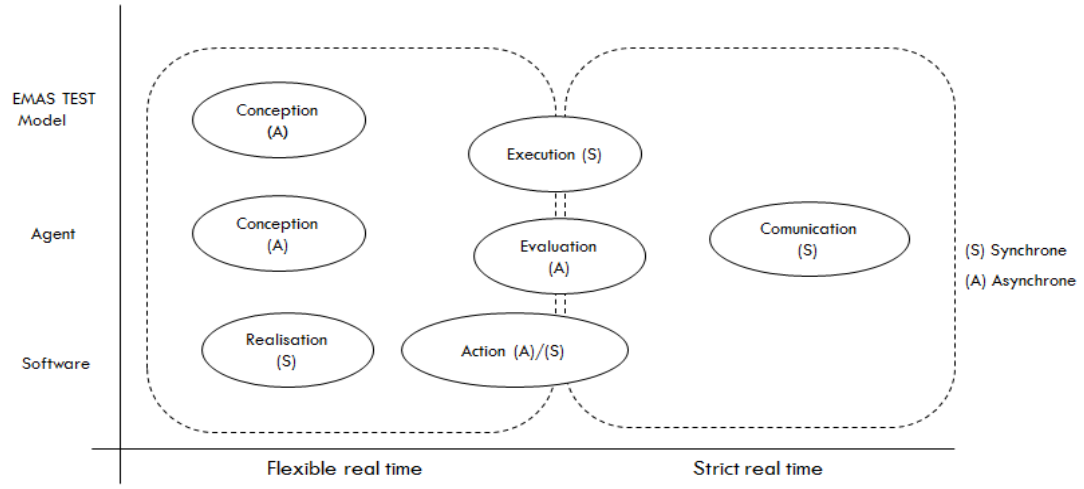

Figure 4.9 Real-time constraint consideration in the EMAS Test Model

Table 4.3 Table of constraints

Constraint	Definition
Task Accomplishment	The tasks of an agent should be achieved in constrained time
Task time	Each task should be achieved in the limited time
Interaction with other agent	Each agent should interact with its responsible agent and other agent for completing task
Interaction with tester and display result	- Some agent interact with tester in phase of input data -Some agent should display the result of test such as Agent test level and interact indirectly with tester

4.3.3.6 The interaction and communication via message in EMAS TEST

The agent test level will carry out a test of the related agent property at that level. The experiment consists of an agent test level that generates a number of messages to be sent to the agents under test. For example, an agent test-secured CPS sends a message to (agent safety and agent security, agent test level 2). These messages can be classified into five types.

- Messages requesting presence (used to check if an agent is active on the platform). The reply only verifies the agent's existence with a basic acknowledgement; the typical response time for a message of this nature was a few microseconds, provided the user was not engaged in any other cognitive activities.
- Requests messages (asking for accomplishing task): Verify the property and return a property value. The response to these messages is the solution, the result of the property test, and the value found for the given property. The agent takes more time to respond than to the presence request messages. The time required was approximately a few minutes.
- Response message (confirm the presence): The agent property sends a message of presence confirmation to its related agent test level.
- The response message (returns the value of the property and confirms that the property is verified).
- Message to the next level (test result of level): send a message to the next level to confirm that the actual level is verified and the result of the test is sent via decisions D1, D2, and D3.

There are two types of messages: message request presence (Pr) and message accomplish task (At). Table 4.4 shows the set of messages.

Table 4.4Table of interactions via messages

Agent test level One	Message	msg1to (agent safety)	Msg3 to (agent safety)	Msg5to (agent security)	Msg7 to (agent security)	Msg9 to next level(return test result)				
	Type	Pr	At	Pr	At					
	Response	Msg2(confirm presence)	Msg4(return test result)	Msg6(confir m presence)	Msg8(return test result)					
Agent test level two	Message	msg10to (agent compatibi-lty)	Msg12 to (agent compatibilty)	Msg14to (agent performance)	Msg16to (agent performance)	Msg18to (agent dependabil-ity)	Msg20 to (agent dependa-bility)	Msg22 to (agent privacy)	Msg24 to (agent privacy)	Msg 26 to next level(retu rn test result)
	Type	Pr	At	Pr	At	Pr	At	Pr	At	
	Response	Msg11 (confirm presence)	Msg13 (return test result)	Msg15 (confirm presence)	Msg17 (return test result)	Msg19 (confirm presence)	Msg21 (return test result)	Msg23 (confirm presence)	Msg25 (return test result)	
Agent test level three	Message	Msg27to (agent Usability)	Msg29to (agent Usability)	Msg31to (agent correctness)	Msg 33to (agent correctness)	Msg 35 to next level(return test result)				
	Type	Pr	At	Pr	At					
	Response	Msg28 (confirm presence)	Msg30 (return test result)	Msg32 (confirm presence)	Msg34 (return test result)					

4.3.3.7 Checking Algorithm

The verification process in the EMAS TEST model is based on a checking mechanism that consists of two fundamental components: checking the rules and evaluating CPS trustworthiness.

1. *Checking the rules*

The verification rules and agent tasks are presented in Table 4.2. The checking of rules involves carrying out the accomplishment of tasks by agent propriety with consideration of rules. The total number of rules is five. Each rule is related to the verification objectives and levels. The agent property verifies whether R1 (if all requirements are identified and filtered from all actors) agent will inform that the set of requirements for the initial conception is verified, that is means R1 = TRUE, and so on for the other rules. If the total number of rules is verified and all Ri are true, the agent property will announce that the property is verified positively, and Pi = 1. This check led to the evaluation of metrics.

Rules : Rk , k=1,5
Tasks : Tn, n=1,5
Property: Pi, i=1.8
Decision: Dj, j=1,3
Agent property (Pi), n=1, 8 / the number of mandatory property for trust is 8

2. Evaluation of CPS trustworthiness

The trust CPS equation was introduced in [155]. We used a simple formula when a set of measurements contained shared variables for each degree.

We anticipate expressing a quantifiable view of the guarantees of security and trust qualities. We consider that each mandatory attribute will be rated as 1 if confirmed and 0 if not to reduce uncertainty.

The evaluation is beneficial in that if the evaluation criteria are already specified before the adoption of the technology, they will not only be used to measure the quality of the system during its use, but also to determine whether the envisaged system is capable of meeting the needs.

In the proposed verification process, we consider only the mandatory metrics for each degree: secured, trustworthy, and trusted. Each agent is assigned one mandatory metric for each degree, as mentioned in the EMAS TEST model level in Figure 4.4. The roles of the agents are listed in table 4.5.

Table 4.5 the roles of Agent

Agent	*Role*	*Agent Type, description*
Agent propriety	- *Checking the verification Rules* - *Evaluating the proprerty* - *Announce the property's value to Agent test level*	*Reactive agent* *- There is no explicit representation of the environment.- No memory of their past.- Simple agent (stimulus/reaction operation.- Large number of agents.*
Agent test level	- *Verify the functionality of related agents that carry out the propriety test (verify their presence and their accomplishment of the task)* - *Verify and evaluate the value of trustworthiness of degree and make decision* - *Announce the decision to the next level of test*	*Reactive agent* *- There is no explicit representation of the environment.- No memory of their past.- Simple agent (stimulus/reaction operation.- Large number of agents.*

Each Agent has an embedded checking algorithm that performs the duties listed in Table 4.5.

Embedded checking algorithm: Agent property

∀ P ∈(Pi)(set of property)
∀ T ∈(Tn)(set of Tasks)
∀ R ∈(Rk)(set of verification rules)
Enter RQ,RS,PD
For k=1,5 ; (5 the total number of rules)
For n=1,5; (5 the total number of tasks)
If RQ=1 then R1= TRUE AND R2=TRUE
If RS=1 then R4= TRUE
IF PD=1 then R3= TRUE and R5= TRUE
If Rk ⟵TRUE then Tn is achieved and Tn ⟵TRUE;
k ⟵k+1; n ⟵n+1
if (R1 AND R2 AND R3 AND R4 AND R5) then Pi=1
Return "The property is verified positively"
Agent property send message to Agent Test Level (property Pi is verified positively and Pi=1)
Else Pi=0
Return "property Pi is verified negatively"

Each Agent Test Level has an embedded checking algorithm 2 that performs the duties listed in Table 4.5.

Embedded checking algorithm: Agent Test Level

Agent Test Level 1
∀ P ∈(Pi)(set of property, the total number of property is 8)
For i=1,2
If Pi=1 then D1= P1+P2
D1=2
Return (the level one is verified and the system is secured)
Agent test level 1 send a message to the Agent Test Level2 (The level one is secured and D1=2)
Else Return "System is not secured" and send message(the system is not secured)
Agent Test Level 2
In this stage we consider that the previous level is verified positively and system is secured
For i=3,6
If Pi=1 then , D2= D1+P3+P4+P5+P6
D2= 6
Return "the level two is verified and the system is trustworthy"
And Agent Test Level 2 send a message to the Agent Test Level3 (The level two is trustworthy and D2=6)
Else Return "the level two is not trustworthy" and send message(the level two is not trustworthy)
Agent Test Level 2
In this stage we consider that the previous level is verified positively and system is trustworthy
For i=7,8
If Pi=1 then , D2= D2+P7+P8
D3= 8
Return "the level three is verified and the system is trusted)
Else Return "the system is not trusted"

3. Algorithm efficiency

An algorithm is considered efficient if its resource consumption is below the acceptable level. It should run in a reasonable amount of time on an available computer or according to hardware specifications. The two most commonly used measures are as follows:

- Time: how long does the algorithm take to complete.
- Space: how much working memory is needed by the algorithm?

Complexity analysis of Algorithm

Analysis of time complexity: The running time depends on the compiler used/R/W speed to memory and disk/ machine architecture: 32 bit-64/Input size (rate of growth of time).

To analyze the time complexity, we can take two approaches: estimation of running time and order of magnitude, which provides a general idea of performance. Estimation of the running time

a) Operations counts- select operations that are most frequently and how many time each done.
b) Step counts: Determine the total number of steps, possibly the lines of code executed by the program.

In our case, we have two algorithms, one for the agent propriety and one for the Agent Test Level.

1) Algorithm Agent Property

For n=1,5;----------n=5
If RQ=1 then R1= TRUE AND R2=TRUE---------3
If RS=1 then R4= TRUE----------2
IF PD=1 then R3= TRUE and R5= TRUE--------3
If Rk← TRUE then Tn is achieved and Tn ← TRUE;-----2
k ← k+1; n ← n+1----------2
if (R1 AND R2 AND R3 AND R4 AND R5) then Pi=1------2
Return "The property is verified positively" ----------1
n(3+2+3+2+2)+3 --→12n+3, n=5 , the total time is 12*5+3 --→63 unit of time

2) Algorithm Agent test level

Agent Test Level 1
For i=1,2 ----------2
If Pi=1 then D1= P1+P2----------2
If D1=2 ----------1
Return (the level one is verified and the system is secured)----------1
4*2---→ 8 unit of time
Agent Test Level 2
For i=3,6 ----------4
If Pi=1 then , D2= D1+P3+P4+P5+P6-------2
If D2= 6----------1
Return "the level two is verified and the system is trustworthy"------1
4*4→16 unit of time

Agent Test Level 2
For i=7,8 ----------2
If Pi=1 then , D2= D2+P7+P8----------2
D3= 8----------1
Return "the level three is verified and the system is trusted)------1
2*4 ---→8 unit of time

4. The advantages of algorithms

The proposed algorithm is particularly relevant to the EMAS TEST model, which is related to the trust degree and results in the verification of trust in CPS with a focus on trust computation to provide a quantitative view of trustworthiness. It includes three processes for determining and verifying confidence in CPS: test degree one (secured system), test degree two (trustworthy degree), and test degree three (trusted system). The algorithm is simple and includes simple variables. A simple test and verification instruction with a set of established verification rules and an agent task list was used in the test model. The first algorithm is designed for agent properties that share the same method of testing, with the distinction of which propriety the agent is allocated to, and the second type of agent is the agent test level. Three agent test levels focus on verifying the truthfulness of each degree. These algorithms were embedded in the agent.

4.3.3.8 Agent characteristics

The incorporation of agents in the design of the CPS propiety enhances the manner in which each agent tests the property and exchanges information with the tester of a property collection related to the degree of trustworthiness. An agent possesses several characteristics [180, 181], with the main characteristics being:

Autonomy: the ability of an agent to function autonomously with minimal human involvement.

Temporal continuity: requires that an agent operate constantly.

Social skills: an agent must be able to interact, communicate, and display certain social skills.

Pro-activeness: An agent reacts to its surroundings and pursues its objectives. The agent can respond to the inputs in its environment and use planning procedures.

Adaptability refers to an agent's ability to change their behavior and use AI approaches to fulfill tasks autonomously. Other sub-features include learning and submissions.

Mobility: refers to an agent's ability to move, either logically or physically, such as moving between machines on a network, or allowing an agent to run on a single machine to be accessible remotely from another place via the Internet.

Collaboration: Cooperation with other agents and successful operations in a timely manner. An agent should coordinate with other agents by sending and receiving messages using different agent communication languages and aid in social activities, such as distributed problem solving.

Reactivity: The agents' ability to adapt to changes in their surroundings and perceive them.

4.3.3.9 EMAS TEST Conception

1. *Agent cycle life*

The FIPA standard details the life cycle of an agent [FIPA00023]. There are four states describing the behavior of an agent within an EMAS. An agent can be created. It then goes to the "initialized" state, from which an invocation can make it "active."

The other states, "blocked" and "waiting," represent specific statuses destined to return to the "active" state; "Waiting" and "Blocked" both represent wait states. The life cycle of an agent according to the FIPA standard is shown in Figure 4.10.

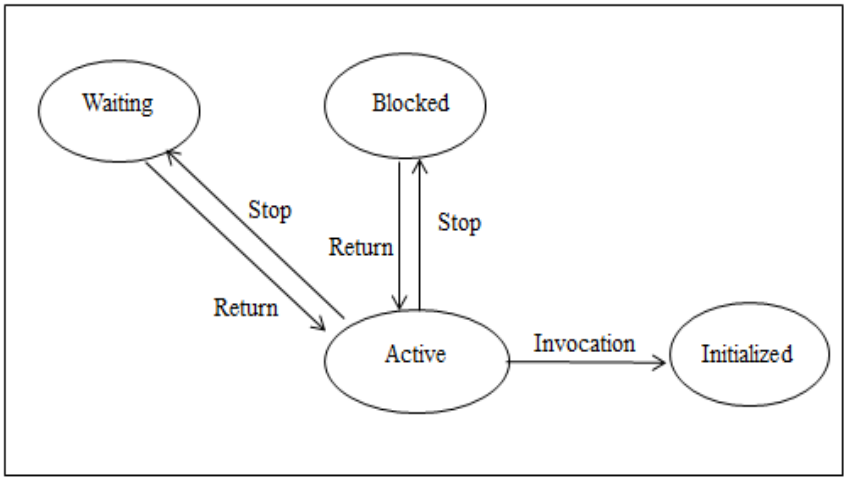

Figure 4.10 The agent life cycle

In embedded and ubiquitous computing systems, agents are the essential components. They are especially crucial in mission- or goal-oriented settings. In our case, the essential role of agents in EMAS TEST is to verify trust in the CPS. Goal-driven is generally a feature of agents, especially in our TEST EMAS model. We used the composite structure diagram and stereotype to define the goal-driven aspect of an agent. The UML 2.1 composite structure diagram illustrates an agent's internal organization, as well as the points at which it interacts with other agents in the system. It displays the arrangement and connection of the components that work together to carry out the function of the contained classifier.

For composite structure-based "Agent" stereotype schematic, it needs to have a name (in our example, Agent property, "property" which is the target to test), a component that manages the agent's efforts to accomplish a task (in our model, Agent test level), and at least one port that indicates that it is participating and role as presented in figure 4.11.

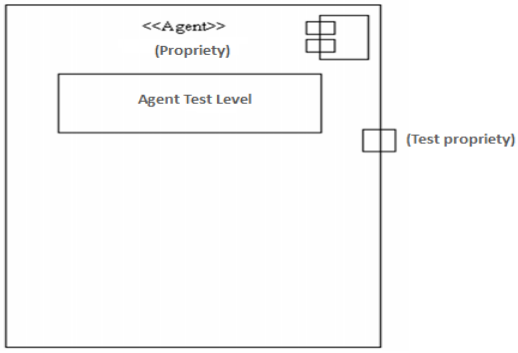

Figure 4.11 Definition of Agent

The goal-driven characteristics of the agent are presented in a composite structure diagram as presented in figure 4.12.

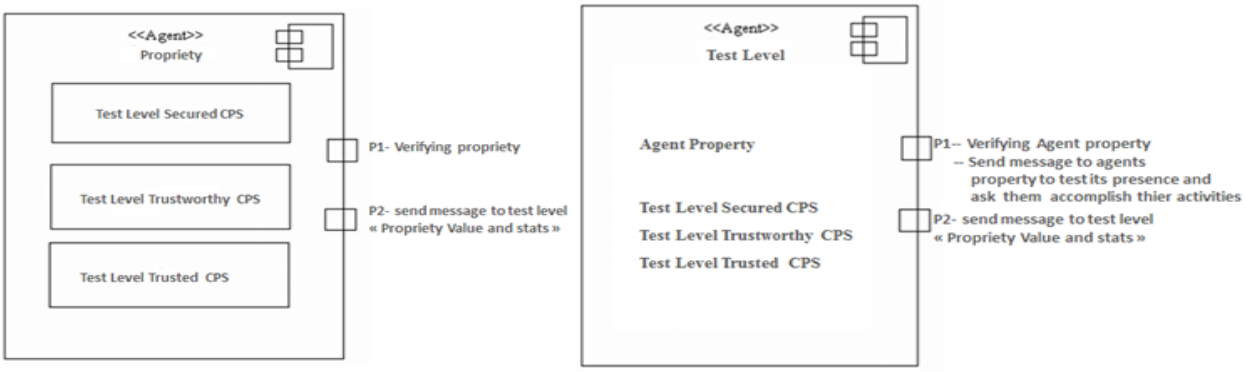

Figure 4.12 Composite structure diagram for agent goal driven characteristics

2. Requirements diagram for EMAS TEST Model

The SysML requirement diagram over functionality of EMAS TEST is presented in figure 4.13.

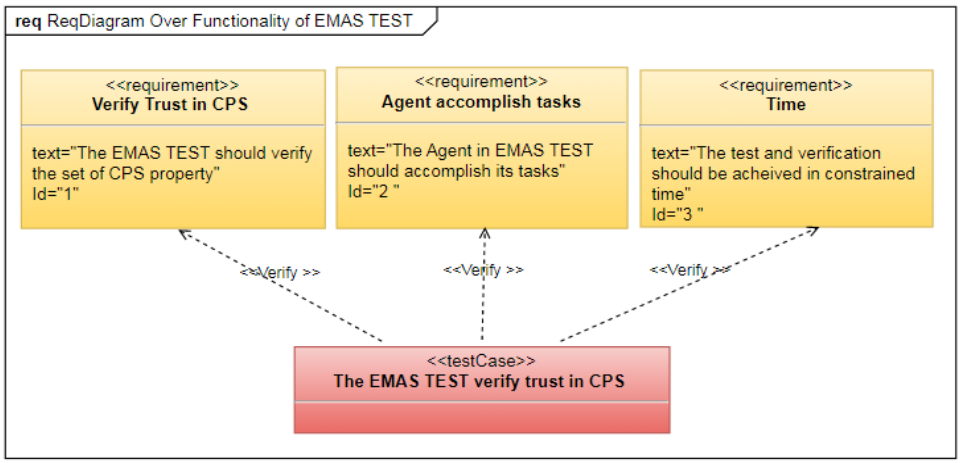

Figure 4.13 SysML requirement diagram over functionality for EMAS TEST

3. *Class diagram for EMAS TEST MODEL*

A class diagram for the conceptual level of the EMAS Test is shown in Figure 4.14. We added a package to bring together all the agents in the EMAS Test.

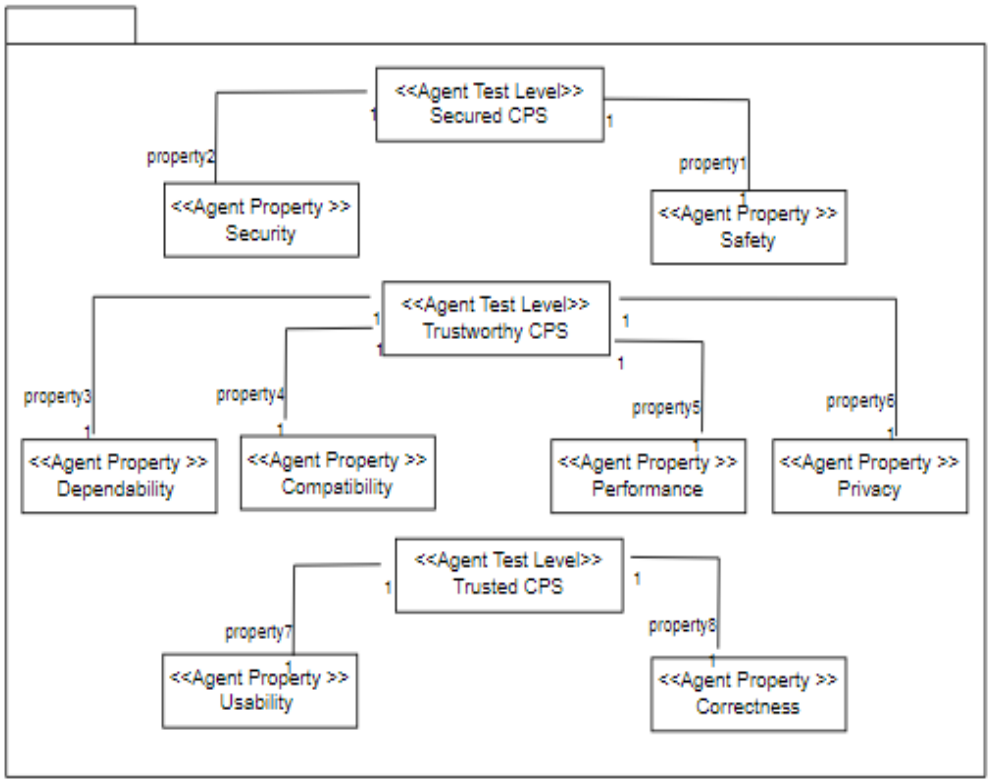

Figure 4.14 UML CLASS diagram for EMAS TAST (conceptual level)

4. *Use case diagram(Agent Tester)*

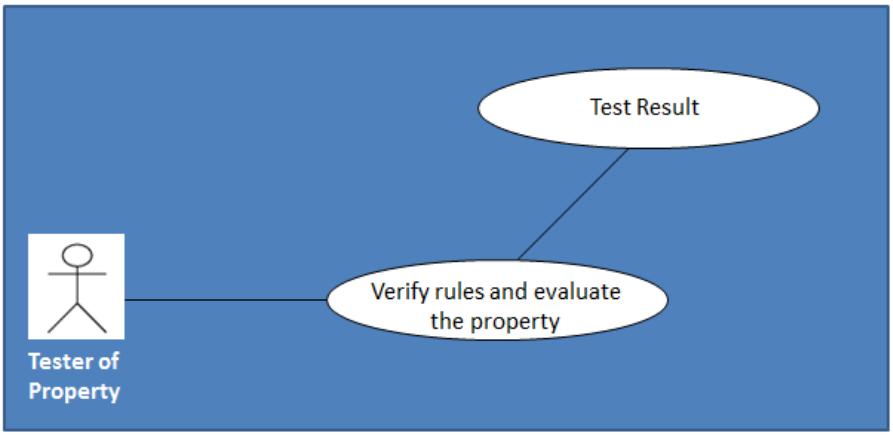

Figure 4.15 Use case diagram (Agent Tester)

The diagram in Figure 4.15 shows the steps of the test process, which begins by verifying the rule and ends by filling out the test results.

5. *Use case diagram(Test Operation)*

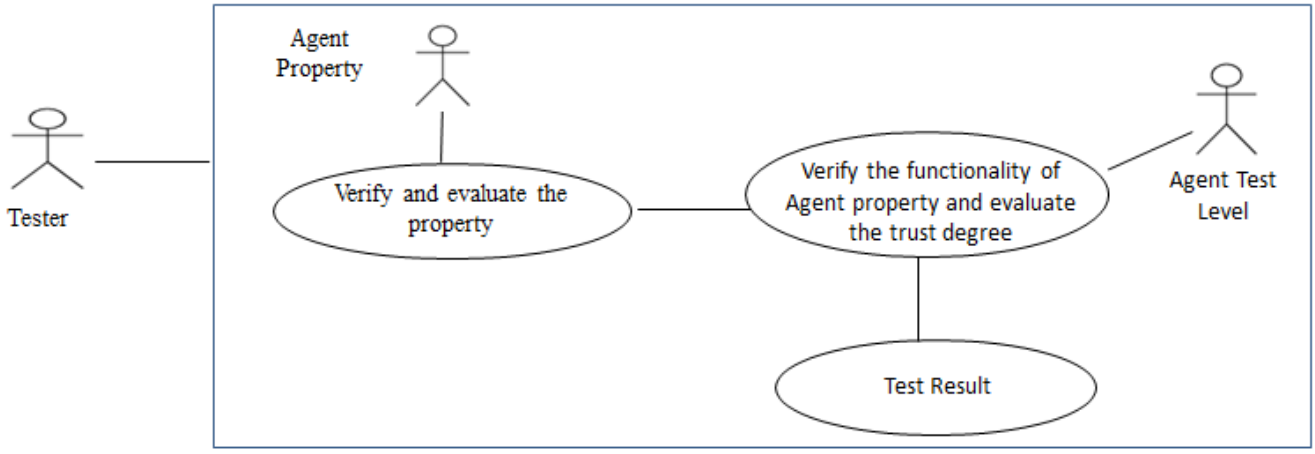

Figure 4.16 Use case diagram (Test operation)

The tester who uses the EMAS TEST is responsible for gathering the recommended data related to the CPS and verification rules. In figure 4.16, it presents a use-case diagram for the test operation. This diagram presents the roles of the agent property, which is the verification of the property, and the agent test level, which is responsible for testing the functionality of the agent property and testing the level and trust degree. The agent test level verifies the presence of the agent's properties and functionality. Each agent test property sends a result to the agent test level that relates. The agent test level evaluates the trust degree and determines whether the level of trustworthiness is verified, after which it explores the test results. The agent test level may be considered a tester and supervisor of the agent property.

6. *Sequence Diagram of interaction among Agents*

The sequence Diagram of interaction among Agents is presented in Figure 4.17.

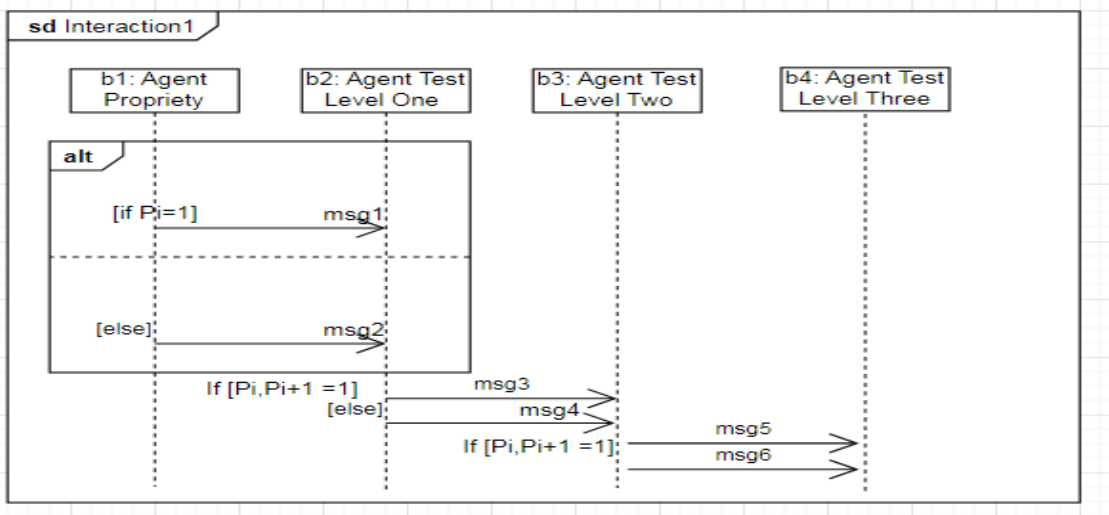

Figure 4.17 Sequence Diagram of interaction

4.3.4 Advantages of the EMAS TEST Model

- These test techniques using EMAS are strictly required if the system relies on a high throughput of elements such as CPS(sensors, actuators, network, hardware, and software) and (manufacturing chains with a large number of interacting items) and, therefore, should be deliberately tested.
- The EMAS test technique ensures a decentralized verification with an agent for each property and quality level. The agent verifies the property separately from the others, which improves rule verification and ensures test efficacy.
- The proposed EMAS TEST model ensures the testing of CPS in all the phases of the development cycle.
- The proposed test model was designed to test the hardware of the CPS, software, and network separately.
- EMAS allows engineers to readily model, simulate, and analyze any logistic processes. Applications vary from small factories to sophisticated industrial chains [182] and large supplier networks.
- The proposed test model is flexible and can test more proprieties and sub-properties. In this thesis, we focused only on the proposed mandatory proprieties. The set of mandatory proprieties depends on the industry and quality required by stakeholders.
- The proposed list of verification rules and the list of sub-rules related to a specific propriety may extend and contain more significant tests for enhancing the sense of verification and quality assurance.

4.4 System for testing with EMAS TEST

For the experimental part of this thesis, we chose the CPS in the field of oil and gas. For this topic of petroleum CPS, we contribute model-based system engineering for trust SCADA and ICS in [183].

4.4.1 Petroleum SCADA

In the oil and gas sector, SCADA systems are used to oversee and monitor offshore drilling, pipeline systems, wells, production, and processes [184]. Independent PLCs are administered by a software application bundle for human-machine interfaces (HMI), usually running on a personal computer, and they carry out I/O control tasks on field devices in a SCADA system [185].

The processed data can be stored on a PC using the SCADA HMI program for future verification. Table 4.6 presents the activities and equipment of the SCADA components.

Table 4.6 SCADA functions, equipment, and component parts

Components				
Input Field Devices	Output Field Devices	Remote Terminal/ Telemetry Unit	Wireless Medium	Human Machine Interface
Activity				
Transforming the physical parameters of a process into data that SCADA recognizes.	Affecting the SCADA output process.	Throughout the process, remote terminal units (RTUs) connect to sensors and convert sensor signals to digital data. It is made up of telemetry equipment that sends digital data to and receives digital orders from the supervisory system.	Wireless Microwave communication is used by field devices to connect to the PLC. Each field device uses an RTU to broadcast data using a Microwave Digital Systems (MDS) Remote Radio, which is directed toward an MDS Master. For radio communications, remote devices primarily use the encrypted Modbus Protocol. Encrypting MDS transmission ensures security.	Provide management information, forecasts, analytical data, and statistics in real time. Also, logistic data, scheduled maintenance methods, comprehensive graphic presentations for a particular piece of equipment or sensor, and professional troubleshooting support. The operating staff frequently sees data shown by the HMI system in diagrammatic and graphical form.
Equipment				
Pressure Transmitter (Transducer)	Pressure Regulators	Telemetry equipment	Field devices	Input devices
Temperature Transmitter	Block Valves	Sensors	PLC	Output devices
Differential Pressure Transmitter	Supervisory Switching and Solenoids.		Wireless Microwave	Graphical representation
Multi Variable Transmitter			Modbus protocol	HMI software
Gas chromatograph			MDS Master Remote Devices	
Flow computers			RTU	
Electro Volume -			Omniplexers	
Position Indicators for the Block Valves.				

SCADA system component diagram: The UML component diagram for the SCADA system is shown in Figure 4.18.

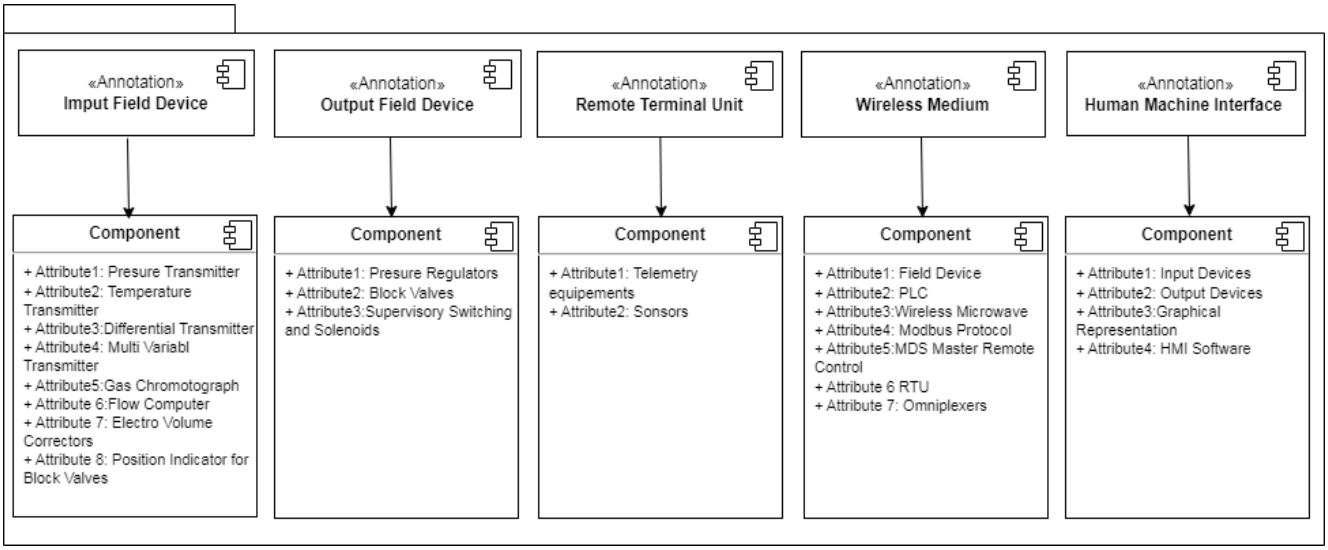

Figure 4.18 UML component diagram for SCADA systems [183]

Real-time data collection from field controllers is possible using SCADA. An interface human machine (HMI) was used to show this to operators in a graphical format. Remotely guided control procedures having real-time access to data, even from a distance, can help operators make the necessary adjustments at a particular site to maintain production. It makes use of (HMI) to facilitate communication with devices such as sensors and valves. It is possible to log and store historical data. This implies that manufacturers can track movements and analyze their production. Figure 4.19 shows the functionality diagram for SCADA.

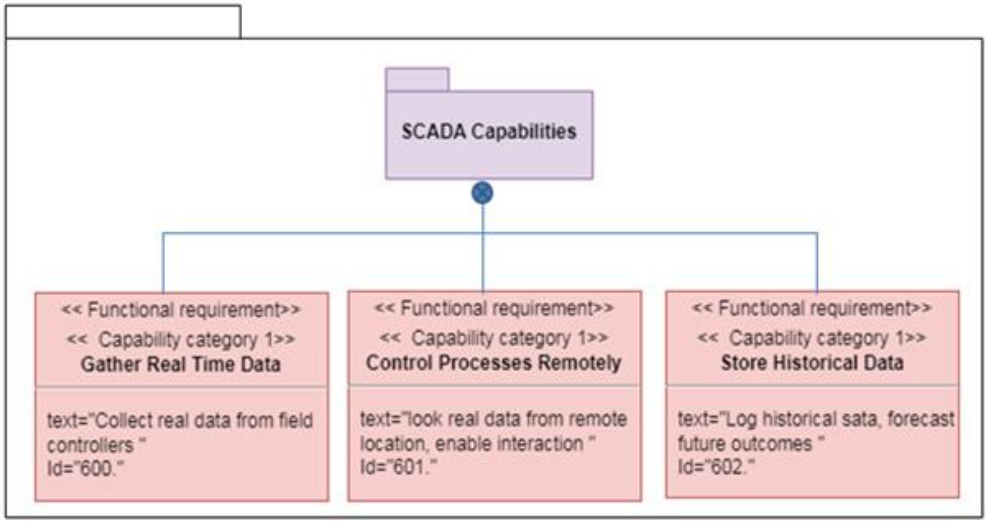

Figure 4.19 Diagram of SCADA functionality [183]

4.4.2 Petroleum ICS

Industrial control systems (ICSs) are groups of field equipment that are operated and controlled at a single location.

The layers that define ICS architecture are:

- The hardware consists of several components such as sensors, actuators, RTUs, PLCs, routers, smart cards, RFID readers, RACKs, and CPUs for the server, as well as valves, ATGs, and slaves.
- Software, such as HMIs, APIs, and proprietary software.
- Modems/routers, firewalls, and other communication protocols are all parts of a network.
- The Firmware layer contains operating systems, material management manuals, and AMIs.
- The Process layer led to the construction of control systems and the design of the ICS corporate logic.

An ICS system component diagram is shown in Figure 4.20.

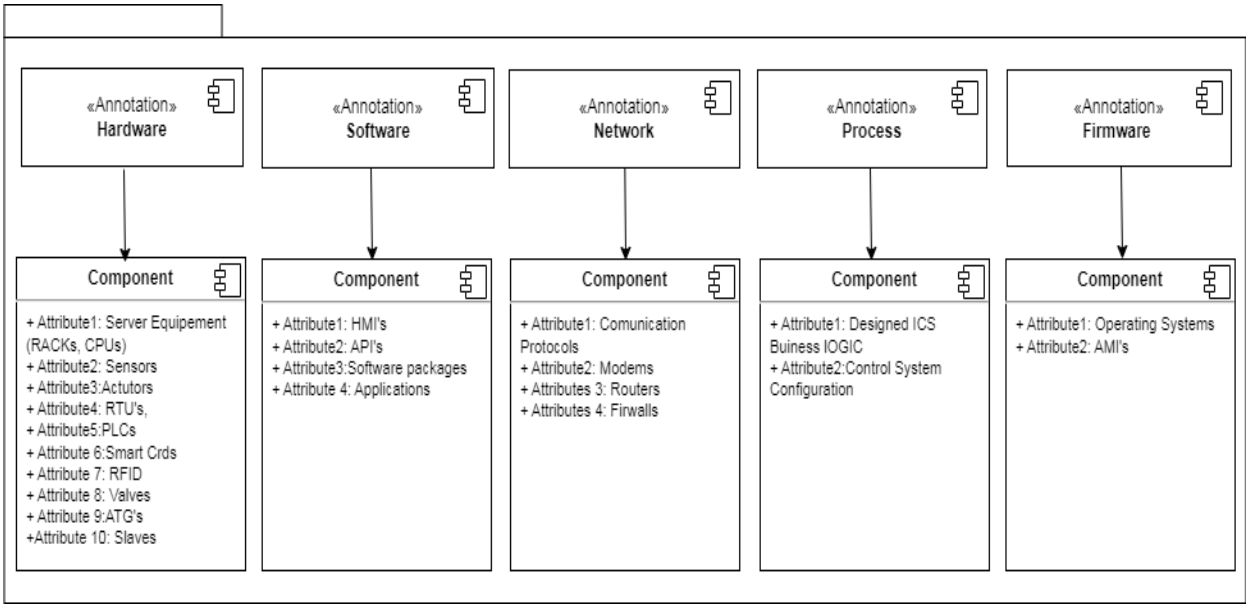

Figure 4.20 UML component diagram for ICS systems

Industrial processing information gathered at remote sites is transferred to a control center via wired and wireless connections using field equipment such as programmable logic controllers (PLCs), intelligent devices, and remote terminal units (RTUs). Clients can use the master terminal device to access data using standard protocols.

A human operator can view processed data on the human-machine interface (HMI) by gaining access to the time-stamped data collected by the data historian. The remote controllers received control commands after the data were collected and analyzed.

For engineering modeling, activity diagrams are useful because they provide information about the processes involved in system activities. A diagram of ICS activities is shown in Figure 4.21.

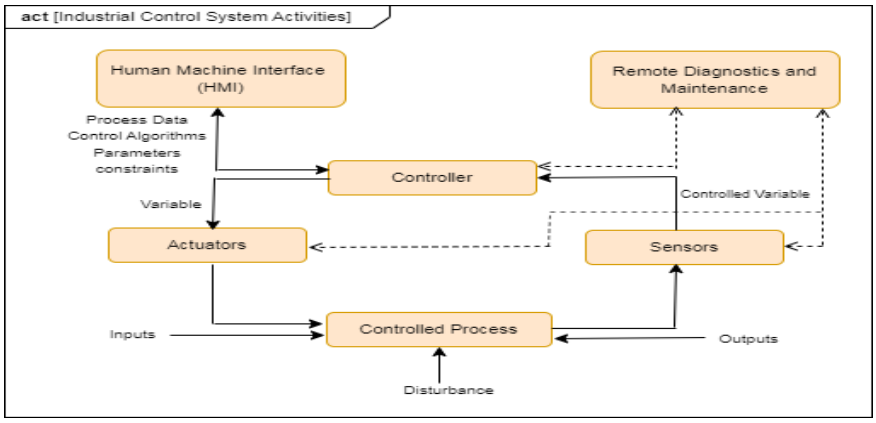

Figure 4.21 Diagram of ICS Activities [183]

4.4.3 Trust requirement for petroleum SCADA and ICS

Figure 4.22 illustrates the entire set of SCADA and ICS trust needs through the use of a SysML requirement diagram.

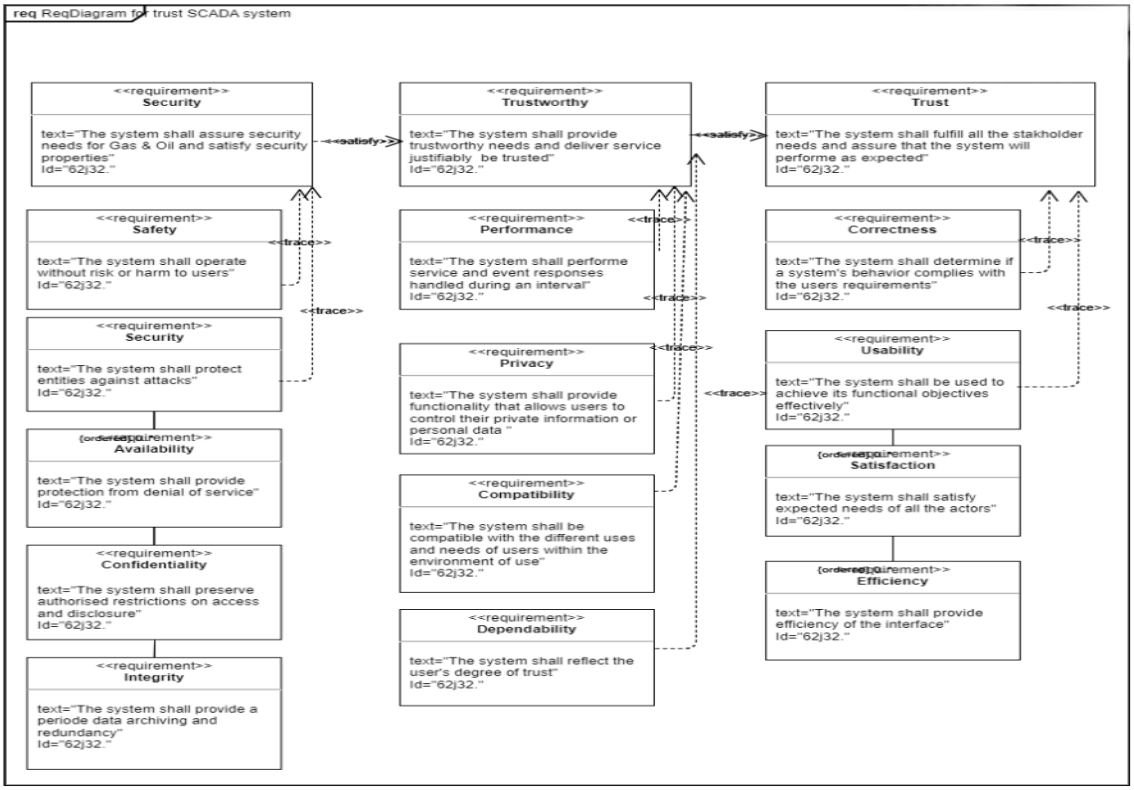

Figure 4.22 SysML requirement diagram for ICS and SCADA systems for the petroleum industry

4.5 Conclusion

In this chapter, we propose a strategy for testing trust in CPS. It starts with an analysis of trust quality in CPS as a first phase and modeling trust in CPS as a second phase. The deep comprehension of trust quality, which gathers many properties and sub-properties, aids in conceptualizing the cyber-physical system and reducing the complexity of the CPS aspect and quality within its environment. The suggested trust degree framework is highly useful and versatile; it enables separate computation of the trust of CPS components, which might be software, hardware, or a network. In addition, it makes the quantitative computation of trust easier and simpler. Foster the development of trust in CPS from early stage.

The second phase is a continuation of the first step and targets the modeling of trust quality in CPS with the actual and most recommended model, MBSE, which is the key tool for the development of complex systems and complex qualities, such as security and trustworthiness, and is widely used for collecting and controlling the system's requirements. In our proposed MBSE method, a verification phase is considered and presented step-by-step (element verification, requirement verification, attribute verification, and quality verification). This testing mechanism is combined with embedded multi-agents in the third phase, which targets the verification of trust quality in CPS. The methodological level, decentralized nature of EMAS, and analysis help to address the complexity of trustworthiness quality.

We have observed that certain aspects of the overall system must be captured and specified in the form of requirements (in the case of security and trust, which we specify in the form of a non-constraint functional). Other aspects, also directly linked to the specificity of ESMAs, must be considered during the analysis phase of our test model, such as the embedded constraints. In our case, we focus only on real-time constraints related to certain characteristics of the agent, such as timeliness.

A description of the EMAS TEST model is provided with details of the verification steps, agent behavior, and characteristics. In addition to the required algorithms, definition of verification rules, and task of the agent within the EMAS TEST, the embedded constraints of the EMAS Test are also defined.

The conception of EMAS TEST is detailed in the class diagram as well as the requirement diagram for the functionality of the model. The use-case diagram of an agent is presented, as is the use-case diagram of the test operation. Interaction and communication in the EMAS Test are vital activities and are described and presented via a sequence diagram of the interaction.

The advantages of the EMAS Test are highlighted and how the proposed strategy targets the gap in complex system verification with comprehensive and analytic tools and a variety of models is shown. These analytical and conceptual tools enabled us to cover the entire development cycle of CPS and to carry out its design and testing. The topic of the next chapter is how to use the proposed strategy, its approach, and models for testing CPS and its trustworthiness in a special industry.

CHAPTER 5

EMAS TEST IMPLEMENTATION

"Strong minds discuss ideas, mediocre minds discuss events, and weak minds discuss people"

Socrates

5.1 Introduction

In the previous chapter, we described our proposed strategy for verifying trust in CPS. In this chapter, we implement a verification strategy using CPS in the oil and gas industry. This industry drives the global economy. The oil sector is combined with Industry 4.0, and the most essential pillar is the cyber-physical system (CPS). Oil-related tasks have made use of cyber-physical systems (CPS), and CPS optimization techniques can help with the exploration, production, and management of petroleum resources. The process of verifying petroleum SCADA with EMAS TEST starts by modeling trust in SCADA and ICS with MBSE. The definition of trust requirements, as well as risk assessment and the establishment of mitigation in SCADA and ICS, are mandatory. The implementation of EMAS TEST for SCADA and ICS in oil and gas fields is presented in this chapter.

5.2 SCADA and ICS in the oil and gas industry

In the oil and gas sector, SCADA systems are utilized to monitor offshore drilling, pipeline systems, well operations, and production [184]. In an SCADA system, independent PLCs perform I/O control tasks on field devices under the supervision of a software application bundle for human-machine interfaces (HMI), which is usually run on a personal computer [185]. The processed data can be stored on a PC using the SCADA HMI program for future verification.

In an ICS, the control center receives industrial processing information gathered at remote sites via wired and wireless communications from field equipment such as intelligent devices, programmable logic controllers (PLCs), and remote terminal units (RTUs). The master terminal device allows clients to access data through established protocols.

The data historian stores time-stamped information accessed by the human-machine interface (HMI) to show the processed data to a human operator. Data processing and gathering are performed, and distant controllers receive the control commands. Figure 5.1 shows the ICS and SCADA components.

5.3 Modeling trust in SCADA and ICS with MBSE

There is a dearth of studies on essential design techniques in the petroleum and natural gas industries [186]. The design, testing, and maintenance processes do not fully incorporate security and trustworthiness. In the petroleum and gas sector, ICS and SCADA systems encounter challenges, such as increasing complexity, evolving attack methodologies, and difficulty predicting unanticipated threats.

Analytical modeling techniques foresee and mitigate hazards, in addition to reducing operational mistakes. Fostering safety, security, and trust in these technologies and in a complicated sector such as the oil and gas industry is mandatory.

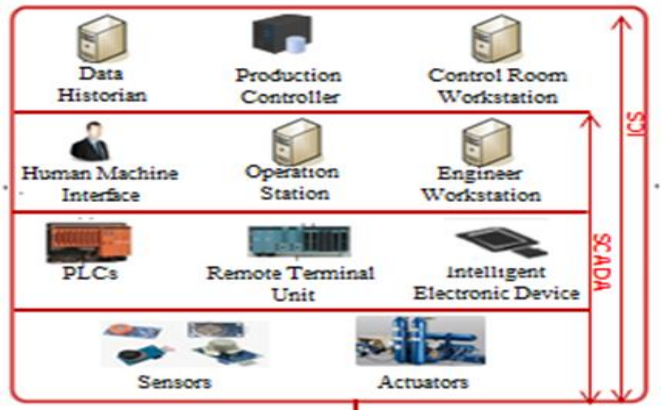

Figure 5.1 SCADA and ICS components

In [183], we covered the concepts of ICS and SCADA systems as well as the Model-Based Systems Engineering (MBSE) methodologies used in the design of modern ICS and SCADA systems for the oil and gas industry. Together with an examination of the MBSE phases and activities in stages, an overview of the steps involved in creating MBSE for trust in SCADA/ICS is provided. An SysML/UML-based profile, trust requirements definition, risk and threat assessment, user interaction modeling, and control and mitigation strategies are all included in the MBSE approach.

In the previous chapter, we explained that anticipating trust quality in the design of complex systems is critical, resulting in a lower percentage of system failure, and that including tests in the system's conception phase is also advantageous.

The EMAS Test Model was described in the preceding chapter. The technique, which is based on an embedded multi-agent system, is provided, as is the test procedure, including its parts and levels. This chapter intends to apply the EMAS TEST concept to ICS and SCADA systems in the oil and gas industry. As presented in the previous chapter, the EMAS TEST approach depends on some phases in the MBSE model and is applied in cooperation with its parts. The verification rules that agents should verify and accomplish their test tasks are directly related to requirement identification, risk assessment, and defense tools.

The MBSE for SCADA and ICS systems is presented in the preceding chapter. For the SCADA and ICS test with EMAS TEST, as an initial stage, we define the trust requirements for SCADA and ICS, followed by the identification and assessment of potential risks. Thus, risk mitigation and defense. These parts aid in the test process of the trustworthiness of SCADA and ICS systems because the verification rules in the EMAS TEST model are directly related to verifying these requirements and the assessment of the risk and availability of defense techniques for systems.

5.3.1 Requirement definition

A comprehensive and methodical approach to requirement formulation and management is to determine the trust needs of SCADA and ICS [183]. Typically, the most important stage in the design of software and systems is the requirement phase. They function as implementation guidelines and as a point of reference for the validation and testing of the finished product. The vital requirements for trust systems and the required trust qualities are outlined in [155]. The set of trust requirements for SCADA and ICS is presented in Table 5.1.

Table 5.1 Trust requirement for SCADA and ICS systems

Trust Level	Requirements
Level 1	*Safety*: SCADA and ICS should operate with the assurance that there won't be any catastrophic consequences for the environment or the lives, health, assets, or data of stakeholders. *Security*: Every process, system, and service that SCADA and ICS provide should be protected from unwanted and accidental access, alteration, damage, loss, or usage, whether from the inside or the outside. *As sub-properties of security:* *Confidentiality*: ICS and SCADA systems should operate while upholding authorized access and disclosure restrictions. *Integrity*: Non-repudiation and authenticity, as well as protection against unauthorized system modification or destruction, are essential components of SCADA and ICS functionality. *Availability*: SCADA and ICS must be accessible and usable in a reliable and timely manner.
Level2	*Privacy*: The operation of SCADA and ICS should be based on preventing entities (computers and people) from accessing data created, stored, or transferred through the system or any of its components; privacy is essential to building the system's credibility. Users are in charge of and have access to their personal data. *Performance*: The quality that defines how well a service performs should be met by SCADA and ICS, as well as the necessary operational targets. system operates as planned within predetermined parameters, such as speed, accuracy, or memory usage (IEEE-610.12). *Dependability*: Accuracy, Availability, Robustness, Reliability, Scalability, and Maintainability are the sub-properties that make up Dependability. The service it offers backs up the capabilities. Everything went according to plan with a proper and predictable execution. *Compatibility*: SCADA and ICS ought to operate with hardware and software from different sources that are connected without the need for modifications.
Level3	*Usability*: ICS and SCADA should function and be used efficiently to meet user needs and operational goals (ISO 9241-210). *Correctness*: refers to the way a system operates in accordance with user specifications, including user standards and expectations on reliability.

5.3.2 Risk identification and assessment for SCADA and ICS

The natural gas and petroleum sectors are intricate ecosystems comprising pipelines that run upstream and downstream, as well as onshore and offshore operations. Although these organizations are all in the same sector, there are risks and potential hazards unique to each

of them. This sector remains open to numerous threats, including unauthorized access, cyber-attacks, natural disasters, operational mishaps, and political instability.

There is a dearth of risk modeling in the petroleum field literature. The O&G industry lacks classification of all potential risks. Risk is currently divided into two categories: 1) internal and external risks and 2) broad classifications, which include operational mishaps, natural disasters, and geopolitical conflicts. Not all possible risks were addressed, and all aspects or causes of risk were highlighted in these ratings. We addressed this shortcoming in [187] and offered a risk classification for the oil and gas industry as a whole. We classified the risks based on the knowledge that the cyber-physical system is the most crucial element of the industry and that the risks are either physical, cyber, or related to permission and authorization for oil and gas companies. The well-known risks to the oil and gas industry's SCADA, ICS, petroleum CPS, and system structure are presented in [187].

Risk identification is the foundation of risk management and is essential for protective action plans. The table 5.2 summarizes the identified risks for petroleum and gas SCADA and ICS.

Table 5.2 Risks identification for SCADA and ICS systems

Risk	Type	Impact
-Tank attacks – Attacks target production - Wellhead production data exfiltration – Drone attacks	Physical	-Physical damage destruction, burning, vandalism - Destruction to buildings, transportation equipment, storage levels, and storage equipment
-Denial-of-service (DoS) attacks -Command Injection attack- Data exfiltration attacks - Data tampering attacks	Cyber	- Stop information from passing through control systems. - Putting worker, reputation, operations, environment at risk
-*Internal*: lack of threat awareness by employees - lack of strong authentication and authorization rules for personnel, human errors -*External* operating SCADA /ICS remotely	Authorization	Firmware modifications, incorrect setups, open ports, faulty equipment, communication issues

5.3.3 Risks mitigation and defense in SCADA and ICS

The most commonly used vulnerability mitigation and management strategies in the oil and gas industry are network segmentation and various technologies [183].

We proposed a multifaceted approach in [187] as a mitigation technique that led to the prevention of SCADA and ICS from the classified risks: physical, cyber, and authorization.

To improve security, we split the petroleum zone into systems (ICS and SCADA), as shown in Figure 1. The goal of the system identification is to divide the system into discrete security subzones and add layers of protection to isolate the most essential components.

We consider the proposed mitigation solution that has already been provided in our case, and the SCADA and ICS are mitigated. A summary is presented in Table 5.3.

Table 5.3 Mitigation for SCADA and ICS systems

Type of risk	Mitigation	Objectives
Physical	-The ISO 45001 standard for occupational health and safety - API Standard 780 is employed by pipeline operators	- Aid in protecting employees and visitors from diseases and incidents that may arise from their jobs. - Aid for security risk assessments (SRAs),and led to identify and reduce hazards.
Cyber	- (IEC) 62443 is a leading set of standards for industrial control systems (ICS) security - (ISO) 27000: the standards for information security management systems	SCADA and ICS, and their security lifecycles in the oil and gas sector, should be maintained and accorded with a set of security standards.
Authorization	-Using Next generation firewalls (NGFW) -Companies enforce rules, regulations, policies, standards, and directives. - Authentication procedures	-Reduce the risk of unwanted access (or network traffic).- Reduces the permission and provides continuous surveillance of all traffic.-Provide automatic real-time analytics detection

5.4 Testing trust in SCADA and ICS with EMAS TEST approach

The test in the EMAS TEST was based on verification rules. Each agent property in the EMAS TEST uses these rules to accomplish their tasks.

5.4.1 Define relation between rule verification and risk assessment and defense and mitigation availability.

The set of verification rules: Ri
The set of requirement : RQ
The set of assessed risk; RS
The set of provided defense: PD

The Table 5.4 presents the relation between rule verification and risk assessment and defense and mitigation availability

5.4.2 Define a confident and logical interval

The agent property embeds an algorithm to verify this property. The previously identified requirements, risks, and mitigation were used as inputs. Therefore, we need a confident value for these inputs, and the interval is logical [0, 1]. Table 5.5 defines values of verified rules for SCADA and ICS systems.

Table 5.4 Relation between rule verification and risk assessment and defense and mitigation availability

Verification Objectives	Ri : Verification Rules	RQ	RS	PD
Requirement	**R1**:If all requirements are identified and filtered from all actors, agent will inform that the set of requirements for initial conception is verified	√	×	×
CPS Realization	**R2**:If the requirement are verified during realization of CPS or after the agent will announce that the requirement are meet	√	×	×
CPS Usage , Certification , Standard	**R3**:If the CPS is used and tried within many scenarios, integrate standards, or certified, the agent will announce that CPS is standardized , or certified in this property.	×	×	√
Identification and Assessment of Risks	**R4**:If the risks and threats surrounding CPS are assessed, the agent will announce that risks in CPS are assessed	×	√	×
Verification of availability of Mitigation and Protection	**R5**:If a mitigation tools or defensive methods integrated within CPS, the agent will announce that CPS is protected	×	×	√

Table 5.5 Defined values of verified rules for SCADA and ICS systems

Property	RQ	RS	PD
Safety	1	1	1
Security	1	1	1
Privacy	1	1	1
Performance	1	1	0
Dependability	1	1	0
Compatibility	1	1	1
Usability	1	0	1
Correctness	1	0	0

For each property, RQ should be equal to 1. If RQ equals 0, R1 and R2 will be considered not true. In this case, the property is considered negative, and Pi equals zero.

For safety and security, privacy, performance, and compatibility, the RS should be equal to 1. For dependability, usability, and correctness, the RS may be equal to zero or one. Because the RS parameter is related to and attached to secured level and trustworthy level, threats and attacks target more the properties of safety, security, and trustworthiness.

The PD parameter should be equal to one for the properties of safety and security, privacy, performance, and compatibility, because their RS should equal one because they are vulnerable to the risks that should be mitigated. In our case of SCADA and ICS, Tables 5.2 and 5.3 shows that the risk for SCADA was identified and that adequate mitigation was provided.

5.5 Implementation of EMAS TEST with JADE platform

The platform that the system will run on was selected as the JADE. There are several reasons. The primary reason is communication, particularly remote communication, which involves locating an agent on the platform, message-sending options, and an agent's independence from other agents. Our EMAS TEST model consists of communication between agents to complete the entire test and to ensure cooperation. Additionally, JADE is widely used by researchers, and many of them use it in their publications and experiments [188]. In addition, a main user interfaces that is easy to use and facilitates agent control.

5.5.1 Brief Description of development environment

We used the JADE platform and the JAVA language to develop our EMAS TEST. One well-known multi-agent platform is the Java Agent Development Framework (JADE). The Foundation for Intelligent Physical Agents (FIPA) is the foundation for agents inside. JADE is designed to make it easier to create agent applications that meet the FIPA requirements for intelligent multi-agent systems. The other goal is to maximize the performance of the distributed-agent system. Because of its architecture, which is based on agents (agents and controlling agents) rather than layers, and appears to be highly flexible, the majority of researchers believe that JADE is the most sophisticated platform. Jade contains:

- A runtime environment is an environment in which agents can exist. This runtime environment must be activated to deploy the agents.
- A class library: which the developers use to write their agents.
- Graphical tools: A set of graphical tools that make it easier to manage and supervise the agent platform.

Each JADE instance is referred to as a "container" and can contain several agents [189]. A platform was formed by a group of content creators. Each platform must have a special container called the main container, and all other containers must be registered before they can be launched. A main container differs from other basic containers in that it always contains two special agents, AMS and DF, which are launched automatically when the container is launched.

- AMS Agent Management System (AMS), which provides the name of service (for example, ensuring that each agent has a unique identifier on the platform) and represents the platform's authority (for example, it is possible to create/stop agents by sending requests to the AMS).
- A DF (Directory Facilitator) provides a yellow page system that allows agents to find service providers.

5.5.2 EMAS TEST in JADE

5.5.2.1 Main container and agents (creation and activation)

The figures 5.2 and 5.3, 5.4 represent the creation and activation of main container and agents.

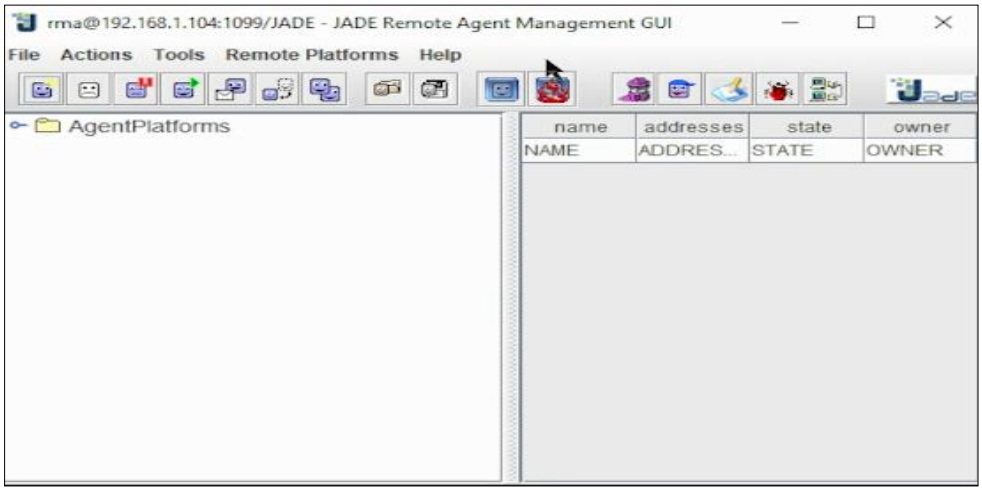

Figure 5.2 RMA JADE Start

```
package TEST;

import jade.core.Profile;

public class MyMain {
    public static void main(String[] args) {
        // Get a hold on JADE runtime
        Runtime rt = Runtime.instance();

        // Create a default profile
        Profile profile = new ProfileImpl();
        profile.setParameter(Profile.MAIN_HOST, "localhost");
        profile.setParameter(Profile.GUI, "true");

        // Create the Main-container
        AgentContainer mainContainer = rt.createMainContainer(profile);
```

Figure 5.3 Main container creation

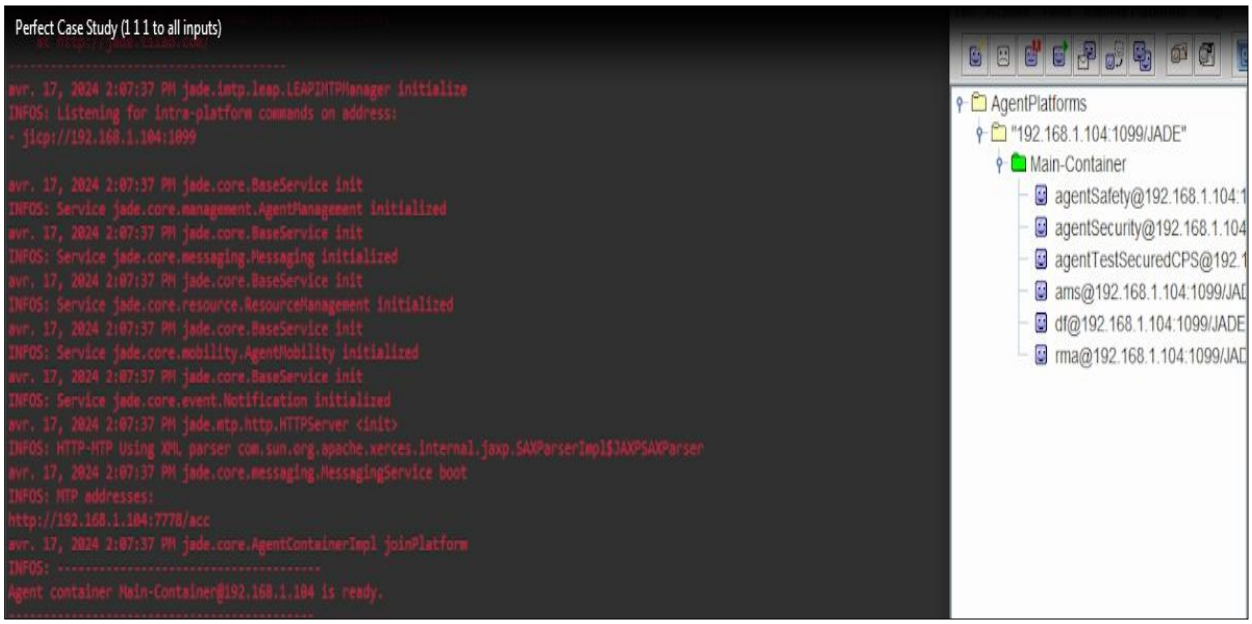

Figure 5.4 Main container and agent activation

```
private void sendConfirmPresenceMessage(String receiverName) {
        try {
            Thread.sleep(5000); // Delay for 2 seconds
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    ACLMessage reply = new ACLMessage(ACLMessage.CONFIRM);
    reply.addReceiver(new AID(receiverName, AID.ISLOCALNAME));
    reply.setContent("Presence confirmed");
    send(reply);

    System.out.println("Sent presence confirmation to " + receiverName + "(Presence Confirmed)");
}
```

```
      public void action() {
          try {
              Thread.sleep(5000); // Delay for 2 seconds
          } catch (InterruptedException e) {
              e.printStackTrace();
          }
          MessageTemplate mt = MessageTemplate.MatchPerformative(ACLMessage.REQUEST);
          ACLMessage msg = myAgent.receive(mt);
          if (msg != null) {

              System.out.println("Received message from " + msg.getSender().getLocalName() + ": " + msg.getContent());

              if (msg.getContent().equals("Requesting presence")) {
                  sendConfirmPresenceMessage(msg.getSender().getLocalName());
              } else if (msg.getContent().equals("Requesting task accomplishment")) {
                  // Perform the safety test and send the results
                  performSafetyTest();
              }
          } else {
              block();
          }
      }
  });
```

Figure 5.5 Test of reputation for agent safety

5.5.2.2 Test via EMAS TEST for the reputation of agent in the system

One of the objectives of the EMAS Test was to test the reputation of each agent. Reputation is verified by sending a message test for presence and a message request for task accomplishment. If the agent confirms its presence and sends the result of its task, it will be considered a reputable agent from its supervisor agent test level. Figure 5.5 presents the reputation testing code for agents.

5.5.2.3 Test the safety quality by agent safety

Each agent property carries out a test of the unique properties of SCADA in the oil and gas industry. Figure 5.6 presents the test of the safety property in SCADA. The other agent property uses the same method of testing, which was explained in detail in the previous sections.

```
public String performSafetyTest() {

    int RQ = 1; // Case Study
    int RS = 1;
    int PD = 1;

    if (RQ == 1) {
        R1 = true;
        R2 = true;
    }
    if (RS == 1) {
        R4 = true;
    }
    if (PD == 1) {
        R3 = true;
        R5 = true;
    }
    if (R1 && R2 && R3 && R4 && R5) {

        sendTestResultsToAgentTestSecuredCPS();

        return "The Safety is verified positively";
    } else {
        sendTestResultsToAgentTestSecuredCPSNegative();
        return "Safety is verified negatively";
    }
```

Figure 5.6: Testing safety property for petroleum SCADA

5.5.2.4 Test the Level One: Secured SACAD and ICS

Figure 5.7 presents the result of testing level one, which is secured CPS. At this level, safety and security are mandatory properties for testing level one. A test of an agent's reputation was also presented. We start by verifying the presence of the agent test of property, then the confirmation of presence, and then the request to accomplish the task. When the agent sends the result of its task, which is the testing property, that means the agent is reputable.

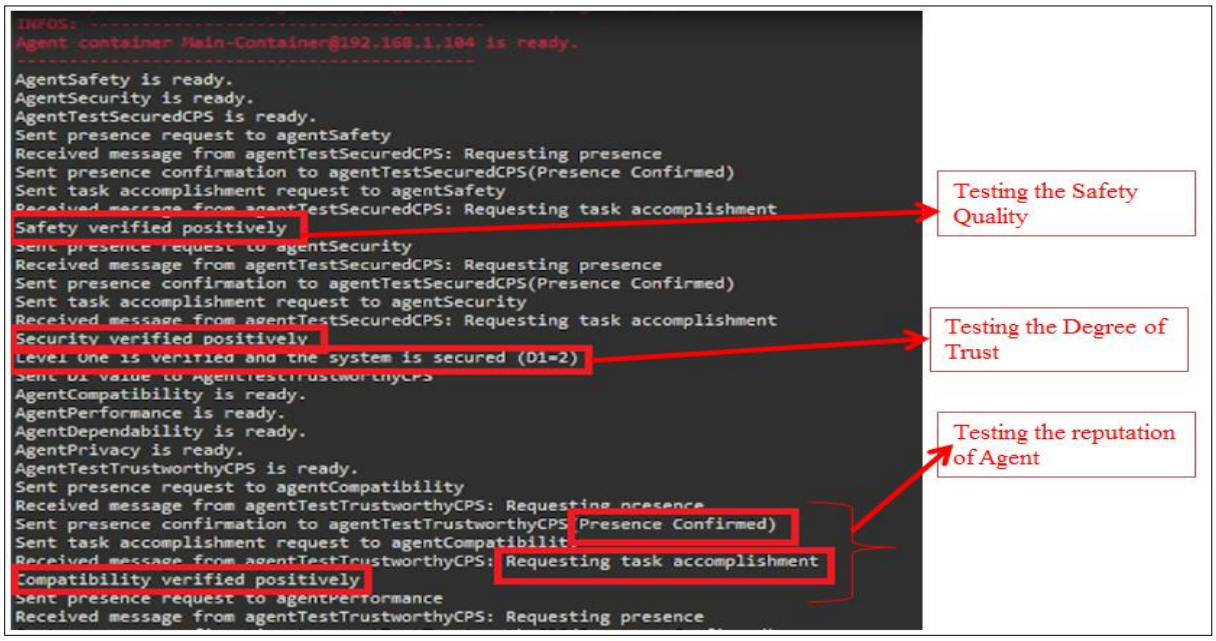

Figure 5.7: implementation result for the test of trust petroleum SCADA and ICS systems

5.5.2.5 Test the Level Two: Trustworthy SCADA and ICS

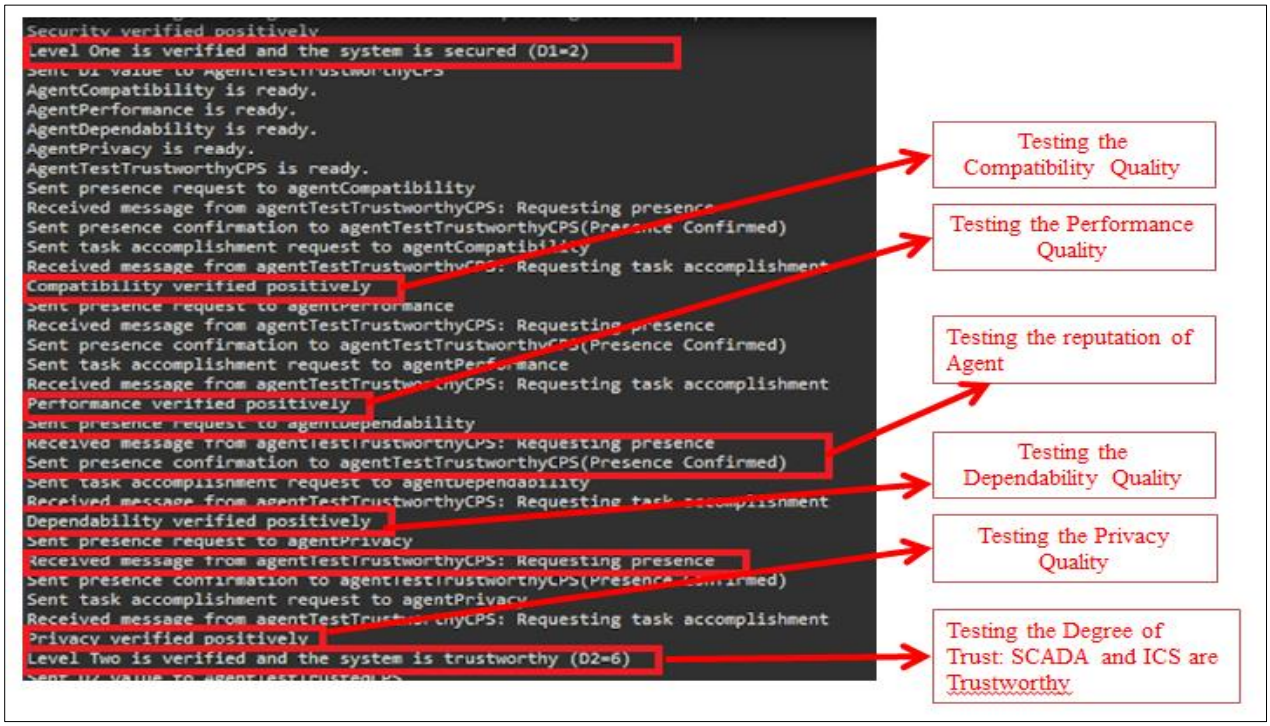

Figure 5.8: Test results for level two of trust petroleum SCADA and ICS systems

Figure 5.8 presents the result of testing level two, which is a trustworthy CPS. At this level, the mandatory properties include compatibility, performance, dependability, and privacy. The reputation of each agent's properties is also verified.

5.5.2.6 Test the Level Three: Trusted SCADA and ICS

Figure 5.9 Test results for level three of trust petroleum SCADA and ICS systems

Figure 5.9 presents the result of testing level three, which is the trusted CPS. At this level, the mandatory properties are usability and correctness. The reputation of each agent's properties is also verified.

5.5.2.7 Some negatives test cases

Some cases represent reports of errors and negative test results if the verification rules are applied to the system and the quality of testing is not assured, the input data for the test are empty, the test of agent reputation is negative, and the agent in the EMAS TEST is not reputable. Table 5.6 presents some negative test cases

Table 5.6 Some negative test cases

Agent	Negative Case	Task	Error message
Agent test level 1	If the agent property does not answer the PR message from the agent test level and does not confirm their presence,	The Agent test level will announce that this agent property is malicious and unreputable and untrustworthy	Agent property is malicious and unreputable and untrustworthy
Agent property	If the inputs data are empty	Agent property can't test the property	Property cannot be tested
Agent test level 2	The level one is verified negatively and CPS is not secured	The agent property of level two will test each related property	The properties are verified but the level one not secured
Agent test level 3	The level two is verified negatively and CPS is not trustworthy	The agent property of level three will test each related property	The properties are verified but the level two is not trustworthy

The figure 5.10 presents some negatives test cases.

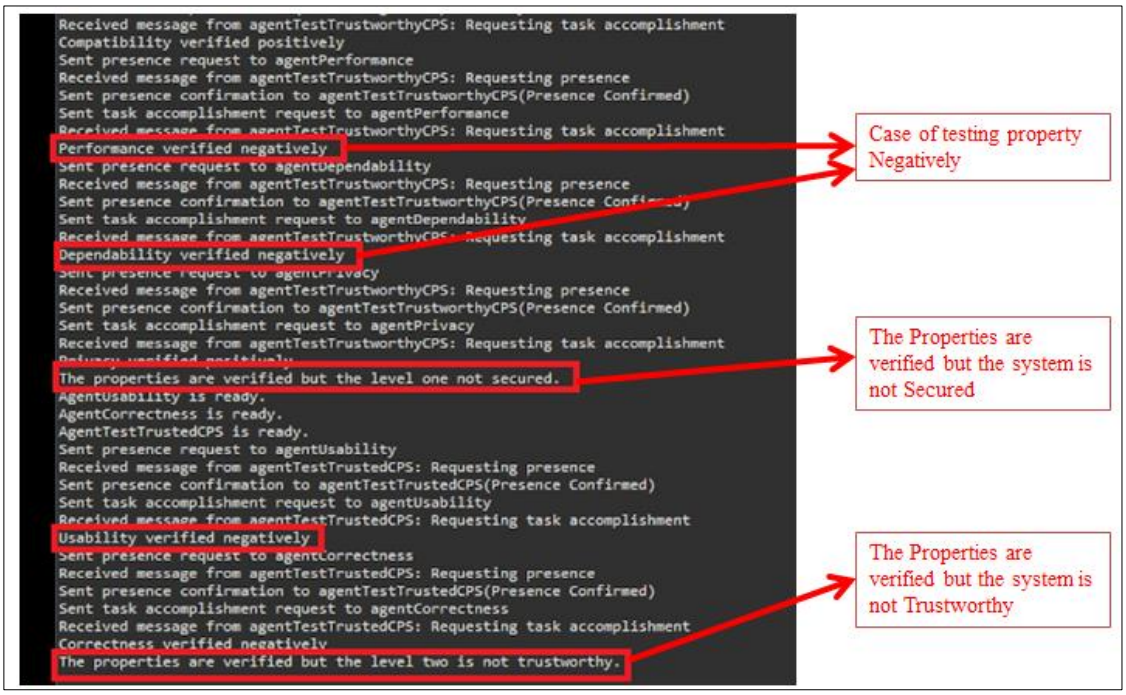

Figure 5.10. Some negative case in testing trust petroleum SCADA and ICS systems

5.6 Results and evaluation

5.6.1 Results

The result of study can be divided into two types: Theoretical and practical

a) Theoretical

Many challenges in the literature related to trust and cyber-physical system modeling and testing have been addressed, and we propose suitable solutions. The analysis of trust in CPS and well-defined concerns, requirements, and properties are provided. The evaluation of trust in CPS is proposed from a quantitative perspective using a simple mathematical formula that will ease the test and aid in the assurance of trust in CPS.

CPS, which is a complex and heterogeneous system, is well modeled with the most recommended multifaceted approach for security and confidence design, MBSE. The deep concepts of system engineering and requirement engineering were employed, as well as the design tools, such as the different diagrams of SysML and UML.

The use of artificial intelligence in combination with modern modeling tools was presented in our study. A multifaceted test approach results from this merger. An embedded multi-agent system is used with verification technique-based rules to design a test model for trust CPS. For efficiency and assurance of the proposed TEST EMAS, we also provided a

reputation test for each agent within our test model, which enables trust testing in an embedded multi-agent system.

b) Practical

The proposed models and approaches are experimented, such as MBSE, and validation for EMAS TEST is deployed on critical CPS, which are oil and gas SCADA and ICS systems. Model-based system engineering is designed, and the trust requirement and verification rules are applied to those systems, as well as risk assessment and mitigation. EMAS TEST was employed to test trust in SCADA and ICS on the JADE platform. All levels of trust were tested in CPS, and the reputation of the embedded multi-agent systems was verified. The JADE platform facilitates the evaluation and testing of trust as it is designed.

5.6.2 Evaluation

The table 5.7 presents the evaluation of the EMAS TEST model. The evaluation was based on all phases of the design and implementation of the approach.

Table 5.7 Facets of EMAS TEST and Evaluation

Conception	**Modeling**	**Implementation**	**Objectives**			
			Real Time	**Test Agent Reputation**	**Test Property**	**Test Trust Degree**
✓ Verification Technique of MBSE ✓ Real time consideration ✓ Agent test algorithms ✓ Agent test level algorithms ✓ Verification rules	✓ SysML and UML ✓ Components Diagram ✓ Functionality Diagram ✓ Requirement Diagram ✓ Risk Assessment ✓ Mitigation	✓ Embedded Multi agent system with JADE Platform ✓ Agent creation ✓ Agent verification ✓ Trust verification	√	√	√	√

The conception and implementation of EMAS Test help us reveal some points:

- The EMAS TEST was designed based on the system (CPS) and quality (trust) analysis and modeling. This can be considered as a multidisciplinary approach.
- The choice of the Jade platform is based on the system requirements for runtime and the ease of communication that Jade assures.
- The remote monitoring agent (RMA) controls the life cycle of the agent platform and all registered agents.
- Dummy agent tool allows the tester to interact with JADE agent.
- Communication: Agent sends/receives Java objects represent ACL messages within the scope of the interaction protocols.

- The reputation of the agent in EMAS TEST is tested to guarantee agent activation and tasks in JADE, the agent cycle life managed by RMS, and AMS (Agent Management System).
- The test of trust quality in CPS is performed via efficient communication assured by the ACL and FIPA standards in JADE.

There are many challenges that led to the crystallization of some shortcomings in our work, which can be mentioned as limitations:

- The proposed testing approach is not automated while cyber-physical systems operate in an automated environment.
- The operating platforms for the SCADA systems and their related software were not available to enable real testing of the proposed testing method within realistic scenarios.

5.7 Conclusion

This chapter details the experimentation of the proposed models and approaches, starting with the introduction of SCADA and ICS and then modeling trust SCADA and ICS for the oil and gas industry. Defining trust requirements, assessing risk for SCADA and ICS, and applying verification rules. The EMAS Test assessed trust in the SCADA and ICS using the JADE platform. The CPS was evaluated, all levels of trust were verified, and the reputation of the embedded multi-agent systems was tested. The JADE platform simplifies the evaluation and testing of trust as planned. The results are then presented and evaluated.

CHAPTER 6

GENERAL CONCLUSION

"Woe to the seeker of knowledge if he is satisfied with himself."

Taha Hussien

6.1 Thesis Summary

The complexity of cyber-physical systems presents unparalleled testing issues in terms of functionality, non-functionality, and quality. In this study, we tested the quality of trust in the CPS. We addressed the challenges of defining and analyzing trust in a cyber-physical system (CPS). In addition, we addressed the issue of modeling trust in CPS and incorporating verification into the CPS lifecycle. Given the breadth and depth of this topic, we divided our study into parts.

Chapter one introduced the context of the thesis as well as its objectives and contributions. The second part presents the state of the art of the topic. The cyber physical system is presented in Chapter 2. The fundamental design principles of CPS and real-world applications are presented, along with the definition of embedded multi-agent systems (EMAS) and their characteristics. The integration of multi-agent systems into cyber-physical systems was addressed. Chapter Three presents testing trust in cyber-physical systems and embedded multi agent systems. Many points have been addressed, such as the importance of trust in CPS, the challenges in testing trust in cyber-physical systems, and methods and techniques for testing trust in cyber-physical systems. Methods for testing trust in embedded multi-agent systems and the challenges of testing trust in EMAS are presented.

Part two contains chapter four and five. The design of proposed solution is presented in chapter four; a technique for testing trust in CPS includes a study of trust quality in CPS as well as a presentation on modeling trust in CPS. A thorough understanding of trust quality, which encompasses numerous qualities and sub-properties, aids in conceptualizing the cyber physical system and decreasing the complexity of the CPS aspect and quality within its context. The trust degree framework is extremely useful and versatile; it allows for separate computation of the trust of CPS components, which might be software, hardware, or a network. This also facilitates and simplifies the quantitative computation of trust. Boost the development of trust in CPS from the beginning

MBSE, which is the key tool for the development of complex systems and qualities such as security and trustworthiness and is widely used for collecting and controlling the system's requirements, is used in this thesis to model trust CPS. In our proposed MBSE method, a verification phase is considered and presented step-by-step (element verification, requirement verification, attribute verification, and quality verification). This testing mechanism is combined with embedded multi-agents that target

verification of trust quality in CPS. The methodological level, decentralized nature of EMAS, and analysis help address the complexity of trustworthiness quality.

A description of the EMAS TEST model is presented, including the verification methods, agent behavior, and characteristics. In addition to the required algorithms, verification rule definitions and agent tasks inside EMAS TEST.

The concept of EMAS TEST is detailed, and the advantages of the EMAS Test are highlighted. The proposed strategy targets the gap in complex system verification with comprehensive and analytic tools and a variety of models. These analytical and conceptual tools enabled us to cover the entire development cycle of CPS and conduct its tests.

Chapter five describes the experiments with the proposed models and techniques, beginning with an introduction to SCADA and ICS, and progressing to modeling trust SCADA and ICS for the oil and gas industry. Defining trust needs, evaluating risk for SCADA and ICS, and implementing verification standards. The EMAS Test, which utilized the JADE platform, tested trust in the SCADA and ICS systems. CPS was assessed, all degrees of trust were validated, and the reputation of the embedded multi-agent systems was checked. The JADE platform facilitates the examination and testing of trust as planned.

As a result, we address several difficulties in the literature related to trust and cyber physical system modeling and testing and provide appropriate solutions. A trust analysis for CPS is presented, along with a well-defined set of concerns, criteria, and properties. A quantitative measurement of trust in CPS is proposed, along with a simple mathematical formula that makes the test easier and helps ensure faith in CPS. Figure 6.1 presents the trust CPS analysis.

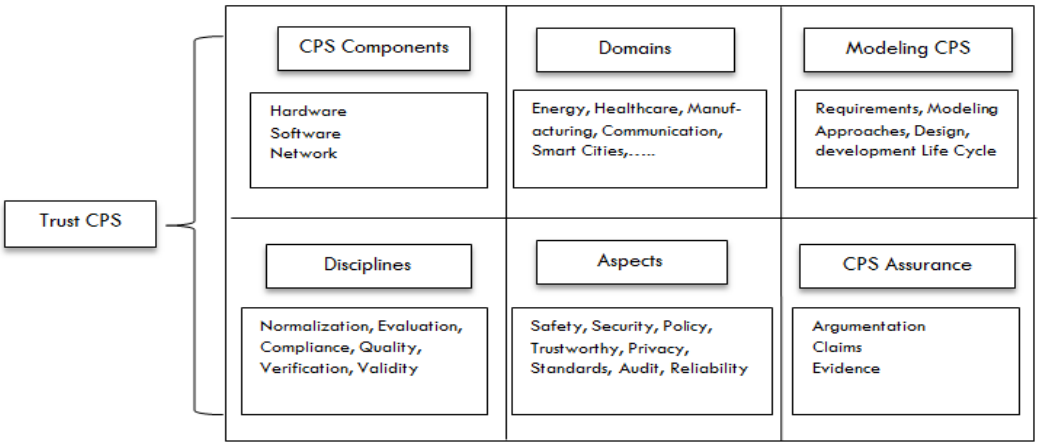

Figure 6.1: Trust CPS Analysis

This study combined artificial intelligence with a variety of modern modeling techniques. The merger resulted in a comprehensive testing strategy. An embedded multi-agent system is utilized in conjunction with verification technique-based rules to create a test model for trusting CPS. The suggested TEST EMAS is backed by a reputation test for each agent in our test model, allowing us to test trust in an embedded multi-agent system.

This study emphasizes the significance of establishing and modeling trust propriety in embedded multi agent systems (EMAS), where agents collaborate to achieve shared tasks for the goal of quality verification. We propose a novel perspective on trust verification, wherein a group of agents verifies the necessary quality of trustworthiness in a CPS and provides a computational view.

EMAS TEST was employed to test trust in SCADA and ICS on the JADE platform. All levels of trust were tested in CPS, and the reputation of the embedded multi-agent systems was verified. The JADE platform facilitates the evaluation and testing of trust as it is designed.

6.2 Future Work

The presented thesis achieved an analysis of trust in cyber-physical systems and touched on many facets, such as conception, implementation, and verification. CPS is the pillar of technology 4.0, and is employed in all fields. Our future work will extend the EMAS Test technique to establish and test trust architectures for fleets of drones, which are considered a collective cyber physical system.

Unmanned aerial vehicles (UAVs) with differing components are employed for a range of purposes, such as transportation, military activities, and surveillance. A ground station controller, flight control board (FCB), rotor system, electronic speed controller (ESC), transceiver control unit (TCU), sensors, actuators, and power management system (PMS) comprise the drone system. It is predicted that fleets of these vehicles, or drones, will perform a variety of tasks, such as battlefields for conflict and combat missions. They can also function alone, on solo missions, or in groups, in a process called fleet missions.

The risks that drones face have been examined in the literature from a number of angles, such as network issues, drone forensics, communication, and collisions, but not all actual and prospective threats and concerns have been covered. A range of mitigation strategies were also addressed. Current and implemented strategies focus on a particular risk, and there are few innovative long-term mitigation and defensive strategies that safeguard UAV functionality and guarantee their security and reliability in their surroundings.

The scientific community is mainly concerned with the security, privacy, and trust of drone fleets as well as their performance and energy consumption.

Our future research plan will address these topics, which are still in the same sphere as cyber-physical systems and trust, but are handled in a different field, specifically UAVs. This technology has become the focal point of all recent and ongoing conflicts and warfares.

As preparation for future work, we first started by classifying existing risks for the security of drones as internal and external risks [190]. The proposed security threat classification for UAVs facilitates risk assessment and design of a defensive security plan for UAVs.

Our proposed defensive security plan (DSP) presented in Figure 6.2 is suitable for all drone types and applications, both military and non-military. DSP's primary benefit is that it offers long-term security verification and upgrades defenses with new equipment.

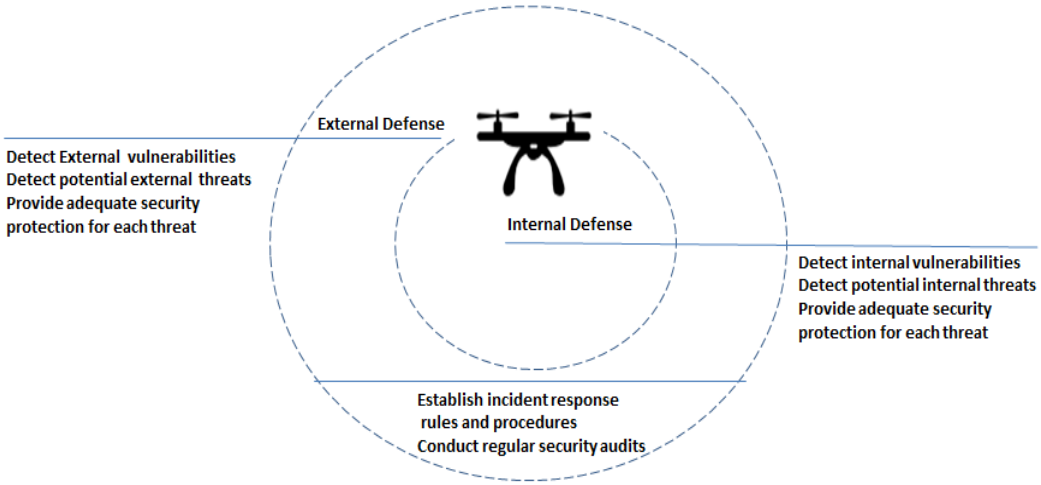

Figure 6.2: Defensive security plan (DSP) for UAV [190]

Many questions should be considered, and our future work will be addressed.

Q1: How can the EMAS TEST model be extended to fleet drones and how can it be integrated within the DSP?

Q2: We suggest trusting UAV architecture for drones [190] as a means of protecting all components of a drone, including hardware, software, network, and communication. How can we implement this trust architecture for fleet drones?

Q3: How is employing the extended EMAS TEST model for testing fleet drones?

Q4: The EMAS TEST model requires that risk assessment and mitigation be established in the system. What countermeasures and mitigation techniques should be established for the fleet of drones?

Q5: Fleet drones are complex and collective cyber-physical systems that require a powerful and strong platform for implementing the extended EMAS test. Which platform is most suitable for these challenges?

Bibliography

[1] Schneider, D., Armengaud, E., & Schoitsch, E. (2014). Towards trust assurance and certification in cyber-physical systems. In Computer Safety, Reliability, and Security: SAFECOMP 2014 Workshops: ASCoMS, DECSoS, DEVVARTS, ISSE, ReSA4CI, SASSUR. Florence, Italy, September 8-9, 2014. Proceedings 33 (pp. 180-191). Springer International Publishing.

[2] Ebert, C., Jones, C.: Embedded Software: Facts, Figures and Future, pp. 42–52. IEEE Computer Society (2009)

[3] Somers, R. J., Douthwaite, J. A., Wagg, D. J., Walkinshaw, N., & Hierons, R. M. (2023). Digital-twin-based testing for cyber–physical systems: A systematic literature review. Information and Software Technology, 156, 107145.

[4] Abbaspour Asadollah, S., Inam, R., & Hansson, H. (2015). A survey on testing for cyber physical system. In Testing Software and Systems: 27th IFIP WG 6.1 International Conference, ICTSS 2015, Sharjah and Dubai, United Arab Emirates, November 23-25, 2015, Proceedings 27 (pp. 194-207). Springer International Publishing.

[5] Perumalla, K. (2021, April). Trust-but-Verify in Cyber-Physical Systems. In Proceedings of the 2021 ACM Workshop on Secure and Trustworthy Cyber-Physical Systems (pp. 1-2).

[6] Akintunde, M., Yazdanpanah, V., Salehi Fathabadi, A., Cirstea, C., Dastani, M., & Moreau, L. (2024). Actual Trust in Multiagent Systems.

[7] https://fr.wikipedia.org/wiki/Syst%C3%A8me_cyber-physique

[8] Lee, E.A. (2008). Cyber physical systems: Design challenges. In 2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC), Orlando, FL, USA, pp. 363-369. https://doi.org/10.1109/ISORC.2008.25

[9] Lee, E.A. (2006). Cyber-physical systems-are computing foundations adequate. In Position Paper for NSF Workshop on Cyber-Physical Systems: Research Motivation, Techniques and Roadmap, pp. 1-9.

[10] Shah Ahsanul Haque,1 Syed Mahfuzul Aziz,1 andMustafizur Rahman2, "Review of Cyber-Physical System in Health care" 'International Journal of Distributed Sensor Networks" vol 2014.

[11] Aguida, M.A., Ouchani, S., Benmalek, M. (2020). A review on cyber-physical systems: models and architectures. In 2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Bayonne, France, 275-278.

[12] G. Magureanu, M. Gavrilescu, D. Pescaru, and A. Doboli, 'Towards UML modeling of cyber-physical systems: A case study for gas distribution', in IEEE 8th International Symposium on Intelligent Systems and Informatics, Subotica, Serbia, 2010

[13] E. Palachi, C. Cohen, and S. Takashi, 'Simulation of cyber physical models using SysML and numerical solvers', in 2013 IEEE International Systems Conference (SysCon), Orlando, FL, 2013
[14] J. Fitzgerald, P. G. Larsen, and M. Verhoef, Collaborative Design for Embedded Systems. Berlin Heidelberg: Springer- Verlag, 2014
[15] J. Liu, Continuous Time and Mixed-Signal Simulation in Ptolemy II.TechnicalReportNo. UCB/ERL M98/74. 1998
[16] E. A. Lee, M. Niknami, T. S. Nouidui, and M. Wetter, 'Modeling and simulating cyber-physical systems using CyPhySim', in 2015 International Conference on Embedded Software (EMSOFT), Amsterdam, Netherlands, 2015.
[17] S. Oueida et al., "A smart healthcare reward model for resource allocation in smart city," in Multimed. Tools Appl., vol. 78, pp. 24573- 24594, 2019.
[18] S.K. Polu, "Modeling of Efficient Multi-Agent based Mobile Health Care System," in Int. J. Innov. Res. Sci. Tech., vol. 5, pp. 10-14, 2019.
[19] M. Abu-Matar, "Towards a software defined reference architecture for smart city ecosystems," in Proc. IEEE ISC2, Trento, 2016, pp. 1-6 pp. https://doi.org/10.1109/WETICE49692.2020.00060
[20] F. Baena, A. Guarin, J. Mora, J. Sauza, and S. Retat, "Learning factory: The path to industry 4.0," Procedia manufacturing, vol. 9, pp. 73–80, 2017.
[21] D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, "Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation," Computer-aided design, vol. 59, pp. 1–14, 2015.
[22] C. Liu and P. Jiang, "A Cyber-physical System Architecture in Shop Floor for Intelligent Manufacturing," in Procedia CIRP, vol. 56 , pp. 372-377, 2016.
[23] X.F. Liu et al., "Cyber-physical manufacturing cloud: Architecture virtualization communication and testbed", in J. Manuf. Syst., vol. 43, pp. 352-364, 2017.
[24] Y. Zhang et al., "Health-CPS: Healthcare Cyber-Physical System Assisted by Cloud and Big Data," in IEEE Syst. J., vol. 11, no. 1, pp. 88-95, 2017.
[25] E. Sultanovs et al., "Centralized healthcare cyber-physical system's architecture development," in Proc. IEEE RTUCON, pp. 1-6, 2016.
[26] L.C. Silva et al., "A model-based architecture for testing medical cyber-physical systems," in Proc. 29th ACM S. Appl. Comput., pp. 25-30, 2014.
[27] S. Sakr and A. Elgammal, "Towards a comprehensive data analytics framework for smart healthcare services," in Big Data Res., vol. 4, pp. 44-58, 2016.
[28] Hahn, A.; Ashok, A.; Sridhar, S.; Govindarasu, M. Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid. IEEE Trans. Smart Grid 2013, 4, 847–855.
[29] Sun, C.C.; Hong, J.; Liu, C.C. A co-simulation environment for integrated cyber and power systems. In Proceedings of the 2015 IEEE International Conference on Smart Grid Communications (SmartGridComm), Miami, FL, USA, 2–5 November 2015; pp. 133–138.
[30] Shi Tianjin , Wu Xu, Guan Jizhen, Chen Yangzhou "The analysis of traffic control cyber physical systems" ." 13th COTA International Conference of Transportation Professionals (CICTP-2013)" pgno 2487- 2496.
[31] C.G. Cassandras, "Smart Cities as Cyber-Physical Social Systems," in Eng. J., vol. 2, no. 2, pp. 156-158, 2016.

[32] Daniel Pickem, Paul Glotfelter, Li Wang, Mark Mote, Aaron Ames, Eric Feron, and Magnus Egerstedt; "The Robotarium: A remotely accessible swarm robotics research testbed"; 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, May 29 - June 3, 2017.

[33] J. A. Tran et al., "Intelligent Robotic IoT System (IRIS) Testbed," IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018.

[34] T Farnham†, S Jones‡,"UMBRELLA Collaborative Robotics Testbed and IoT Platform".In 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC).

[35] Zhou, J., Li, L., Vajdi, A., Zhou, X. & Wu, Z. Temperature-constrained reliability optimization of industrial cyber-physical systems using machine learning and feedback control. IEEE Trans. Autom. Sci. Eng. 99, 1–12 (2021). DOI: 10.1109/TASE.2021.3062408

[36] Chen, X., Zhou, Y., Zhou, H., Wan, C., Zhu, Q., Li, W., & Hu, S. (2016, November). Analysis of production data manipulation attacks in petroleum cyber-physical systems. In 2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD) (pp. 1-7). ACM. DOI: 10.1145/2966986.2980091.

[37] Alcaraz, C., & Zeadally, S. (2013). Critical control system protection in the 21st century. Computer, 46(10), 74-83. DOI: 10.1109/MC.2013.69

[38] Stellios, I., Kotzanikolaou, P., Psarakis, M., Alcaraz, C., & Lopez, J. (2018). A survey of iot-enabled cyberattacks: Assessing attack paths to critical infrastructures and services. IEEE Communications Surveys & Tutorials, 20(4), 3453-3495. https://doi.org/10.1109/COMST.2018.2855563.

[39] J.-P. Jamont and M. Occello. Meeting the challenges of decentralised embedded applications using multi-agent systems. Int. Journal of Agent- Oriented Software Engineering, 5(1):22–68, 2015.

[40] Ferber, J. (1995). Les Systèmes multi-agents : vers une intelligence collective.

[41] Ferber, J., & Gutknecht, O. (1998, July). A meta-model for the analysis and design of organizations in multi-agent systems. In Proceedings international conference on multi agent systems (Cat. No. 98EX160) (pp. 128-135). IEEE.

[42] Fox, M.S., An organizational view of distributed systems. IEEE Trans. Syst.Man. Univ. Cybern., vol. SMC-11; 1981, pp. 70-80.

[43] Ferber, J. et Mansour. S. 2007 AGRS: un modèle organisationnel pour les systèmes mu/ti- agents ouverts. LIRMM.

[44] M. Luck, P. McBurney, and C. Preist, Agent Technology: Enabling Next Generation Computing – A Roadmap for Agent Based Computing, AgentLink II (Jan. 2003).

[45] Leitão, Paulo; Karnouskos, Stamatis (March 26, 2015). Industrial agents : emerging applications of software agents in industry. Leitão, Paulo,, Karnouskos, Stamatis. Amsterdam, Netherlands.

[46] Leitao, Paulo; Karnouskos, Stamatis; Ribeiro, Luis; Lee, Jay; Strasser, Thomas; Colombo, Armando W. (2016). "Smart Agents in Industrial Cyber–Physical Systems". Proceedings of the IEEE. 104 (5): 1086–1101. doi:10.1109/JPROC.2016.2521931

[47] Kazemi, Hamidreza; Liasi, Sahand; Sheikh-El-Eslami, Mohammadkazem (November 2018). "Generation Expansion Planning Considering Investment Dynamic of Market Participants Using Multi-agent System

[48] Singh, Vijay; Samuel, Paulson (June 6, 2017). "Distributed Multi -Agent System Based Load Frequency Control for Multi- Area Power System in Smart Grid". IEEE Transactions on Industrial Electronics. 64 (6): 5151–5160. doi:10.1109/TIE.2017.2668983

[49] Brettel, M., Friederichsen, N., Keller, M., & Rosen- berg, M. (2014). How virtualization, decentral- ization and network building change the manu- facturing landscape: An industry 4.0 perspective. International Journal of Mechanical, Industrial Science and Engineering, 8(1), 37–44.
[50] Lee, J., Bagheri, B., & Kao, H.-A. (2015). A cyber- physical systems architecture for industry 4.0- based manufacturing systems. Manufacturing Letters, 3, 18–23
[51] Aliyuda, A. (2016). Towards the design of cyber-physical system via multi-agent system technology. Int. J. Sci. Eng. Res, 7(10), 155-161.
[52] President's Council of Advisors on Science and Technology (PCAST), ―Leadership under Change: Information Technology R&D in a Competitive World,‖ Retrieved from 2007. 25, 2014,June on http://www.nitrd.gov/Pcast/reports/PCAST-NIT-FINAL.pdf.
[53] Amato, A., Quarto, A., & Di Lecce, V. (2021). An application of cyber-physical system and multi-agent technology to demand-side management systems. Pattern recognition letters, 141, 23-31.
[54] Zhu, Q., Bushnell, L., & Başar, T. (2013). Resilient distributed control of multi-agent cyber-physical systems. In Control of Cyber-Physical Systems: Workshop held at Johns Hopkins University, March 2013 (pp. 301-316). Springer International Publishing.
[55] Occello, M., Jamont, J. P., Ben-Yelles, C. B., & Hoang, T. T. H. (2019). A multi-level generic multi-agent architecture for supervision of collective cyber-physical systems. International journal of autonomous and adaptive communications systems, 12(2), 109-128.
[56] Lyu, G., & Brennan, R. W. (2023). Multi-agent modelling of cyber-physical systems for IEC 61499-based distributed intelligent automation. International Journal of Computer Integrated Manufacturing, 1-27.
[57] H. Mubarak and P. Göhner, "An agent-oriented approach for selfmanagement of industrial automation systems," In Proceedings of 8th IEEE International Conference on Industrial Informatics, Osaka, Japan,July 2010, pp. 721-726.
[58] N. Cai, M. Gholami, L. Yang, and R. W. Brennan, "Application-oriented intelligent middleware for distributed sensing and control," IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, vol. 42, no. 6, pp. 947-956, Nov. 2012.
[59] M. Sahnoun, Y. Xu, B. Belgacem, B. Imen, B. David and A. Louis, "Fractal modeling of Cyber physical production system using multi-agent systems," 2019 International Conference on Applied Automation and Industrial Diagnostics (ICAAID), Elazig, Turkey, 2019, pp. 1-6, doi: 10.1109/ICAAID.2019.8934976.
[60] M. Broy, M. V. Cengarle, and E. Geisberger, 'Cyber-Physical Systems: Imminent Challenges', in Large-Scale Complex IT Systems. Development, Operation and Management, Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 1–28
[61] J. Fitzgerald, P. G. Larsen, and M. Verhoef, Collaborative Design for Embedded Systems. Berlin Heidelberg: Springer- Verlag, 2014
[62] J. Liu, Continuous Time and Mixed-Signal Simulation in Ptolemy II.TechnicalReportNo. UCB/ERL M98/74. 1998.
[63] E. A. Lee, M. Niknami, T. S. Nouidui, and M. Wetter, 'Modeling and simulating cyber-physical systems using CyPhySim', in 2015 International Conference on Embedded Software (EMSOFT), Amsterdam, Netherlands, 2015
[64] M. Burmester, E. Magkos, and V. Chrissikopoulos, 'Modeling security in cyber–physical systems', Int. J. Crit. Infrastruct. Prot., vol. 5, no. 3–4, pp. 118–126, Dec. 2012.

[65] L. Petnga and M. Austin, 'An ontological framework for knowledge modeling and decision support in cyber-physical systems', Adv. Eng. Inform., vol. 30, no. 1, pp. 77–94, Jan. 2016.
[66] S. Pourtalebi and I. Horváth, 'Information schema constructs for instantiation and composition of system manifestation features', Front. Inf. Technol. Electron. Eng., vol. 18, no. 9, pp. 1396–1415, Sep. 2017.
[67] G. Magureanu, M. Gavrilescu, D. Pescaru, and A. Doboli, 'Towards UML modeling of cyber-physical systems: A case study for gas distribution', in IEEE 8th International Symposium on Intelligent Systems and Informatics, Subotica, Serbia, 2010.
[68] E. Palachi, C. Cohen, and S. Takashi, 'Simulation of cyber physical models using SysML and numerical solvers', in 2013 IEEE International Systems Conference (SysCon), Orlando, FL, 2013.
[69] Y. Wang, 'Probabilistic modeling of information dynamics in networked cyber–physical–social systems', IEEE Internet Things J., vol. 8, no. 19, pp. 14934–14947, Oct. 2021.
[70] M. Derdour, A. Alti, M. Gasmi, and P. Roose, 'Security architecture metamodel for Model Driven security', J. Innov. Digit. Ecosyst., vol. 2, no. 1–2, pp. 55–70, Dec. 2015.
[71] Y. Wang, 'Trustworthiness in Designing Cyber-Physical Systems', in Proceedings of the 12th International Symposium on Tools and Methods of Competitive Engineering (TMCE2018), Las Palmas, Gran Canaria, Spain, 2011, pp. 27–40.
[72] V. Fritzson, 'Modelica-A unied object-oriented language for system mod- elling and simulation', in ECCOP '98: Proceedings of the 12th European Conference on Object-Oriented Programming, Springer- Verlag, 1998, pp. 67–90.
[73] H. Jifeng, From csp to hybrid systems, A Classical Mind: Essays in Honour of CAR Hoare. 1994.
[74] E. A. Lee and H. Zheng, 'Operational Semantics of Hybrid Systems', in Hybrid Systems: Computation and Control, Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 25–53.
[75] J. F. M- Y Ni, 'A co-modelling method for solving incompatibilities during co-design of mechatronic devices', Advanced Engineering Informatics, vol. 28, no. 3, pp. 232–240, 2014.
[76] J. Fitzgerald, P. G. Larsen, and M. Verhoef, Collaborative Design for Embedded Systems. Berlin Heidelberg: Springer- Verlag, 2014.
[77] MODELISAR. Functional Mock-Up Interface, Version 2.0. Interface Specification. 2017. Available online: https://fmi-standard. org/downloads/
[78] Modelica Association. System Structure and Parameterization, Version 1.0. Available online: https://ssp-standard.org
[79] Fors, N.; Hedin, G. Bloqqi: Modular feature-based block diagram programming. In Proceedings of the 2016 ACM Interna- tional Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software, Onward! Amsterdam, The Netherlands, 2–4 November 2016; pp. 57–73.
[80] Modelica Association. Modelica-A Unified Object-Oriented Language for Physical Systems Modeling-Language Specification Version 3.4. Available online: https://www.modelica.org/documents/ModelicaSpec34.pdf.
[81] Zeng, Y.; Chad, R.; Taha, W.; Duracz, A.; Atkinson, K.; Philippsen, R.; Cartwright, R.; O'Malley, M. Modeling electromechanical aspects of cyber-physical systems. J. Softw. Eng. Robot. 2016, 7, 100–119.

[82] "AGENT-0: a simple agent language and its interpreter", Shoham Y., In Proceedings of the Ninth National Conference on Artificial Intelligence, Vol II (pp. 704-709), Anaheim, CA, MIT Press, 1991.
[83] "Agent Oriented Programming", Shoham Y., Artificial Intelligence, 60(1), pp. 51-92, North-Holland, 1993.
[84] "PLACA, an Agent Oriented Programming Language", S. R. Thomas, Ph.D. Thesis, Stanford University, 1993.
[85] "A meta-model for analysis and design of multi-agent systems", J. Ferber & O. Gutknecht, Proceedings of the 3rd International Conference on Multi-Agent Systems, (ICMAS'98), IEEE, pp. 155-176, August 1998
[86] "A formal Specification of dMARS", Mark d'Inverno, David Kinny, Michael Luck, and Michael Wooldridge, In Singh et al, editors, Proceedings of the 4th International Workshop on Agent Theories, Architectures, and Languages (ATAL'97), LNAI, Vol. 1365, pp. 155- 176, Springer, 1998
[87] Nazila, G.M. (2019). Trustworthy Cyber-Physical Systems: A Systematic Framework towards Design and Evaluation of Trust and Trustworthiness. Springer Vieweg. https://doi.org/10.1007/978-3-658-27488-7
[88] Zhu, Q., Rieger, C., Başar, T. (2011). A hierarchical security architecture for cyber-physical systems. In 2011 4th International Symposium on Resilient Control Systems,USA,15-20.ID, https://doi.org/10.1109/ISRCS.2011.6016081
[89] Babiceanu, R.F., Seker, R. (2017). Trustworthiness requirements for manufacturing cyber-physical systems. Procedia Manufacturing, 11 973-981 : https://doi.org/10.1016/j.promfg.2017.07.202
[90] Anwar, R.W., Ali, S. (2012). Trust based secure cyber physical systems. In Proc. of Workshop Proceedings: Trustworthy Cyber-Physical Systems, Tech Report Series.
[91] Caldeira, F., Monteiro, E., & Simoes, P. (2010, October). Policy Based and Trust Management for Critical Infrastructure Protection. In The Carnegie Mellon Portugal Program-NET-SCIP Workshop on Security.
[92] Das, D., Banerjee, S., Chatterjee, P., Ghosh, U., Biswas, U., & Mansoor, W. (2023, January). Security, trust, and privacy management framework in cyber-physical systems using blockchain. In 2023 IEEE 20th Consumer Communications & Networking Conference (CCNC) (pp. 1-6). IEEE.
[93] Duma, C., Karresand, M., Shahmehri, N., Caronni, G. (2006). A trust-aware, p2p-based overlay for intrusion detection. In 17th International Workshop on Database and Expert Systems Applications (DEXA'06), Krakow, Poland, 692-697. pp. https://doi.org/10.1109/DEXA.2006.21
[94] Jiang, Y., Wu, S., Yang, H., Luo, H., Chen, Z., Yin, S., Kaynak, O. (2022). Secure data transmission and trustworthiness judgement approaches against cyber- physical attacks in an integrated data-driven framework. IEEE Transactions on Systems, Man, and Cybernetics: 52(12): Systems, https://doi.org/10.1109/TSMC.2022.3164024.
[95] Harlamova, M., Kirikova, M. (2018). Trust handling framework for networks in cyber physical systems of Industry 4.0. In: Zdravkovic, J., Grabis, J., Nurcan, S., Stirna, J. (eds) Perspectives in Business Informatics Research. BIR 2018. Lecture Notes in Business Information Processing, vol 330. Springer, Cham. https://doi.org/10.1007/978-3-319-99951-7_3
[96] Spensky, C., Machiry, A., Busch, M., Leach, K., Housley, R., Kruegel, C., Vigna, G. (2020). TRUST. IO: protecting physical interfaces on cyber-physical systems. In 2020

IEEE Conference on Communications and Network Security (CNS), Avignon, France, pp. 1-9. https://doi.org/10.1109/CNS48642.2020.9162246

[97] Zhao, J., Zhang, W., & Liu, J. Contextual Sequence Labeling Model Integrating Symbolic Rules for Extracting Indicators of Compromise. Available at SSRN 4496393

[98] Orojloo, Modelling and evaluation of the security of cyber-physical systems using stochastic Petri nets, IET Cyber-Phys. Syst., Theory Appl., № 4, c. 50. https://doi.org/10.1049/iet-cps.2018.0008.

[99] De Florio, V., & Primiero, G. (2015). A framework for trustworthiness assessment based on fidelity in cyber and physical domains. Procedia Computer Science, 52, 996-1003.

[100] Lochner, M.; Duenser, A.; Sarker, S. Trust and cognitive load in semi-automated UAV operation. In Proceedings of the 31st Australian Conference on Human-Computer-Interaction (OZCHI'19), Association for Computing Machinery, New York, NY, USA, 2–5 December 2019; pp. 437–441.

[101] Barka, E.; Kerrache, C.A.; Hussain, R.; Lagraa, N.; Lakas, A.; Bouk, S.H. A trusted lightweight communication strategy for flying named data networking. Sensors 2018, 18, 2683.

[102] Singh, K.; Verma, A.K. A trust model for effective cooperation in flying ad hoc networks using genetic algorithm. In Proceedings of the 2018 International Conference on Communication and Signal Processing (ICCSP 2018), Chennai, India, 3–5 April 2018; pp. 491–495.

[103] A. Rajan and T. Wahl, CESAR: Cost-Efficient Methods and Processes for Safety-Relevant Embedded Systems. New York, NY, USA: Springer, 2013.

[104] Zahid, A. Tanveer, M. M. Kuo, and R. Sinha, "A systematic mapping of semi-formal and formal methods in requirements engineering of industrial cyber-physical systems," Journal of Intelligent Manufacturing, vol. 33, no. 6, pp. 1603–1638, 2022.

[105] C. J. Budnik, S. Eckl, and M. Gario, "Testbed for model-based veri- fication of cyber-physical production systems." in ARCH@ CPSWeek, 2017, pp. 92–99.

[106] S. Yang, "Review on testing of cyber physical systems: Methods andtestbeds.

[107] Oudina, Z., Derdour, M., & Bouhamed, M. M. (2022, October). Testing cyber-physical production system: Test methods categorization and dataset. In 2022 4th International Conference on Pattern Analysis and Intelligent Systems (PAIS) (pp. 1-8). IEEE.

[108] Hahn, A.; Ashok, A.; Sridhar, S.; Govindarasu, M. Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid. IEEE Trans. Smart Grid 2013, 4, 847–855.

[109] Hahn, A.; Govindarasu, M. An evaluation of cybersecurity assessment tools on a SCADA environment. In Proceedings of the 2011 IEEE Power and Energy Society General Meeting, Detroit, MI, USA, 24–28 July 2011; pp. 1–6.

[110] Sun, C.C.; Hong, J.; Liu, C.C. A co-simulation environment for integrated cyber and power systems. In Proceedings of the 2015 IEEE International Conference on Smart Grid Communications (SmartGridComm), Miami, FL, USA, 2–5 November 2015; pp. 133–138.

[111] Vellaithurai, C.B.; Biswas, S.S.; Srivastava, A.K. Development and application of a real-time test bed for cyber–physical system. IEEE Syst. J. 2015, 11, 2192–2203.

[112] Sun, C.C.; Hong, J.; Liu, C.C. A coordinated cyber attack detection system (CCADS) for multiple substations. In Proceedings of the 2016 Power Systems Computation Conference (PSCC), Genoa, Italy, 20–24 June 2016; pp. 1–7.

[113] Biswas, S.S.; Kim, J.H.; Srivastava, A.K. Development of a smart grid test bed and applications in PMU and PDC testing. In Proceedings of the 2012 North American Power Symposium (NAPS), Champaign, IL, USA, 9–11 September 2012; pp. 1–6.
[114] Liu, R.; Vellaithurai, C.; Biswas, S.S.; Gamage, T.T.; Srivastava, A.K. Analyzing the cyber-physical impact of cyber events on the power grid. IEEE Trans. Smart Grid 2015, 6, 2444–2453
[115] Stanovich, M.J.; Leonard, I.; Sanjeev, K.; Steurer, M.; Roth, T.P.; Jackson, S.; Bruce, M. Development of a smart-grid cyber-physical systems testbed. In Proceedings of the 2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 24–27 February 2013; pp. 1–6.
[116] Ashok, A.; Sridhar, S.; McKinnon, A.D.; Wang, P.; Govindarasu, M. Testbed-based performance evaluation of attack resilient control for agc. In Proceedings of the 2016 Resilience Week (RWS), Chicago, IL, USA, 16–18 August 2016; pp. 125–129.
[117] Hong, J.; Liu, C.C.; Govindarasu, M. Detection of cyber intrusions using network-based multicast messages for substation automation. In Proceedings of the ISGT 2014, Washington, DC, USA, 19–22 February 2014; pp. 1–5.
[118] R. Akella and B. M. McMillin, "Model-checking BNDC properties in cyber-physical systems," in Proc. 33rd Annu. IEEE Int. Comput. Softw.Appl. Conf. (COMPSAC), 2009, vol. 1, pp. 660–663
[119] M. Xia, M. Sun, G. Luo, and X. Zhao, "Design and implementation of automatic verification for PLC systems," in Proc. 12th IEEE Int. Conf. Cogn. Inform. Cogn. Comput., 2013, pp. 374–379.
[120] M. Perin and J.-M. Faure, "Building meaningful timed models of closed- loop DES for verification purposes," Control Eng. Pract., vol. 21, no. 11, pp. 1620–1639, 2013
[121] D. Soliman, K. Thramboulidis, and G. Frey, "Function block diagram to UPPAAL timed automata transformation based on formal models," Inf. Control Probl. Manuf., vol. 14, no. 1, pp. 1653–1659, 2012.
[122] F Zahid, Awais Tanveer," A systematic mapping of semi-formal and formal methods in requirements engineering of industrial Cyber-Physical systems", in 2021Journal of Intelligent Manufacturing.
[123] Kossel, R., Tegethoff, W., Bodmann, M., and Lemke, N.(2006). Simulation of complex systems using modelica and tool coupling. In 5th Modelica Conference, volume 2, 485–490
[124] A Arrietaa, S Wang," Pareto efficient multi-objective black-box test case selection for simulation-based testing", 2019in Information and Software Technology 114
[125] V Brandstetter,J C Wehrstedt, "A Framework for Multidisciplinary Simulation of Cyber-Physical Production Systems" in Part of special issue:16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018: Bergamo, Italy, 11–13 June 2018.
[126] R. Malhotra, —Study and Comparison of Various Cloud Simulators Available in the Cloud Computing,‖ SIJ Trans. Comput. Sci. Eng. its Applications (CESA), vol. 1, no. 3, 2013.
[127] Choi, S.; Woo, J.; Kim, J.; Lee, J.Y. Digital twin-based integrated monitoring system: Korean application cases. Sensors 2022, 22, 5450.
[128] Amini, A., Kanfound, J., & Gan, T. H. (2019, October). An ai driven real-time 3-D representation of an off-shore WT for fault diagnosis and monitoring. In Proceedings of the 3rd International Conference on Advances in Artificial Intelligence (pp. 162-165).

[129] Peng, C. C., & Chen, Y. H. (2021). Digital twins-based online monitoring of TFE-731 turbofan engine using Fast orthogonal search. IEEE Systems Journal, 16(2), 3060-3071.
[130] Yu, Q., Huang, Y., Liu, Y., Yu, S., & Wang, S. (2021, January). Research on application of information model in wind turbine fault diagnosis. In Proceedings of the 2021 2nd International Conference on Artificial Intelligence in Electronics Engineering (pp. 67-74).
[131] Xu, Q., Ali, S., & Yue, T. (2021, April). Digital twin-based anomaly detection in cyber-physical systems. In 2021 14th IEEE Conference on Software Testing, Verification and Validation (ICST) (pp. 205-216). IEEE.
[132] Xu, Y., Sun, Y., Liu, X., & Zheng, Y. (2019). A digital-twin-assisted fault diagnosis using deep transfer learning. Ieee Access, 7, 19990-19999.
[133] Yoginath, S., Tansakul, V., Chinthavali, S., Taylor, C., Hambrick, J., Irminger, P., & Perumalla, K. (2019, November). On the effectiveness of recurrent neural networks for live modeling of cyber-physical systems. In 2019 IEEE International Conference on Industrial Internet (ICII) (pp. 309-317). IEEE.
[134] Rajkumar, R., Lee, I., Sha, L., & Stankovic, J. (2010, June). Cyber-physical systems: the next computing revolution. In Proceedings of the 47th design automation conference (pp. 731-736).
[135] Zhou, X., Gou, X., Huang, T., & Yang, S. (2018). Review on testing of cyber physical systems: Methods and testbeds. IEEE Access, 6, 52179-52194.
[136] E. A. Lee, CyberPhysicalSystems:DesignChallenges. 2008
[137] S. Ali and T. Yue, ''U-test: Evolving, modelling and testing realisticuncertain behaviours of cyber-physical systems,'' in Proc. IEEE Int. Conf.Softw. Test., Verification Validation, Apr. 2015, pp. 1–2
[138] B. Aminian, J. Araújo, M. Johansson, and K. H. Johansson, ''GISOO:A virtual testbed for wireless cyber-physical systems,'' in Proc. IECON,vol. 20, 2013, pp. 5588–5593.
[139] Li and R. Kang, ''Strategy for reliability testing and evaluation ofcyber physical systems,'' in Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage.,Dec. 2016, pp. 1001–1006
[140] Sarvapali D Ramchurn, Dong Huynh, and Nicholas R Jennings. 2004. Trust in multi-agent systems. The knowledge engineering review 19, 1 (2004), 1–25.
[141] Xiaowei Huang, Marta Kwiatkowska, and Maciej Olejnik. 2019. Reasoning about cognitive trust in stochastic multiagent systems. ACM Transactions on Computational Logic (TOCL) 20, 4 (2019), 1–64.
[142] Sarvapali D Ramchurn, Nicholas R Jennings, Carles Sierra, and Lluis Godo. 2004. Devising a trust model for multi-agent interactions using confidence and reputa- tion. Applied Artificial Intelligence 18, 9-10 (2004), 833–852
[143] J. Bentahar, N. Drawel, and A. Sadiki. 2022. Quantitative Group Trust: A Two- Stage Verification Approach. In Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS'22). International Foun- dation for Autonomous Agents and Multiagent Systems, Auckland, New Zealand, 100–108.
[144] David Harel, Dexter Kozen, and Jerzy Tiuryn. 2000. Dynamic Logic. MIT press.
[145] Mingxi Cheng, Chenzhong Yin, Junyao Zhang, Shahin Nazarian, Jyotirmoy Deshmukh, and Paul Bogdan. 2021. A General Trust Framework for Multi-Agent Systems. In AAMAS. 332–340
[146] Zikratov, I. A., Viktorovna, Z. T., Lebedev Ilya, S., & Gurtov, A. V. (2014). Trust and reputation model design for objects of multi-agent robotics systems with decentralized control. Journal Scientific and Technical Of Information Technologies, Mechanics and Optics, 91(3), 30-38.

[147] Khamis, M. A., & Nagi, K. (2013). Designing multi-agent unit tests using systematic test design patterns-(extended version). Engineering Applications of Artificial Intelligence, 26(9), 2128-2142.
[148] Tiryaki, A. M., Öztuna, S., Dikenelli, O., & Erdur, R. C. (2007). Sunit: A unit testing framework for test driven development of multi-agent systems. In Agent-Oriented Software Engineering VII: 7th International Workshop, AOSE 2006, Hakodate, Japan, May 8, 2006, Revised and Invited Papers 7 (pp. 156-173). Springer Berlin Heidelberg.
[149] [149] Mikulski, D. G., Lewis, F. L., Gu, E. Y., & Hudas, G. R. (2012, May). Trust method for multi-agent consensus. In Unmanned systems technology xiv (Vol. 8387, pp. 146-159). SPIE.
[150] Aref, A., & Tran, T. (2015, April). A trust establishment model in multi-agent systems. In Workshops at the twenty-ninth aaai conference on artificial intelligence.
[151] Rosaci, D., Sarné, G. M., & Garruzzo, S. (2012). Integrating trust measures in multiagent systems. International Journal of Intelligent Systems, 27(1), 1-15.
[152] Akintunde, M., Yazdanpanah, V., Salehi Fathabadi, A., Cirstea, C., Dastani, M., & Moreau, L. (2024). Actual Trust in Multiagent Systems.
[153] Sarvapali D Ramchurn, Nicholas R Jennings, Carles Sierra, and Lluis Godo. 2004. Devising a trust model for multi-agent interactions using confidence and reputa- tion. Applied Artificial Intelligence 18, 9-10 (2004), 833–852.
[154] Teacy, W. L., Patel, J., Jennings, N. R., & Luck, M. (2006). Travos: Trust and reputation in the context of inaccurate information sources. Autonomous Agents and Multi-Agent Systems, 12, 183-198.
[155] Oudina, Z., Derdour, M., Boudour, R., Dib, A., & Yakoubi, M. A. (2023). Trust cyber physical systems: Trust degree framework and evaluation. Journal homepage: http://iieta. org/journals/ijsse, 13(2), 213-225.
[156] Piètre-Cambacédès, L., Bouissou, M. (2013). Cross- fertilization between safety and security engineering. Reliability Engineering & System Safety, 110: 110-126. https://doi.org/10.1016/j.ress.2012.09.011
[157] Furrer, F.J. (2022). Cyber-Physical Systems. In Safety and Security of Cyber-Physical Systems: Engineering dependable Software using Principle-based Development, pp. 9-76. Wiesbaden: Springer Fachmedien Wiesbaden.
[158] Fletcher, K.K., Liu, X. (2011). Security requirements analysis, specification, prioritization and policy development in cyber-physical systems. In 2011 Fifth International Conference on Secure Software Integration and Reliability Improvement-Companion, Jeju, Korea (South), pp. 106-113. https://doi.org/10.1109/SSIRI- C.2011.25
[159] Piètre-Cambacédès, L., Bouissou, M. (2010). Modeling safety and security interdependencies with BDMP (Boolean logic Driven Markov Processes). In 2010 IEEE International Conference on Systems, Man and Cybernetics, Istanbul, Turkey, pp. 2852-2861. https://doi.org/10.1109/ICSMC.2010.5641922
[160] Hollnagel, E. (2018). Safety-I and Safety-II: The Past and Future of Safety Management. CRC Press
[161] Williamson, O.E. (1993). Calculativeness, trust, and economic organization. The Journal of Law and Economics, 2):453-486. https://doi.org/10.1086/467284
[162] Avizienis, A., Laprie, J.C., Randell, B., Landwehr, C. (2007). Basic Concepts and Taxonomy of Dependable Secure Computing. In A Process for Developing a Common Vocabulary in the Information Security Area, pp. 10-51. IOS Press.

[163] abaliauskaite, G., Mathur, A.P. (2015). Aligning cyber- physical system safety and security. In: Cardin, MA., Krob, D., Lui, P., Tan, Y., Wood, K. (eds) Complex Systems Design & Management Asia. Springer, Cham. https://doi.org/10.1007/978-3-319-12544-2_4
[164] Oudina, Z., Derdour, M., Dib, A., Yaakoubi, M.A. (2024). Identifying and addressing trust concerns in cyber-physical systems for the oil and gas industry. Ingénierie des Systèmes d'Information, Vol. 29, No. 2, pp. 469-478. https://doi.org/10.18280/isi.290208
[165] Mohammadi, N.G., Paulus, S., Bishr, M., Metzger, A., Koennecke, H., Hartenstein, S., Pohl, K. (2013). An analysis of software quality attributes and their contribution to trustworthiness. In CLOSER, pp. 542- 552. https://doi.org/10.5220/0004502705420552
[166] Kalloniatis, C., Kavakli, E., Gritzalis, S. (2008). Addressing privacy requirements in system design: The PriS method. Requirements Engineering, 13: 241-255. https://doi.org/10.1007/s00766-008-0067-3
[167] Oudina, Z., & Derdour, M. (2023). Toward Modeling Trust Cyber-Physical Systems: A Model-based System Engineering Method. International Journal of Advanced Computer Science and Applications, 14(7).
[168] A. M. Madni and M. Sievers, "Model-based systems engineering: Motivation, current status, and research opportunities," Systems Engineering, vol. 21, no. 3, pp. 172–190, 2018, doi: 10.1002/sys.21438.
[169] Morkevicius, A. Aleksandraviciene, D. Mazeika, L. Bisikirskiene, and Z. Strolia, 'MBSE grid: A simplified SysML-based approach for modeling complex systems', INCOSE Int. Symp., vol. 27, no. 1, pp. 136–150, Jul. 2017.
[170] 'International Council on Systems Engineering. Systems Engineering Handbook; Version 3.1; International Council on Systems Engineering',2007.
[171] A. Madni and S. Purohit, 'Economic analysis of model-based systems engineering', Systems, vol. 7, no. 1, p. 12, Feb. 2019.
[172] Akintunde, M., Yazdanpanah, V., Salehi Fathabadi, A., Cirstea, C., Dastani, M., & Moreau, L. (2024). Actual Trust in Multiagent Systems.
[173] Sarvapali D Ramchurn, Nicholas R Jennings, Carles Sierra, and Lluis Godo. 2004. Devising a trust model for multi-agent interactions using confidence and reputa- tion. Applied Artificial Intelligence 18, 9-10 (2004), 833–852.
[174] Teacy, W. L., Patel, J., Jennings, N. R., & Luck, M. (2006). Travos: Trust and reputation in the context of inaccurate information sources. Autonomous Agents and Multi-Agent Systems, 12, 183-198.
[175] D. Mažeika and R. Butleris, 'Integrating security requirements engineering into MBSE: Profile and guidelines', Secur. Commun. Netw., vol. 2020, pp. 1–12, Mar. 2020.
[176] Elmenreich. Intelligent methods for embedded systems. In Proceedings of the First Workshop on Intelligent Solutions in Embedded Systems, pages 3–11, 2003.
[177] M. Occello and Y. Demazeau. Une approche du temps réel dans la conception d'agents. In J. Muller and J. Quinqueton, editors, 4 èmes Journées Francophones IAD-SMA, pages 101–112, Port Camargue, France, Avril 1996. Hermès.
[178] Botti, V., Barber, F., Crespo, A., Onaindia, E., Garcia-Fornes, A., Ripoll, I., Gallardo, D., and Hernndez, A temporal black-board for a multi-agent envirnment, Data and Knwoledge Engineering 15 (1995), 189-211
[179] Baker, C.T., Paul, C.A., Willé, D.R. (1995). Issues in the numerical solution of evolutionary delay differential equations. Advances in Computational Mathematics, 3: 171-196. https://doi.org/10.1007/BF02988625

[180] Hector, A. and Lakshmi Narasimhan, V. (2005). —A New Classification Scheme for Software Agents.‖ Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05), pp. 191 – 196, ISBN: 0-7695-2316-1, Sydney, Australia, July 2005, IEEE Computer Society, Washington, DC

[181] Faisal Alkhateeb, Eslam Al Maghayreh and Iyad Abu Doush (2011). —Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications‖. Janeza Trdine 9, 51000 Rijeka, Croatia. ISBN 978-953-307- 174-9, pp. 7, 327 – 328

[182] Sebastian Huhn, Heike Sonnenberg, Stephan Eggersglüß, Brigitte Clausen, and Rolf Drechsler. Revealing properties of structural materials by combining regression-based algorithms and nano indentation measurements. In IEEE Symposium Series on Computational Intelligence, 2017

[183] Oudina, Z., Derdour, M., Dib, A., & Tachouche, A. M. A. (2023, October). Model Based System Engineering for trust SCADA and ICS Systems in Oil & Gas Industry. In 2023 5th International Conference on Pattern Analysis and Intelligent Systems (PAIS) (pp. 1-8). IEEE.

[184] K. Erickson, A. Miller, E. Stanek, and S. Dunn-Norman, "Survey ofscada system technology and reliability in the offshore oil and gasindustry," MMS TA&R Program Program SOL, pp. 1435–01, 2000.

[185] Aga 12 cryptographic protection of SCADA Communications.Available at: https://icscsi.org/library/Documents/Standards/AGA -Cryptographic Protection of SCADA Communications - 12 Part1.pdf.

[186] F. A. Asa, "Application of MBSE to oil and gas project/productmanagement cycle: a model-based development approach forengineering management and design," PhD Thesis, MassachusettsInstitute of Technology, 2020.

[187] Oudina, Z., Dib, A., Yakoubi, M. A., & Derdour, M. (2024). Comprehensive Risk Classification and Mitigation in the Petroleum Cyber-Physical Systems of the Oil and Gas Industry. International Journal of Safety & Security Engineering, 14(1).

[188] Krol D., Zelmozer M., Structural Performance Evaluation of Multi-Agent Systems, Journal of Universal Computer Science, Springer-Verlag Berlin / Heidelberg, 2008, 14,1154-1178.

[189] Mecibah, Z., & Mokhati, F. (2012). Génération des diagrammes auml à partir de programme jade.

[190] Z. Oudina, M. Derdour, A. Dib and M. M. Bouhamed, "Empirical Analysis of the Security Threats and Risks that Drones Face, Represent, and Mitigation," 2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS), EL OUED, Algeria, 2024, pp. 1-8, doi: 10.1109/PAIS62114.2024.10541193