



A Novel Multi-Level and Community-Based Agent Ecosystem to Support Customers Dynamic Decision-Making in Smart Grids

HELDER JORGE VILARINHO PEREIRA

Novembro de 2022



A Novel Multi-Level and Community-Based Agent Ecosystem to Support Customer's Dynamic Decision-Making in Smart Grids

Helder Jorge Vilarinho Pereira

1170836

Dissertation to obtain the Master's degree in Artificial Intelligence Engineering

Supervisor: Doctor Zita Vale, Principal Coordinator Professor, Polytechnic of Porto - School of Engineering

Co-supervisor: Doctor Luis Gomes, Polytechnic of Porto - School of Engineering

Jury

President:

Doctor Luiz Felipe Rocha de Faria, Coordinator Professor, Polytechnic of Porto - School of Engineering

Vocals:

Doctor Pedro José Ramos Moreira de Campos, Assistant Professor, University of Porto - Faculty of Economics

Doctor Zita Maria Almeida do Vale, Principal Coordinator Professor, Polytechnic of Porto - School of Engineering

Porto, October 2022

«If you set your goals ridiculously high and it's a failure, you will fail above everyone else's success.»

James Cameron

Abstract

Electrical systems have evolved at a fast pace over the past years, particularly in response to the current environmental and climate challenges. Consequently, the European Union and the United Nations have encouraged the development of a more sustainable energy strategy. This strategy triggered a paradigm shift in energy consumption and production, which becoming increasingly distributed, resulted in the development and emergence of smart energy grids.

Multi-agent systems are one of the most widely used artificial intelligence concepts in smart grids. Both multi-agent systems and smart grids are distributed, so there is correspondence between the used technology and the network's complex reality. Due to the wide variety of multi-agent systems applied to smart grids, which typically have very specific goals, the ability to model the network as a whole may be compromised, as communication between systems is typically non-existent. This dissertation, therefore, proposes an agent-based ecosystem to model smart grids in which different agent-based systems can coexist.

This dissertation aims to conceive, implement, test, and validate a new agent-based ecosystem, entitled A4SG (agent-based ecosystem for smart grids modelling), which combines the concepts of multi-agent systems and agent communities to enable the modelling and representation of smart grids and the entities that compose them. The proposed ecosystem employs an innovative methodology for managing static or dynamic interactions present in smart grids. The creation of a solution that allows the integration of existing systems into an ecosystem, enables the representation of smart grids in a realistic and comprehensive manner.

A4SG integrates several functionalities that support the ecosystem's management, also conceived, implemented, tested, and validated in this dissertation. Two mobility functionalities are proposed: one that allows agents to move between physical machines and another that allows "virtual" mobility, where agents move between agent communities to improve the context for the achievement of their objectives. In order to prevent an agent from becoming overloaded, a novel functionality is proposed to enable the creation of agents that function as extensions of the main agent (i.e., branch agents), allowing the distribution of objectives among the various extensions of the main agent.

Several case studies, which test the proposed services and functionalities individually and the ecosystem as a whole, were used to test and validate the proposed solution. These case studies were conducted in realistic contexts using data from multiple sources, including energy communities. The results indicate that the used methodologies can increase participation in demand response events, increasing the fitting between consumers and aggregators from 12 % to 69 %, and improve the strategies used in energy transaction markets, allowing an energy community of 50 customers to save 77.0 EUR per week.

Keywords: agent communities, agents' mobility, energy aggregators, energy customers' active participation, multi-agent systems, reinforcement learning

Resumo

Os últimos anos têm sido de mudança nos sistemas elétricos, especialmente devido aos atuais desafios ambientais e climáticos. A procura por uma estratégia mais sustentável para o domínio da energia tem sido promovida pela União Europeia e pela Organização das Nações Unidas. A mudança de paradigma no que toca ao consumo e produção de energia, que acontece, cada vez mais, de forma distribuída, tem levado à emergência das redes elétricas inteligentes.

Os sistemas multi-agente são um dos conceitos, no domínio da inteligência artificial, mais aplicados em redes inteligentes. Tanto os sistemas multi-agente como as redes inteligentes têm uma natureza distribuída, existindo por isso um alinhamento entre a tecnologia usada e a realidade complexa da rede. Devido a existir uma vasta oferta de sistemas multi-agente aplicados a redes inteligentes, normalmente com objetivos bastante específicos, a capacidade de modelar a rede como um todo pode ficar comprometida, porque a comunicação entre sistemas é, geralmente, inexistente. Por isso, esta dissertação propõe um ecossistema baseado em agentes para modelar as redes inteligentes, onde vários sistemas de agentes coexistem.

Esta dissertação pretende conceber, implementar, testar, e validar um novo ecossistema multi-agente, intitulado A4SG (*agent-based ecosystem for smart grids modelling*), que combina os conceitos de sistemas multi-agente e comunidades de agentes, permitindo a modelação e representação de redes inteligentes e das suas entidades. O ecossistema proposto utiliza uma metodologia inovadora para gerir as interações presentes nas redes inteligentes, sejam elas estáticas ou dinâmicas. A criação de um ecossistema que permite a integração de sistemas já existentes, cria a possibilidade de uma representação realista e detalhada das redes de energia.

O A4SG integra diversas funcionalidades, também estas concebidas, implementadas, testadas, e validadas nesta dissertação, que suportam a gestão do próprio ecossistema. São propostas duas funcionalidades de mobilidade, uma que permite aos agentes mover-se entre máquinas físicas, e uma que permite uma mobilidade “virtual”, onde os agentes se movem entre comunidades de agentes, de forma a melhorar o contexto para a execução dos seus objetivos. É também proposta uma nova funcionalidade que permite a criação de agentes que funcionam como uma extensão de um agente principal, com o objetivo de evitar a sobrecarga de um agente, permitindo a distribuição de objetivos entre as várias extensões do agente principal.

A solução proposta foi testada e validada por vários casos de estudo, que testam os serviços e funcionalidades propostas individualmente, e o ecossistema como um todo. Estes casos de estudo foram executados em contextos realistas, usando dados provenientes de diversas fontes, tais como comunidades de energia. Os resultados demonstram que as metodologias utilizadas podem melhorar a participação em eventos de *demand response*, subindo a adequação entre consumidores e agregadores de 12 % para 69 %, e melhorar as estratégias utilizadas em mercados de transações de energia, permitindo a uma comunidade de energia com 50 consumidores poupar 77,0 EUR por semana.

Palavras-chave: agregadores de energia, aprendizagem por reforço, comunidades de agentes, mobilidade de agentes, participação ativa dos clientes de energia, sistemas multi-agente

Acknowledgements

I would like to thank everyone who contributed, directly or indirectly, to this work's success.

Firstly, I would like to thank my supervisors, Professor Zita Vale and Doctor Luis Gomes, for their guidance and support throughout the development of this work, as well as for the time and effort they invested in advising and revising my work. If this work is successful, I owe a great deal to the wisdom I have gained from engaging conversations and discussions with both.

In addition, I would like to thank my GECAD co-workers, who played a crucial role in the development of this work. I am confident that all the hours of work together had a positive impact on the outcomes of this work.

I would not be where I am today without the encouragement, endless conversations, and great laughs of my family and friends, who kept me on the right path and encouraged me to persevere. I owe you a great deal, Mom, Dad, Sister, Manel, Telmo, Nuno, and my college friends, Danilos, whom I will carry with me for the rest of my life.

Last, but not least, my Joana. Not every moment in life must be serious, and if anyone can shine and make those around them shine, it is you. Because you never let me down, even when I do not believe in myself, you have my deepest gratitude. Never change, you are unique!

Table of Contents

1	Introduction	1
1.1	Contextualization	1
1.2	Research Question and Objectives.....	3
1.3	Scientific Contributions	5
1.4	Document Organization	8
2	State of the Art	9
2.1	Smart Grids and Energy Customers' Participation	9
2.1.1	Transactive Energy	11
2.1.2	Demand Response	12
2.1.3	Aggregators	13
2.1.4	Energy Communities	15
2.2	Agents and Multi-Agent Systems	16
2.2.1	Organizational Paradigms	19
2.2.2	Agents' Mobility	20
2.2.3	Agents' Intelligence and Learning	21
2.2.4	Interoperability.....	23
2.2.5	Agent-based Platforms and Frameworks	24
2.3	Machine Learning.....	25
2.3.1	Supervised Learning.....	26
2.3.2	Unsupervised Learning.....	28
2.3.3	Reinforcement Learning.....	30
2.4	Chapter Conclusions	32
3	Methods and Materials	33
3.1	Materials and Tools	33
3.1.1	Multi-Agent Systems Development Frameworks	33
3.1.1	Python and Machine Learning Libraries	35
3.1.2	Datasets	36
3.2	Methods and Models	38
3.2.1	Customers' Aggregation	38
3.2.2	Multi-Agent System's Organizational Paradigms	39
3.2.3	Energy Peer-to-Peer Trading Markets Participation Optimization.....	40
3.3	Technological and Social Challenges	41
3.3.1	Data Privacy and Protection	42
3.3.2	Agents and Data Security.....	43
3.3.3	Ethical and Social Issues.....	44
3.4	Chapter Conclusions	46
4	Agent-based Ecosystem for Smart Grid Modelling - A4SG	47
4.1	Architecture	48

4.1.1	Citizen Energy Communities Operator System Layer.....	49
4.1.2	Common Services Layer	51
4.1.3	A4SG Agents Layer	52
4.1.4	Customer/Entity Side.....	52
4.2	Agent Types.....	53
4.2.1	Ecosystem Agents	53
4.2.2	Representation Agents	54
4.2.3	Branch Agents	55
4.2.4	Agents Characteristics in A4SG.....	56
4.3	Tools and Services for A4SG	57
4.3.1	Static Code Analysis	57
4.3.2	Translator Interface for Different Communication Protocols.....	58
4.3.3	Group Selector.....	59
4.4	Novel Functionalities for Multi-Agent Systems.....	60
4.4.1	Physical Mobility and Computing Load Balancing.....	60
4.4.2	Virtual Mobility	62
4.4.3	Branching.....	65
4.4.4	Objective Interdependency.....	67
4.4.5	Intra-agent Conflict Resolution	68
4.5	Chapter Conclusions	70
5	Service-based Agents for Smart Grid Business Models	71
5.1	Energy Demand Forecast.....	71
5.2	Electric Network Analysis	72
5.3	Demand Response.....	73
5.3.1	Price-based Program.....	73
5.3.2	Incentive-based Program.....	74
5.4	Reinforcement Learning for Peer-to-Peer Transactions	74
5.5	Chapter Conclusions	78
6	Case Studies	79
6.1	Smart Grid Services Integration in Citizen Energy Community Operator System ...	79
6.2	Customer-Aggregator Fitting for Demand Response Efficient Participation	83
6.3	Optimize Participation in Peer-to-Peer Markets Using Reinforcement Learning....	87
6.4	Smart Building Participation in Peer-to-Peer Markets Considering Energy Storage Systems.....	91
6.5	Chapter Conclusions	95
7	Conclusions	97
7.1	Main Conclusions	97
7.2	Future Work.....	99
	References	101

List of Figures

Figure 1 – Demand Response programs classification. Adapted from (Jordehi, 2019)	13
Figure 2 - Relationships established by the aggregator. Adapted from (Ottesen et al., 2018) .	14
Figure 3 – Agents’ physical mobility.....	20
Figure 4 – Agent-Environment interaction	22
Figure 5 – Comparison between supervised, semi-supervised, and unsupervised learning data labelling (adapted from (Schmarje et al., 2020))	26
Figure 6 – Clustering example. Extracted from (Varghese et al., 2018)	29
Figure 7 – Comparison between a) Q-Learning, and b) Deep Q-Learning. Adapted from (Choudhary, 2019)	31
Figure 8 – PEAK framework architecture	34
Figure 9 – Dimensionality reduction exemplification. Extracted from (Karim, 2019)	39
Figure 10 – Comparison between (a) data-shipping, and (b) code-shipping	44
Figure 11 – Representation of the A4SG architecture	49
Figure 12 – Centralized CECOS data flow (Pereira et al., 2021).....	50
Figure 13 – Decentralized CECOS architecture (Pereira et al., 2022c).....	51
Figure 14 – A4SG agent types and a brief description	53
Figure 15 - Example of the representation of a customer, considering the agents’ branching	55
Figure 16 – Static Code Analysis flowchart	58
Figure 17 – Exemplification of the Computing Load Balancing functionality	62
Figure 18 – Exemplification of the Virtual Mobility functionality	63
Figure 19 – Virtual Mobility functionality - options filtering flowchart	64
Figure 20 – Branching functionality flowchart.....	65
Figure 21 – Example of the representation of a service provider in A4SG, considering the agents’ branching functionality	66
Figure 22 – Communication between branch agents before the implementation of the Objective Interdependency functionality	67
Figure 23 – Communication between branch agents after the implementation of the Objective Interdependency functionality.....	68
Figure 24 – Intra-agent Conflict Resolution functionality flowchart.....	69
Figure 25 – Physical energy network graphic visualization (exported from CECOS)	73
Figure 26 – RL environment and agents interaction in training	75
Figure 27 – Most profitable tariff comparison with the original tariff (exported from CECOS)	80
Figure 28 – Customers’ aggregation evaluation metrics (exported from CECOS)	80
Figure 29 – Customers’ aggregation results for three clusters (exported from CECOS)	81
Figure 30 – A4SG Agents Layer architecture considering the case study structure, between week 9 and 10 of simulation (i.e., agents and groups)	85
Figure 31 – Percentage of events in which the aggregated reduction target was achieved for each aggregator per week	86
Figure 32 – Customers’ flexibility fitting rate area (minimum and maximum) and average	86
Figure 33 – P2P participation optimization case study steps	88

Figure 34 – Mean reward for each agent per training episode: (a) DDPG, and (b) TD3	89
Figure 35 – Maximum, mean, and minimum reward per episode: (a) DDPG, and (b) TD3	89
Figure 36 – Comparison of the energy costs during the week before and the week after the RL training, for both RL algorithms and the agents without training	90
Figure 37 – Energy transacted in the P2P market before and after the RL training	90
Figure 38 – Ecosystem structure (extracted and adapted from the PEAK graphical interface) representing the names and the number of agents of agent communities: a) agent communities at week 1 showing the distribution of agents among the forecast providers, and b) agent communities at week 3 showing the distribution of agents among the forecast providers.	93
Figure 39 – Forecast services comparison for one of the studied energy community members	94
Figure 40 – Energy costs in the second and third weeks compared to the previous week	94
Figure 41 – ESS load (state-of-charge) comparison between weeks for each studied period ..	95

List of Tables

Table 1 – Research Questions and Corresponding Objectives	5
Table 2 – Comparison between the traditional power system and the smart grid. Adapted from (Farhangi, 2010)	10
Table 3 – Applications of multi-agent systems to smart grids.....	18
Table 4 – MAS organizational paradigms. Adapted from (Horling and Lesser, 2004)	19
Table 5 – Deep Learning models comparison. Adapted from (Pouyanfar et al., 2018).....	28
Table 6 – Description of the first used dataset’s features	37
Table 7 – Integration of agents’ characteristics into A4SG agents	56
Table 8 – Overview of proposed functionalities	60
Table 9 – Proposed case studies summary	79
Table 10 – Demand Response simulation considering power flow results for days 2 and 3.....	82
Table 11 – Customers’ description and mobilities for each aggregator per week	85
Table 12 – Ecosystem structure to support the case study: groups and agents	92
Table 13 – Forecast services groups’ customers and errors for each week of simulation	93
Table 14 – Scenarios comparison considering P2P and mobility.....	95

Acronyms and Symbols

List of Acronyms

A4SG	Agent-based ecosystem for Smart Grids modelling
ACOM	Agent Community
AEM	Automated Energy Management
AI	Artificial Intelligence
AMI	Advanced Metering Infrastructure
ANN	Artificial Neural Networks
ARC	A4SG Representation Community
CEC	Citizen Energy Community
CECOS	Citizen Energy Communities Operator System
DBI	Davies-Bouldin Index
DDPG	Deep Deterministic Policy Gradient
DER	Distributed Energy Resources
DR	Demand Response
DRL	Deep Reinforcement Learning
DSO	Distribution System Operator
EM	Electricity Markets
EMS	Energy Management System
ESS	Energy Storage Systems
EV	Electric Vehicles
FAIR	Findable, Accessible, Interoperable and Reusable
FIPA	Foundation for Intelligent Physical Agents
FCT	Foundation for Science and Technology
GECAD	Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development
HTTP	Hypertext Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IT	Information Technology
MAE	Mean Absolute Error
MAS	Multi-Agent System
MIBEL	Iberian Electricity Market

ML	Machine Learning
MUC	Multi-User Chat
P2P	Peer-to-peer
PEAK	Python-based framework for hEterogeneous Agent Communities
PES	Power and Energy Systems
REC	Renewable Energy Community
RES	Renewable Energy Sources
RL	Reinforcement Learning
RMSE	Root Mean Squared Error
SDG	Sustainable Development Goals
SG	Smart Grid
SPADE	Smart Python Agent Development Environment
TD3	Twin Delayed Deep Deterministic Policy Gradient
TE	Transactive Energy
TSO	Transmission System Operator
XMPP	Extensible Messaging and Presence Protocol

List of Symbols

A	Action space
R	Reward space
S	State space
t	Current period

1 Introduction

This initial chapter provides a general contextualization of the work described in this dissertation. This chapter presents the contextualization of the work, the main research question and objectives, the scientific contributions, and the document organization.

1.1 Contextualization

Renewable energy generation, like wind and solar, has huge potential to reduce both greenhouse gas emissions and other negative environmental impacts from electricity generation (European Parliament, 2018). However, integrating these technologies into the electric power grid poses ongoing technological and institutional challenges (Rehmani et al., 2018). The large-scale integration of renewable power requires the modernization of the electrical power grid, both the high-voltage transmission network, needed to transport and integrate electricity generated from large, and variable renewable energy projects, and the low-voltage distribution network needed to integrate small-scale, decentralized renewable energy sources (RES) (Fang et al., 2012).

The emergence of the Smart Grid (SG) concept and its implementation stands out as the most viable solution for power systems modernization (Romero Agüero et al., 2017). While traditional power grids are generally used to carry power from a small number of central generators to a large number of customers, in contrast, the smart grid uses two-way flows of electricity and information to create an automated and distributed advanced energy delivery network (Fang et al., 2012). Smart grids are regarded as the new generation of electric power systems, combining the development of Information Technology (IT), distributed systems, and Artificial Intelligence (AI), to enable more features on real-time monitoring (Hossain et al., 2019).

A smarter grid can contribute to both climate change mitigation and adaptation, by increasing low-carbon electricity production and enhancing system reliability and flexibility (Ourahou et al., 2020). In that way, it can have a serious impact on political agreements relative to climate and sustainable goals. For instance, the 2030 Climate and Energy Package (European

Commission, 2014) is binding legislation in force from 2021 to 2030, setting three main targets: (i) a 40 % reduction in CO₂ emissions, compared to the levels verified in 1990; (ii) increase the use of renewable energy sources so that they represent 32 % of the production of energy in the European Union; and (iii) increased energy efficiency in 32.5 %. Besides that, other policies have been specified to motivate the reduction of energy misconduct and ensure that the defined targets are achieved, such as the European Green Deal (European Commission, 2019a) and the European Climate Law (European Commission, 2020). To reinforce the need to address environmental issues and sustainability, the United Nations proposes 17 Sustainable Development Goals (SDGs) to better target current research objectives. In the context of this work, the most relevant are: Affordable and Clean Energy, Sustainable Cities and Communities, Responsible Consumption and Production, and Climate Action.

Despite bringing many benefits, the smart grid approach reinforces the decentralization of Power and Energy Systems (PES), transforming the entire existing system into a more complex one, where the existence of a single centralized control unit is not effective (Ansari et al., 2016). Besides that, the increasing complexity of the smart grid requires dealing with big amounts of data collected in a distributed manner (Zhang et al., 2018). In order to support decision-making processes, data is continuously communicated among equipment (e.g., sensors and actuators), mainly based on Internet of Things (IoT) technology (Casado-Vara et al., 2019). In systems where the decision process is data-based, data is pre-processed, processed, and then information is extracted for decision-making. The decision is based on the objectives to be achieved by the system and its participating entities.

Innovative improvements in distributed AI techniques promoted the development of the Multi-Agent Systems (MAS) (Weiss, 1999), and its subsequent application to the modelling of the smart grid, given its distributed nature. MAS are distributed and concurrent by nature; their agents are autonomous entities in the system with their own sense of the environment and objectives, as they work to accomplish the best for themselves while acting strategically to meet the system's overall objective (Mahela et al., 2022). Although MAS are widely employed in the domain of smart grids, there does not appear to be a system capable of modelling all entities and their interactions, nor are there any tools of simple implementation and deployment capable of unifying different MAS to form a complete solution. Thus, there is a need for an agent-based ecosystem in which different agents and MAS can comprehend and interact with one another seamlessly, so facilitating the access to new information and services and, as a result, achieving both the agents' and the overall system's objectives.

The increasing amount of opportunities in the smart grid opens a new range of possibilities for energy customers to take advantage of their energy resources. With this, energy customers become more active in the search for services and aggregation entities that suit their profile, in order to create more advantages to their active participation in the smart grid (Baidya et al., 2021). The customers' interaction with these services and aggregation entities can be seen as static and dynamic. The static interactions represent the ones where the customers cannot move from, at least, in a seamless way. For instance, the customers' interactions with the distribution system operator (DSO) and transmission system operator (TSO) are considered

static. On the other hand, dynamic interactions allow customers and entities to navigate through services and aggregation entities by signing contracts with various entities over time. Energy aggregators and retailers are examples of dynamic interactions present in the smart grid. In the context of an agents' society, agents must be able to move between systems, which can represent aggregation entities, depending on their context and objectives. As such, systems that are part of the society must be prepared to receive agents who were previously part of other MAS. However, given the closed nature and heterogeneity of the agent-based systems, their agents become immobile. In addition, there is also a need to communicate with agents that do not belong to the society and integrate them as a part of it, as an agent should not be required to be on the same physical host as the other MAS' agents to enable the communication and knowledge sharing with them.

One of the primary topics of discussion when addressing MAS is the agents' intelligence and ability to learn from their own actions. Reinforcement Learning (RL), a trial-and-error learning method capable of utilizing agents' actions to interact with their environment, is one of the primary techniques used to integrate these characteristics in agents. However, RL typically deals with large datasets, and in the context of the smart grid, data is frequently missing during certain periods, making forecasting difficult and creating an uncertainty component in the application of this type of algorithms. In addition, MAS must typically be developed from scratch in order to incorporate RL into its agents, which is impractical for already deployed systems, especially when dealing with agents that represent physical devices and require continuous and real-time monitoring and control.

To address the identified issues, this dissertation proposes the novel Agent-based ecosystem for Smart Grids modelling (A4SG), capable of representing the entities engaged in the smart grid as well as their static and dynamic interactions. Using an agent-based approach, this scalable ecosystem provides its customers with a methodology that enables their dynamic participation and decision-making in the smart grid. This methodology integrates functionalities that allow agents to move between physical hosts and agent-based systems (i.e., physically and virtually) and to create new agents (i.e., branch agents) to represent the main agent in new objectives of the entity it represents, promoting a more accurate and effective management of their energy resources. Besides that, A4SG allows the coexistence of several MAS and agent communities, promoting agents' interaction between different systems. To test and validate the proposed ecosystem, multiple case studies were conceived and executed, considering different possible applications in the smart grid domain, and the results show an increase in the active participation efficiency of the energy customers, thus obtaining several benefits, for instance, at an economic level.

1.2 Research Question and Objectives

This dissertation presents the work developed to address multiple problems observed in the application of MAS to the modelling of the smart grid. To better guide the research within the

scope of this work, four primary research questions were formulated. This dissertation addresses these research questions throughout, and they are described as follows:

- RQ1 – What is the current state-of-the-art of MAS, particularly in what regards its application to smart grid modelling?
- RQ2 – Is an agent-based ecosystem capable of representing and modelling the smart grid in a unified solution?
- RQ3 – Can agent mobility and agent branching be a way for representing the dynamic interactions present in the smart grid in an organic flow?
- RQ4 – Does the integration of RL in an agent-based ecosystem benefit its application in real contexts?

The main objective of this dissertation is the conception, implementation, testing and validation of a robust and effective solution for modelling the smart grid and intelligent participation within it. In a way to facilitate the achievement of the main objective and to answer all the proposed research questions, multiple objectives for this work were appointed. They are described as follows:

- O1 – Investigate the current state-of-the-art of MAS in general and its application to the smart grid domain;
- O2 – Conceiving the agent concept capable of representing the energy customers and entities present in the smart grid;
- O3 – Proposing an agent-based ecosystem to model the smart grid and its entities;
- O4 – Proposing a physical mobility functionality capable of moving agents between hosts, in a request of the agents their selves (or the entities they represent) or of the ecosystem to balance the available host machines;
- O5 – Proposing a virtual mobility functionality to address the dynamic relations of the smart grid, allowing agents to move between services and aggregation entities considering personalized evaluation functions;
- O6 – Proposing a novel functionality, entitled branching, to allow customers and entities to have multiple representations simultaneously in the ecosystem, through multiple agents. This functionality should be able to decrease the overload of the agents in what regards objectives and data;
- O7 – Proposing functionalities to deal with the difficulties introduced by having multiple agents (i.e., branch agents) representing the same customer or entity, such as objectives interdependencies and conflicts;
- O8 – Conceiving an abstract service for RL seamless and intuitive integration in agent-based systems in a smart grid context;
- O9 – Testing the proposed mobility and branching functionalities in a smart grid context, verifying its advantages to energy customers;

- O10 – Testing the proposed agent-based ecosystem and all its services and functionalities as a unified solution.

As all the outlined objectives have been met, this dissertation serves as the final report of the developed work. To facilitate the understanding of which objectives contribute to each research question, Table 1 presents the match between the proposed research questions and the objectives that are useful to answer them.

Table 1 – Research Questions and Corresponding Objectives

Research Questions	Corresponding Objectives
RQ1	O1
RQ2	O2; O3; O4; O5; O6; O7; O8; O10
RQ3	O4; O5; O6; O7; O9; O10
RQ4	O8; O10

1.3 Scientific Contributions

This work was developed in the Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development (GECAD)¹, which have as mission the promotion and development of scientific research in the knowledge and decision sciences domains, having Information Technologies as support. It involves 2 main areas: intelligent systems and power energy systems.

The work developed in this dissertation was supported by two R&D projects funded by the Portuguese Foundation for Science and Technology (FCT):

- MAS-Society² - Multi-Agent Systems SemantiC Interoperability for simulation and decision support in complex energy systems (PTDC/EEI-EEE/28954/2017).
- PRECISE³ - Power and Energy Cyber-Physical Solutions with Explainable Semantic Learning (PTDC/EEI-EEE/6277/2020)

The MAS-Society project focuses on providing effective agent-based solutions to enable the widespread deployment of distributed energy resources (DER), such as renewable-based generation, demand response (DR), energy storage systems (ESS), and electric vehicles (EV), thereby allowing them to realize their full potential for enhancing overall energy efficiency, economic and energetic sustainability. The systems designed to solve particular problems should be able to interoperate with one another by sharing information and services, thereby enabling the concept of a Multi-Agent Systems Society. On the other hand, the primary

¹ GECAD - <http://www.gecad.isep.ipp.pt/>

² MAS-Society - <http://www.gecad.isep.ipp.pt/MAS-Society/>

³ PRECISE - <https://www.gecad.isep.ipp.pt/precise/>

objective of PRECISE is to provide effective automated solutions to enable efficient, intelligent, and real-time consumer-side energy management. PRECISE models enable the widespread use of local Automated Energy Management (AEM), empowering consumers to reduce their energy bills while ensuring their preferences and needs are met.

Besides the direct contribution to these two scientific projects, throughout the development of this work, a total of eight scientific papers were published (from those, four in scientific journals and four in scientific conferences):

- [Conference] Gabriel Santos, Ricardo Faia, **Helder Pereira**, Tiago Pinto, Zita Vale “Blockchain-based Local Electricity Market Solution”, presented in “18th European Energy Market Conference” (EEM22), doi: 10.1109/EEM54602.2022.9921035
- [Journal] **Helder Pereira**, Luis Gomes, Zita Vale, (2022) “Optimizing participation in local energy market through deep reinforcement learning in an agent-based ecosystem”, accepted for publication in Energy Reports
- [Conference] Bruno Ribeiro, **Helder Pereira**, Luis Gomes, Zita Vale “Python-based Ecosystem for Agent Communities simulation”, presented in “17th International Conference on Soft Computing Models in Industrial and Environmental Applications” (SOCO 2022), doi: 10.1007/978-3-031-18050-7_7
- [Journal] **Helder Pereira**, Luis Gomes, Zita Vale, (2022) “Peer-to-Peer Energy Trading Optimization in Energy Communities using Multi-Agent Deep Reinforcement Learning” accepted for publication in Energy Informatics
- [Conference] **Helder Pereira**, Ricardo Faia, Luis Gomes, Pedro Faria, Zita Vale (2022) “Incentive-based and Price-based Demand Response to Prevent Congestion in Energy Communities”, presented in 2022 IEEE International Conference on Environment and Electrical Engineering (EEEIC), doi: 10.1109/EEEIC/ICPSEUROPE54979.2022.9854648
- [Conference] **Helder Pereira**, Bruno Ribeiro, Luis Gomes, Zita Vale (2022) “CECOS: a Centralized Management Platform Supported by Distributed Services to Represent and Manage Resources Aggregation Entities and its End-users in a Smart Grid Context”, presented in “11th Symposium on Control of Power and Energy Systems” (CPES 2022), doi: 10.1016/J.IFACOL.2022.07.054
- [Journal] **Helder Pereira**, Luis Gomes, Pedro Faria, Zita Vale, Carlos Coelho (2021) “Web-based platform for the management of citizen energy communities and their members”, published in Energy Informatics in 2021, doi: 10.1186/S42162-021-00155-7
- [Journal] Bruno Mota, Miguel Albergaria, **Helder Pereira**, José Silva, Luis Gomes, Zita Vale, Carlos Ramos (2021) “Climatization and luminosity optimization of buildings using genetic algorithm, random forest, and regression models”, published in Energy Informatics in 2021, doi: 10.1186/s42162-021-00151-x

In addition to the scientific papers already accepted and published, two other papers are submitted to journals, currently under review:

- [Journal – Under review] **Helder Pereira**, Luis Gomes, Zita Vale “Agent-based Ecosystem with Branching and Mobility Mechanisms to Support Agent Communities: a Smart Grid Application”
- [Journal – Under review] **Helder Pereira**, Bruno Ribeiro, Luis Gomes, Zita Vale “Smart grid ecosystem modelling using a novel framework for heterogenous agent communities”

In addition to the publication of scientific papers, the author of this dissertation presented part of the papers at international conferences:

- Energy Informatics.Academy Conference 2022 (EI.A 2022) – Vejle, Denmark (presented remotely)
- 9th International Conference on Energy and Environment Research (ICEER 2022) – Porto, Portugal
- 22nd International Conference on Environmental and Electrical Engineering (EEEIC 2022) – Prague, Czech Republic (presented remotely)
- 11th Symposium on Control of Power and Energy Systems (CPES 2022) – Fully remote
- 1st Energy Informatics.Academy Conference Asia (EI.A Asia 2021) – Beijing, China (presented remotely)

The European Union encourages research data to be findable, accessible, interoperable and reusable (FAIR), so in this dissertation, all the used datasets follow these guidelines, and besides that, a dataset was built and published, being accessible to the scientific community:

- Helder Pereira, Luis Gomes, Hugo Morais, Zita Vale (2022) “Dataset of 30 energy customers with flexibility data, and distributed generation, considering residential, small commerce, large commerce, and industrial customers”, published in Zenodo, doi: 10.5281/ZENODO.6783289.

As a result of the published scientific papers, two awards were won at international conferences:

- Best Paper Award in Energy Informatics.Academy Asia 2021 (EI.A Asia 2021) with the paper “Helder Pereira, Luis Gomes, Pedro Faria, Zita Vale, Carlos Coelho (2021) “Web-based platform for the management of citizen energy communities and their members”
- Best Presentation Award in 9th International Conference on Energy and Environment Research (ICEER 2022) with the paper “Helder Pereira, Luis Gomes, Zita Vale, (2022) “Optimizing participation in local energy market through deep reinforcement learning in an agent-based ecosystem”.

1.4 Document Organization

This dissertation is divided into seven chapters, each of which has been prepared and organized to facilitate the reading of the thesis as a whole or a part of a chapter.

The first chapter is an introduction and describes the contextualization for the development of the work presented in this dissertation. Additionally, the main research questions, objectives, and scientific contributions are outlined.

In the second chapter, the state of the art of the domains covered by this work is discussed, namely: the smart grids and the customers' active participation, agents and multi-agent systems, and finally, machine learning.

The third chapter describes the methods, models, materials, and tools used in the development of the work presented in this dissertation. Besides that, it contextualizes this work towards data privacy and protection, agents and data security, and possible ethical and social issues associated with it.

The ecosystem developed in the scope of this work, entitled A4SG, is presented in the fourth chapter. Firstly, the ecosystem's architecture is explained and then the types of agents are presented. The services and functionalities to support and integrate intelligence into the ecosystem and its agents are also described.

The fifth chapter describes the smart grid services developed or integrated into the ecosystem as service-oriented agents, as well as how reinforcement learning is used to integrate intelligent service participation in the agents of the ecosystem.

In the sixth chapter, the developed case studies to test and analyze the ecosystem when applied to real data and real contexts are described.

Finally, in the last chapter are presented the final considerations and conclusions of the work and possible future developments.

2 State of the Art

This chapter addresses the state of the art of the most relevant topics used in the project: smart grids, multi-agent systems, and machine learning. In addition to exploring the works proposed in these domains individually, are also analyzed the applications of multi-agent systems and machine learning on the smart grid. Besides that, the gaps in these domains' literature are identified in the chapter's conclusions.

2.1 Smart Grids and Energy Customers' Participation

The traditional electricity grid is a product of rapid urbanization and infrastructure developments in various parts of the world in the past century (Farhangi, 2010). With the advent of the digital economy and information era, the electric power industry is confronted with new challenges. Some problems relevant to climate change, environmental protection and sustainable development became more and more serious (Sachs et al., 2019). The European Union has committed much of its efforts in recent years changing the organization of the energy market and establishing new targets for deregulation and combating climate change (Tagliapietra et al., 2019).

More recently, another aspect increased the difficulty of the management of the energy and power system as it was conceived. The introduction of distributed generation systems allowed consumers to inject energy into the grid, interfering with the normal process of demand and supply balance (Kakran and Chanana, 2018). Meanwhile, customers' demand for higher supply reliability, excellent power quality and satisfactory services has emerged. Although traditional electric power technologies can alleviate these problems, they have proved to be inappropriate (Zame et al., 2018). As a result, new technology should be used to help with the resolution of these challenges. To allow pervasive control and monitoring, the smart grid is emerging as a convergence of information and communication technologies with power system engineering (Farhangi, 2010). Table 2 (Farhangi, 2010) presents a comparison between the traditional grid and the smart grid, considering important topics of innovation.

Table 2 – Comparison between the traditional power system and the smart grid. Adapted from (Farhangi, 2010)

Comparison Feature	Traditional Grid	Smart Grid
Main usage	Electromechanical	Digital
Communication	One-way communication	Two-way communication
Generation	Centralized generation	Distributed generation
Network's data access	Blind	Self-monitoring
Network's recovery	Manual restoration	Self-healing
Behaviour with anomalies	Failures and blackouts	Adaptive and islanding
Control type	Limited control	Pervasive

The information and communication technology layer of the smart grid brings additional problems related to the security and privacy of all the parties involved in the energy system (Kumar et al., 2019). The potential consequences of successful cyber-attacks on the electricity grid are quite harmful to consumers and to the entities that are part of the smart grid. In this way, the information and communication layer must ensure a secure communication channel, where the main security objectives are confidentiality, integrity, and authorization of the exchanged data (Gunduz and Das, 2020).

One of the advantages of the smart grid is the capability of seamlessly integrating microgrids. Small autonomous grids have existed for many decades in remote communities where the interconnection with the main power grid is not feasible due to technical and/or economic reasons (Olivares et al., 2014). The decentralization of power systems potentiates the rising adoption of microgrids. There are multiple definitions for microgrids across the literature, and consequently, there is not one unanimously accepted. However, there are definitions broadly used and consensual. One of the main ones is proposed by the CIGRÉ Working Group C6.22 Microgrids Evolution Roadmap: *“Microgrids are electricity distribution systems containing loads and distributed energy resources, (such as distributed generators, storage devices, or controllable loads) that can be operated in a controlled, coordinated way either while connected to the main power network or while is landed”* (Marnay et al., 2015). The microgrids have two possible classifications, considering if they are connected to the main grid or not: grid-connected or islanded.

Smart grid adoption has a significant impact on larger energy customers, such as distribution and transmission entities, but also on the small and medium customers, that have the opportunity to participate in the smart grid in multiple ways, considering their context (Steriotis et al., 2018). In order to successfully build and implement a smart grid, it is necessary to set key policy objectives to ensure consumer acceptance and to make a scientific analysis of the factors affecting consumer acceptance, since the ultimate deployment of the smart grid depends on the customers' acceptance of smart grid products and services (Park et al., 2014). Consumers must change their behaviour to take full advantage of smart grids. For instance, well-informed and price-responsive consumers are expected to play an active role in managing electricity

consumption rather than being passive customers only (Rathor and Saxena, 2020). In that way, consumers can contribute to energy saving, energy efficiency and peak load shifts through the use of real-time electricity information and smart meters (Avancini et al., 2019).

Therefore, the main advantages of implementing a smart grid include: (i) reliability and security of energy distribution, (ii) shift of the peak load, (iii) enhanced efficiency, (iv) enable high shares of renewables in power system, (v) decreased greenhouse gas intensity of power system, and (vi) active participation of consumers (Moretti et al., 2017).

2.1.1 Transactive Energy

The growing complexity of electrical energy systems has changed the paradigm of how they are managed, involving more consumer participation, and thus creating new opportunities for them. Distributed generation and prosumers (i.e., consumers with energy generation capacity) accentuate this change, as they can actively participate in the growing energy economy. The growing number of prosumers implies the creation and the establishment of an electricity trading mechanism that allows customers to trade electricity with each other (Zia et al., 2020). In this regard, the transactive energy (TE) concept has been addressed.

Transactive energy is defined by the Gridwise Architecture Council as *“techniques for managing the generation, consumption, flow of electricity within the power system that allow the dynamic balance between demand and supply keeping in view the constraints of the entire network”* (Gridwise Architecture Council, 2015). In contrast to the typical hierarchical grid topology, it promotes a network environment for decentralized energy nodes. The network structure facilitates interaction, allowing all levels of energy generation and consumption to connect with one another (Huang et al., 2021). Under the concept of TE, distributed flexible resources are directly controlled by their owners (Chen and Liu, 2017). Transaction mechanisms are designed to align individual behaviours with the system’s interests.

TE is a broad term that encompasses a variety of customer active participation techniques and methodologies, such as local markets and DR, which can be applied at any level of the smart grid, from transmission to distribution (Guerrero et al., 2020). The current principles of wholesale transactive power systems are expanded into retail markets with customers equipped with sophisticated Energy Management Systems (EMS), allowing small electricity customers to participate actively in the electricity markets (Rahimi and Albuyeh, 2016). These systems can also enable peer-to-peer (P2P) management in smart grids through the use of intelligent devices, each of which has its own choice and aim, taking into account the owners, constraints, requirements, and context (Abrishambaf et al., 2019).

The potential benefits of efficient integration of behind-the-meter DER are substantial (Li et al., 2018). Prosumers are empowered by the inherent flexibility of DER as they can manage their electricity usage and production capabilities in response to price and dispatch signals. In this way, DER owners may reduce their electricity costs by managing their demand, reducing their

reliance on the grid, maximising the value of their DER, providing backup supply, or arbitrage their retail tariff (Burger and Luke, 2017).

At the same time, distributed resources used for TE are intermittent (e.g., in the case of renewables) and nonuniformly deployed, posing significant new challenges to be faced in the resources' management (Siano et al., 2019). These issues can be addressed using both centralized and decentralized approaches, with the strategy adopted depending on the context and the system's objectives (Abrishambaf et al., 2019).

2.1.2 Demand Response

Peak energy demand hours are a major challenge to the correct operation of the electric energy grid, as the continuous demand growth raises the possibility of power outages as well as the raise the marginal cost of supply (Uddin et al., 2018). Power failures, or blackouts, can cause inconvenience to energy customers, and also financial losses (Wu et al., 2017). Therefore, recognizing and regulating peak hours allows customers to better manage their energy consumption while also allowing for real-time response strategies (Alduailij et al., 2021).

The Demand Response (DR) concept is one of the most addressed topics in the literature when it comes to reducing the demand from a group of consumers, and the European Parliament defines it as "the change of electricity load by final customers from their normal or current consumption patterns in response to market signals, including in response to time-variable electricity prices or incentive payments, or in response to the acceptance of the final customer's bid to dell demand reduction or increase at a price in an organized market" (European Parliament, 2019) in the Directive 2019/244. DR programs offer efficient solutions for many power system problems, such as high generation cost, high demand's peak to average ratio, high emissions, reliability issues and congestion in generation, transmission and distribution systems (Jordehi, 2019).

DR programs can be applied with multiple strategies, and they are classified into two types: incentive-based and price-based programs. The incentive-based programs aim to benefit or penalize consumers, depending on whether or not they comply with requests from the network (Imani et al., 2018). On the other hand, price-based programs aim at changing the price of energy tariffs to motivate consumers to reduce their demand during the peaks of electricity demand (Yan et al., 2018). In this case, consumers must have dynamic tariffs, which are not regulated by fixed-price contracts, where energy is always paid at the same price. The multiple strategies and approaches for the application of DR programs are represented in Figure 1 (Jordehi, 2019).

Despite the many advantages for consumers in the participation in DR programs, for them to be prepared to participate in these programs in a real context, there are some associated costs and disadvantages. The initial costs/disadvantages are related to the investment in technology that enables the participation and establishing a response plan or strategy, while the inconvenience related to reduced comfort or to the rescheduling of the energy consumption,

and maintenance costs, occur during the participation in the event (Bradley et al., 2013). Another possible situation, this one related to the network itself, is the appearance of

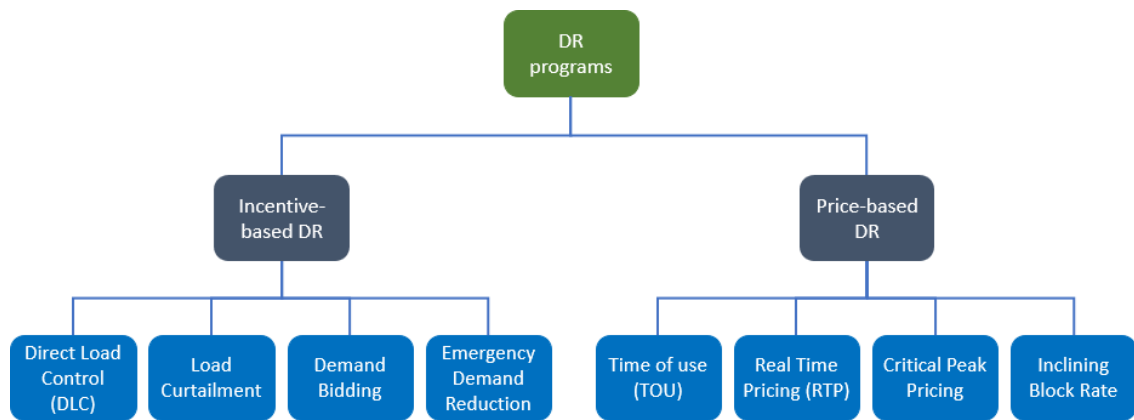


Figure 1 – Demand Response programs classification. Adapted from (Jordehi, 2019)

rebound peaks. These peaks tend to arise from the mass execution of a DR program, which may cause the majority of the participating consumers to move their consumption to the same hours of lower demand, thus being able to transform them into peak hours due to the excessive number of consumers (Ghorashi et al., 2020).

The real benefits of DR programs can only be achieved if they are implemented on a large scale, even because a minimum level of demand cut is required in most DR programs (Ribó-Pérez et al., 2021). As a rule, this minimum reflects a high value to be achieved by a single consumer, and consequently, direct participation in the programs is impossible. The participation of small and medium customers can then be performed by aggregating the cuts performed by them (Faria et al., 2018).

2.1.3 Aggregators

The vast majority of energy consumers are unaware of how energy is used most of the time, and they lack the ability to directly engage in energy markets. Therefore, an intermediary entity is required in this type of procedure, as it can motivate customers to participate actively and facilitate their engagement with energy markets (Kowalska-Pyzalska, 2018).

An energy aggregator is a type of service provider that can increase or reduce the electricity consumption of a group of consumers according to the total electricity demand on the grid (The European Consumer Organisation, 2018). When the customers produce energy in excess, an aggregator can also manage directly or indirectly groups of consumers and prosumers in order to sell pools of loads and generators as single items in the electricity markets, (Ponds et al., 2018). Customers require the services of an aggregator to help them to negotiate through the market complexities since they do not have the know-how and the technical resources to do it on their own. The aggregator provides backup to individual customers, reducing their risk and improving the reliability of their DR participation (Lu et al., 2020).

Energy aggregators can specialize in aggregating different sorts of customers, which are typically split into three categories: residential, commercial/large businesses, and small industries. This focus on a single sort of customer is attributable to several factors. The main one is that, to control various types of loads, different monitoring technologies are required. A washing machine, for instance, may be run manually, whereas heavier loads necessitate more complex control and communication devices (Zhang et al., 2018). Another reason for this specialization could be that different components of legal terms for bundling and selling flexibility may vary based on the type of consumer (Inshakov et al., 2019).

The aggregator operates as an intermediary between the customers and the energy markets, negotiating contracts with several consumers to aggregate their capacity, to reduce or raise energy consumption or perform load shifting during specific periods. Thus, the aggregator can create a collection of consumers' equipment that can be controlled (e.g., heating systems and washing machines) and that has characteristics that allow for flexibility in periods of need. As can be seen in Figure 2, the flexibility acquired by the aggregator through the optimization of consumption by contracted consumers can then be sold on the energy markets to the best offer (Ottesen et al., 2018).

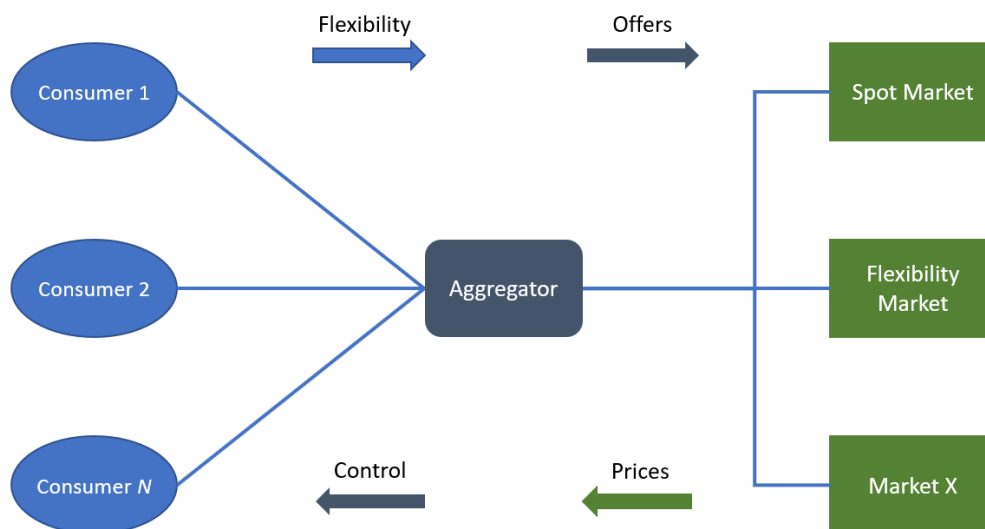


Figure 2 - Relationships established by the aggregator. Adapted from (Ottesen et al., 2018)

When energy aggregators are linked to the final consumer market, they enable the establishment of automatic and tailored controls for various loads or installations that allow remote access, while taking into consideration consumer preferences and behavioural patterns (Iria and Soares, 2019a). Grid operators can also benefit from aggregators in terms of system maintenance expenses, as the need for grid maintenance and expansion is decreased, allowing for even better integration of renewable energy generating technologies (Kerscher and Arboleya, 2022).

The role of an aggregator requires very valuable and specific skills. An aggregator requires, for instance, an in-depth understanding of industry operations as well as experience in identifying each industry's flexibility opportunities and limitations. Furthermore, an aggregator must have

the technical capabilities to connect consumers and incorporate their loads into their list of monitored loads (Poplavskaya and de Vries, 2020).

2.1.4 Energy Communities

The European Commission's Clean Energy Package breaks new ground for consumers by recognizing citizens' and communities' rights to participate directly in the energy industry for the first time under EU law (European Commission and Directorate-General for Energy, 2019). It formally acknowledges and sets out legal frameworks for certain categories of community energy as “energy communities”.

Energy communities are specified in two separate directives of the European Clean Energy Package. The revised Renewable Energy Directive (EU) 2018/2001 sets the framework for ‘renewable energy communities’ (REC) covering renewable energy (European Parliament, 2018). The revised Internal Electricity Market Directive (EU) 2019/944 introduces new roles and responsibilities for “citizen energy communities” (CEC) in the power system, covering all types of electricity (European Parliament, 2019). Energy communities are described in the guidelines as a possible way to organize collective citizen actions in the power system (Frieden et al., 2019). According to the Electricity Market Directive, “the provisions on citizen energy communities do not preclude the existence of other citizen initiatives such as those stemming from private law agreements”. Both directions enable various energy community organizational forms (e.g., association or cooperative) to be established through a legal entity.

Energy communities are incorporated as a sort of non-commercial market actor that combines non-commercial economic goals with environmental and social community goals (Roberts et al., 2019). According to the amended Electricity Market Directive, CECs are a new form of business due to their membership structure, governance criteria, and function (European Parliament, 2019). The revised Renewable Energy Directive makes special reference to the characteristics of local RECs in terms of size and ownership structure (European Parliament, 2018). Furthermore, both citizen energy communities and renewable energy communities can engage in similar activities such as energy generation, distribution, supply, aggregation, consumption, sharing, storage, and the provision of energy-related services (Caramizaru and Uihlein, 2020).

An energy community, as articulated within the notion of sharing economies, is a type of community-driven institution that takes social control of shared energy resources through decentralisation. Individual consumers, producers, and prosumers in an enclosed topology can establish such space in order to build independent projects and actively contribute to a more sustainable paradigm (Moncecchi, 2021).

As governments commit to national sustainability goals (e.g., meeting a certain percentage of the country's energy demand using renewable energy resources), energy communities can form their own goals and plans to achieve them, driving change locally through resources aggregation. (Caramizaru and Uihlein, 2020). Consequently, customers within a community can experience the benefits of the energy transition beyond the economic perspective, while they willingly

acquire several responsibilities as well, the more active their participation is in the community's initiatives (Koirala et al., 2018).

In contrast to microgrids, which aim to aggregate power, energy communities arise as legal entities that seek the aggregation of energy resources in order to lower power prices and power peaks (Radl et al., 2020). As a result, the primary goals of energy communities include: (i) lowering total electricity prices, as well as non-monetary goals such as (ii) ensuring clean energy supply (Gui and MacGill, 2018), (iii) enhancing the understanding and acceptance of renewable energies; or (iv) concentrating the reduction of greenhouse gas emissions (Schram et al., 2019).

In Europe, there are already several energy communities deployed. In (Heaslip et al., 2016) the results of preliminary fieldwork investigations of sustainable energy community development approaches in two Danish islands and one rural village in Ireland are reported. The authors' major evidence suggests that social barriers are interrelated and frequently reinforce each other, posing the most significant challenge to the deployment of energy communities. In (Giordano et al., 2019) it is proposed the development of a smart energy community, which would bring savings of 66 % relative to energy costs, as the community would have a self-consumption of nearly 95 %, meaning less power would have to be transported on the network.

2.2 Agents and Multi-Agent Systems

The software agent definition is not consensual in the literature, especially as it may change depending on its goals and area of application. A commonly accepted definition is that an agent is a computer system in a particular environment, with which it interacts through sensors and actuators, trying to accomplish its design objectives, thus acting reactively and proactively (Wooldridge and Jennings, 1995). Nevertheless, some of the agent characteristics are consensual in numerous definitions. Weiss proposed the characteristics that an agent must comply with to meet their goals (Weiss, 1999):

- Agility: analysing and identifying new opportunities;
- Autonomy: ability to control their actions;
- Character: present a certain personality;
- Flexibility: if your tasks do not need to be predetermined;
- Intelligence: linking the characteristics of learning and autonomy.
- Learning: gaining experience with the result of actions and adapting decisions futures;
- Mobility: ability to move among machines based on the perception of the environment;
- Persistence: how much an agent exists over long periods;
- Pro-Activity: the ability to aim for a goal;
- Reactivity: ability to react and adapt to changes in the environment.

- Sensory Capacity: sensitivity to the perception of the environment in which it operates;
- Social Ability: ability to communicate with other agents in the system.

Single-agent systems are very useful as stand-alone entities performing tasks delegated by a customer, freeing him from hard work (Neto, 2005). On the other hand, MAS are systems composed of multiple interacting computing elements, i.e., agents. Most often, agents coexist in environments containing other agents and interact with each other, thus composing a MAS. MAS are considered dynamic since the environment may change with an agent's interaction. Reliability, modularity, scalability, adaptability, concurrency, parallelism, and dynamism are some of the advantages MAS present over single-agent systems (Dorri et al., 2018). The agents, in the context of MAS, must have two important abilities. First, they are at least to some extent capable of autonomous action – of deciding for themselves in order to achieve their goals, and second, they are capable of interacting with other agents – not simply exchanging data but engaging in social activity (Wooldridge, 2009). A characteristic of MAS, particularly important when applied to power systems, is its robustness. This type of system must carry its normal execution, even when errors are detected in one or more agents of the system since these agents can be replaced by new agents that represent the customer or entity whose agent saw its execution interrupted.

The general concept of a MAS is related to the creation of multiple autonomous agents that interact with each other to satisfy a common goal, although in the system each agent can have individual goals that it will try to achieve (Balaji and Srinivasan, 2010). This type of system can have two aspects in its execution. The system can make use of agents so that they cooperate to try to achieve their objectives, and consequently solve the overall objective of the system. On the other hand, the system can use agents in a competitive mode, where they compete with each other to produce individual goals, being that in this mode it is more likely that some agents fail in the conclusion of their objectives (Balaji and Srinivasan, 2010).

A commonly associated problem with MAS and agents is the data overloading and consequent performance capability. As such, a topic explored for resource optimization is agent cloning, a process in which an agent clones itself. In (Shehory et al., 1998), a model of agent cloning is proposed to solve the issue of local agents overload. Agent cloning encompasses task delegation and agent mobility. According to Shehory et al., agents can clone, delegate work to others, die, or merge (Shehory et al., 1998). As a result, MAS or agents that are overloaded with tasks can delegate responsibilities to other agents, produce new agents to undertake excess jobs and exploit unused resources, or move to other hosts.

MAS and agent-based models can be applied to a wide variety of domains due to their ability to adapt to different contexts. For instance, they can be applied to production scheduling (Shukla et al., 2021), transportation (Monteiro et al., 2014), and fraud detection (Brito et al., 2018). In what regards the application of MAS to smart grids, this technology can be useful to address multiple issues, namely: (i) control, operation and management, (ii) protection, (ii) monitor and diagnosis, (iv) transactive energy management, and (v) security (Hasanuzzaman

Shawon et al., 2019) (Xie and Liu, 2017). In Table 3 are described these main identified smart grid applications with the addressed challenges by each area of application.

Table 3 – Applications of multi-agent systems to smart grids

Smart grid application	MAS Properties	Main addressed challenges
Control; Operation; Management	The agents, normally, represent actuators or have actuators integrated, in order to enable direct actions on the energy network	Load frequency control; Microgrid islanding; Reactive Power Control
Protection	Integration of protection standards (defined by the International Electrotechnical Commission) to ensure that the grid maintains its operational status	Auto-configurable protection schemes; Self-healing strategies
Monitoring; Diagnosis	The agents are mainly used to be connected to sensors, receiving and processing their data, but do not actuate in the environment	Monitoring Systems; Fault-detection
Transactive energy management	Coordination and competitiveness between agents in transactive energy programs. The agents can include ML or optimization models to enhance participation	Demand Response; Peer-to-peer markets simulation; Demand-side management
Security	Cyber-Physical network technical (cyber threats) and non-technical (regulatory policies) security	Cyber-attacks detections; Network vulnerabilities evaluation

MAS have been used in power and energy systems to overcome problems that prove to be a challenge for traditional computing techniques. Within the field of electrical energy, MAS have been demonstrating their usefulness in the representation of entities and their customers, such as microgrids (Gomes et al., 2020a), energy communities (Reis et al., 2020), and their management models (Azeroual et al., 2020). In addition to representing entities belonging to the smart grid, MAS can also be used to individually represent customers and manage their resources, such as energy flexibility, and manage and support their active participation in the smart grid (Coelho et al., 2017). For instance, a topic commonly approached with MAS is participation in P2P energy markets and its optimization through intelligent algorithms and models. In (Gomes et al., 2020b) it is proposed a MAS able to run on single-board computers, to enable P2P trading in a microgrid, with a focus on the customer side, and in (Ye et al., 2021) it is proposed a multi-agent deep reinforcement learning approach capable of improving the scalability of the market. Besides that, MAS can be also used to ensure stability in the grid, by enabling fault diagnosis and grid reconfiguration (Rahman et al., 2018), voltage control (Wang et al., 2020), and voltage regulation (X. Wang et al., 2018).

One of the main applications of MAS in the context of the smart grid is the simulation of aggregators and participation in DR. For instance, in (Woltmann and Kittel, 2022) the authors propose a MAS to simulate DR participation in an industrial context, and in it (Oliveira et al., 2012) is proposed a smart grid simulation platform capable of representing virtual power players and curtailment service providers. In (Elshaafi et al., 2018) it is proposed an agent-based

decentralized approach to automated DR, where the methodology follows a hierarchical agent architecture able to represent different stakeholders to make their own decision processes. Another approach is to combine MAS and reinforcement learning, as proposed in (Aladdin et al., 2020), to optimize the participation in DR. With a different approach, not using MAS, but using a multi-agent-based algorithm to simulate DR, in (Rahmani et al., 2018) a methodology capable of minimizing costs and integrating a green data centre is proposed.

2.2.1 Organizational Paradigms

Associated with the characteristics of each agent and the system itself, various topologies and organizational paradigms can be applied, considering the relationships between the agents and the system's main goals. An agent organization can be defined as a social entity composed of a specific number of agents that accomplish several distinct tasks or functions and that are structured following some specific topology and communication interrelationships in order to achieve the main aim of the organization (Abbas et al., 2015). Thus, agent organizations assume the existence of global common goals, outside the objectives of any individual agent, and they exist independently of agents.

Given the wide range of MAS developed so far, many organizational paradigms have already been extensively explored, with the greatest examples being the following: hierarchies, holarchies, coalitions, teams, congregations, societies, federations, markets, matrix, and compound organizations (Horling and Lesser, 2004). It is generally agreed that there is no single type of organization that is suitable for all situations, and in some cases, no single organizational style is appropriate for a particular situation, and several different, concurrently operating organizational structures are needed (Oh et al., 2015). Table 4 (Horling and Lesser, 2004) presents a summary of MAS organizational paradigms.

Table 4 – MAS organizational paradigms. Adapted from (Horling and Lesser, 2004)

Paradigm	Key Characteristic	Benefits	Drawbacks
Hierarchy	Decomposition	Maps to many common domains; Scalable	Potentially brittle; Can lead to bottlenecks
Holarchy	Decomposition with autonomy	Exploit the autonomy of functional units	Must organize holons; Low performance
Coalition	Dynamic, goal-directed	Exploit strength in numbers	Short-term gains may not balance long-term drawbacks
Team	Group level cohesion	Address larger-grained problems; Task-centric	Increased communication
Congregation	Long-lived, utility-directed	Facilitates agent discovery	Sets may be overly restrictive
Society	Open system	Public services; Well defined conventions	Potentially complex; Agents may require additional social capabilities

Paradigm	Key Characteristic	Benefits	Drawbacks
Federation	Middle agents	Matchmaking, brokering, translation; Facilitates dynamic agent pool	Intermediaries become bottlenecks
Market	Competition through pricing	Good at allocation; Increased fairness	Potential for collusion; Allocation decision complexity can be high
Matrix	Multiple managers	Resource sharing; Multiply influenced agents	Potential for conflicts; Need for increased agent sophistication
Compound	Concurrent organization	Exploit the benefits of several organizational styles	Increased sophistication; Drawbacks of several organizational styles

Additionally, there is much interest in the MAS approach as a means of designing self-organized systems. Agents must be able to adapt to the most appropriate organisations in open settings due to the environment's unpredictable changes. In this way, self-organization is defined as a mechanism or a process which enables a system to change its organization without explicit command during its execution time (Ye et al., 2017).

2.2.2 Agents' Mobility

Mobility is one of the agent's characteristics proposed in (Weiss, 1999), and is defined by the ability of an agent to move its execution to another machine/host. However, in general, the agents that are part of a MAS are not mobile, or, when they are, mobility is just a mechanism that changes the agent to another host, regardless of the environment or context that surrounds them. Figure 3 represents an example of physical mobility between machines.

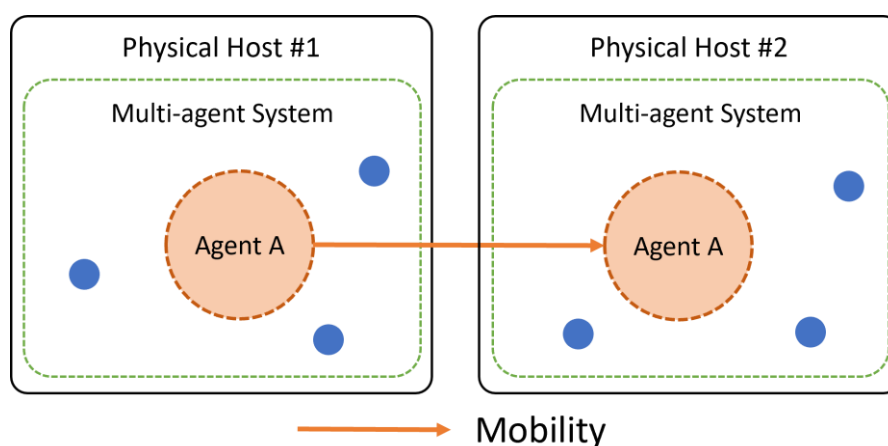


Figure 3 – Agents' physical mobility

Despite this, there are several examples where MAS are implemented with mobile agents. In (Cucurull et al., 2007) is proposed an architecture that enables agent mobility, based on IEEE-

Foundation for Intelligent Physical Agents (FIPA) standards. This architecture divides mobility into multiple steps, that can be adapted to the preference of the developers and/or agents, in order to create a flexible process. In the context of power systems, in (Akyol et al., 2012) is proposed VOLTTRON, an agent execution platform that supports agent mobility. Authorized hosts in VOLTTRON can dispatch and deploy agents using the mobility feature, making management and deployment easier and more flexible.

Mobility is also a critical consideration when discussing agent-based services. This is because the majority of services require the subscriber to provide significant volumes of data to the service's provider, such as historical data on energy consumption and generation. This makes participation in services extremely costly on a computational level for all the entities involved. Despite not being a solution normally used in MAS to solve these difficulties, the code-shipping paradigm can be highly advantageous when compared to data-shipping (Braun, 2003). With this methodology, the service's provider agent moves to the host where the service's subscriber is hosted so that there is not a huge amount of data transported between machines, but only the code of one agent. This type of paradigm is frequently employed in mobile edge computing (S. Wang et al., 2018).

In the smart grid environment, when an agent signs up for a particular service or MAS, it is because, in its current context, it would be the most suitable one among the available to take greater advantage of its energy resources. However, the properties and objectives of the agents and the MAS themselves are not static. By changing these parameters, it may make sense to move the agent to another MAS, that best fits the agent's current context. For instance, relatively to power systems, an aggregator can change its target hours for the DR programs, or an energy retailer can change its tariffs, causing their target clients to change.

Smart grids increasingly provide opportunities for their participants, and as such, they should be able to participate in the various initiatives that they find convenient in their context, something that, in agent-based systems can be achieved through mobility, which allows agents to move among MAS, in order to achieve their goals. As the agents may want to participate in multiple MAS, or even subscribe to different services, simultaneously, there is the need to have a functionality that enables this ability.

2.2.3 Agents' Intelligence and Learning

Although intelligence is normally a characteristic associated with agents, having intelligent behaviour is not something intrinsic to a software agent, if it is not prepared for it. According to Russell and Norving, an intelligent agent must perceive its environment, act autonomously to achieve its goals, and should be able to improve its performance with learning methods or with the use of knowledge (Russell and Norvig, 2002).

For an agent to be able to learn from its actions, it must be evaluated for each action it takes, taking into account the state of the environment that surrounds it, before and after the performed action. In these cases, Reinforcement Learning (RL) - where an agent learns by

interacting with its dynamic environment (Kaelbling et al., 1996), is an adequate approach. In RL, at each time step, the agent perceives the state of the environment and takes an action, which causes the environment to transit into a new state. Figure 4 presents the basic agent-environment interaction. A scalar reward signal evaluates the quality of each transition, and the agent must maximize the cumulative reward along the course of interaction. The RL feedback (the reward) is less informative than in supervised learning, where the agent would be given the correct actions to take (Cherkassy and Mulier, 1998). The RL feedback is, however, more informative than in unsupervised learning, where there is no explicit feedback on the performance (Hinton and Sejnowski, 1999).

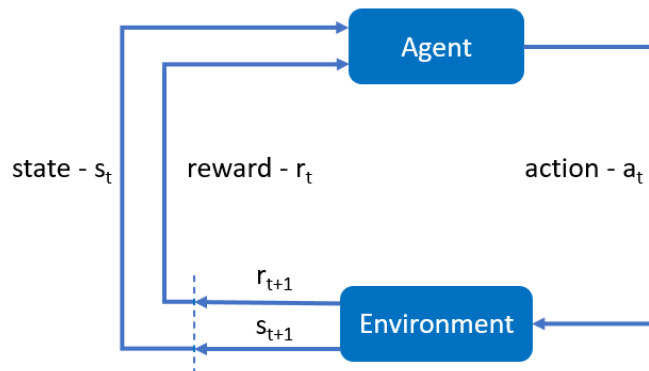


Figure 4 – Agent-Environment interaction

In the context of power and energy systems, with the increasing quantity of data generated by advanced sensors and smart meters, RL is a solution for control and optimization problems, since it can extract optimal operational knowledge from historical data through continuous interactions with the environment while the global optimum is unknown (Cao et al., 2020). Its versatility, mainly due to the personalized definition of environments and training strategies, allows its application to topics such as building control (Wang and Hong, 2020), negotiation of energy contracts (Pinto et al., 2019), control of heating and ventilation systems (Macieira et al., 2021), and electricity consumption forecast (Ramos, 2021).

Applying RL to a MAS can be challenging, but it will be beneficial for the system and for the agents that compose it. Two of the main challenges in the field are the definition of an appropriate formal goal for the learning multi-agent system, and the need for manually designing quality features on which to learn (Hernandez-Leal et al., 2019). Experience sharing can help RL agents with similar tasks learn faster and reach better performance. For instance, skilled agents may serve as teachers for the learner, or the learner may watch and imitate the skilled agents (Ndousse et al., 2021). Another advantage, that can be useful to increase the robustness of the system, is when one or more agents fail in a MAS, the remaining agents can take over some of their tasks (Hernandez-Leal et al., 2019).

Another challenge commonly associated with the development of RL models is the design of training environments. Particularly when dealing with applications in a real-world context (e.g., involving the modelling of a physical electrical network), there are many obstacles to its intuitive

development (Dulac-Arnold et al., 2021). This is because, since RL is a trial-and-error technique, the number of states in an environment, and of actions that can be performed on it, is exponentially proportional to the complexity of the environment itself (Nguyen et al., 2020). In addition, namely in the power and energy domain, the tools used to create training environments are not insightful, creating the need for the existence of tools that facilitate the integration of RL in previously designed and/or deployed systems, without demanding the agents and the system itself to be programmed again.

2.2.4 Interoperability

The basic principle of the MAS is communication among agents, and it is assumed that all parties involved understand each other, since it is through the exchange of information, knowledge, and services that the proposed objectives of the agents and the system itself are achieved (Lazaridou et al., 2017). This topic becomes even more complex when communication can occur among different MAS since the concurrent distributed development has led to many types of MAS that are islands of functionalities, whose agents are unable to interoperate with each other (Poslad and Charlton, 2001).

MAS are becoming increasingly heterogeneous, that is, they are composed of numerous types of agents, whether virtual or physical, which are frequently implemented in different ways (Lee and Shim, 2022). As a result, these systems must be able to transmit all the messages that agents send, which includes supporting various communication protocols. As is the case with IoT devices, different communication protocols are implemented based on the type of agent. It is important, therefore, not only to support these communication protocols but also to be able to translate messages from one communication protocol to another, so that they can reach their final destination regardless of their protocol or format (Derhamy et al., 2017).

In this way, transmitting a message is not enough to create interoperability among agents. Communication is characterized not only by the transmission of information but also by knowledge sharing and coordination among the involved parties. So, to make truly effective interoperability it is necessary to establish standards that consider communication characteristics, such as protocol and language, and semantics that represents, interpretably, the content of the message (Santodomingo et al., 2014).

Therefore, when designing MAS, the use of standards that promote interoperability between agents and systems is becoming increasingly important, if not mandatory (McArthur et al., 2007a). The FIPA *“is an IEEE Computer Society standards organization that promotes agent-based technology and the interoperability of its standards with other technologies”*, officially at the Institute of Electrical and Electronics Engineers (IEEE) since June 2005. From that moment on, FIPA standards began to be used often in the development of MAS (McArthur et al., 2007b).

Following the FIPA standards, agents from heterogeneous platforms should be able to interoperate, which does not mean that the agents are able to exchange any useful information. Even if the agents use the same communication language and content language, if they do not

share a common vocabulary, they will not be able to interpret the incoming messages, nor to communicate effectively.

A common approach to standardising the communication vocabulary among agents is the use of ontologies. In AI literature, one of the most agreed definitions of ontology was proposed by Gruber: *"Ontology is an explicit specification of a conceptualization"* (Gruber, 1993). More recently, Asim et al. defined ontology as the vocabulary and the formal specification of the vocabulary of a specific domain (Asim et al., 2018). The application of ontologies to MAS has been gaining popularity as the number of systems developed is increasing, and consequently, also their heterogeneity.

A problem normally associated with ontologies is the difficulty of implementing them, as well as the difficulty of integrating them into already developed MAS. Thus, other approaches to introducing interoperability in agents emerge, one of which is the definition of a development base, such as an agent-based ecosystem, to ensure that the agents and the systems into which they are integrated follow the same principles, particularly in terms of communication and the definition of their knowledge base (i.e., data and behaviours). This approach has the advantage that, although the agents must follow a particular development base, they can be adapted and customized by the developers, considering the system's and agents' objectives.

Although interoperability between different MAS is an essential topic for the success of the developed systems, it is also important to consider external and independent agents that want to participate or take advantage of the services provided by a MAS. Therefore, it should not be mandatory for a particular agent to be part of a MAS to use services provided by it, as long as it meets the requirements for participating in the system and/or services. For instance, an energy consumer may want to take advantage of the services of a MAS while making use of his own machine to host his agent, and thus take advantage of the system, without being an integral part of it. In this way, another need for interoperability arises, since, in addition to a MAS needing to understand and interoperate with external systems, it must also have the ability to communicate and interact with independent agents that are not part of it.

2.2.5 Agent-based Platforms and Frameworks

When designing and developing agent-based systems, programming language and platform selection become critical. Agent-based platforms are software packages that include the primary functionalities required to deploy and run MAS, hence simplifying their creation and deployment (Cardoso and Ferrando, 2021). On the other hand, an agent framework is a language environment, software library, or a combination of the two that provides the essential tools for constructing the agent-based system backbone. In this section are described some of the most used agent-based platforms and frameworks.

Java Agent DEvelopment Framework (JADE) is an open-source platform for peer-to-peer agent-based applications (Kumar and Kumar, 2014) and it was developed to create a FIPA-compliant framework that would allow MAS developed according to the FIPA specifications to be more

extensible and compatible. It includes agent abstraction, a simple but powerful task execution and composition model, asynchronous peer-to-peer agent communication, and a yellow pages service with publishing, subscribing, and discovering mechanisms, among many other advanced features that make developing an agent-based system easier and less time-consuming. The authors of JADE say that, unlike architectures that rely on a single entity, a peer-to-peer architecture lowers failure circumstances. JADE agents communicate by exchanging messages using the FIPA ACL language.

The Open Agent Architecture (OAA) is presented as “A framework for integrating a community of heterogeneous software agents in a distributed environment” (SRI International, 2022). It was created to allow heterogeneous software agents to work together in distant contexts. The dynamic and extensive character of black board-based systems, the efficiency of moving items, and the vitality of the agents' interactions are all provided by OAA. The InterAgent Communication Language (ICL), a logic-based declarative language capable of expressing high-level, complicated tasks and natural language expressions, is used in OAA for inter-agent communications. It features a conversational protocol that is similar to KQML's communication layer and is specified by a list of event kinds and related parameters. The content layer, on the other hand, is similar to the one in KIF and consists of specific goals, triggers, and data items that may be placed within various events.

Mesa is a Python-based agent-based modelling framework (Masad and Kazil, 2015). It enables the rapid building of computational models based on agents using core components like Spatial Grids and Agent Scheduling, as well as bespoke components. Agent-based models are computer simulations in which several agents behave and interact with one another based on their actions. Because these models assess a system at both an individual (each entity/agent) and a system-wide level, they may determine the impact that each agent has on the system. Mesa has a modular framework that simplifies the addition of new components.

The Smart Python Agent Development Environment (SPADE) is a Python-based multi-agent systems platform based on the Extensible Messaging and Presence Protocol (XMPP) (Palanca et al., 2020). Because SPADE is an asynchronous system, it is quick and efficient. The XMPP protocol is flexible and extendable, allowing the integration and setup of a variety of functionalities, with instant messaging as the primary focus. Because of the modular nature of XMPP, SPADE can interact with agents using any XMPP server. The FIPA specifications can be integrated thanks to this protocol. It also permits the usage of presence notifications, which allow the system to know the current condition of each agent in real-time.

2.3 Machine Learning

Machine Learning (ML) is a subfield of Artificial Intelligence that studies the mechanisms by which a computer may learn to recognise patterns in a given set of data (Pouyanfar et al., 2018). Its objective is to enable a system to enhance its ability to accomplish a certain activity without being explicitly coded. According to Flach, machine learning is the process of developing a

model that enables a computer to complete a job based on a collection of attributes extracted from training data (Flach, 2012). The learning process is described in the literature as transforming experience into knowledge, in which the training set used reflects the experience and the output symbolizes the knowledge that can be applied to different datasets (Shalev-Shwartz and Ben-David, 2014).

While ML algorithms offer the benefit of not requiring explicit programming to enhance their forecast performance in response to fresh evidence in the form of extra data, their use does have certain limitations. To begin, if an ML method is not correctly parameterized, it is susceptible to overfitting, a phenomenon in which the model is not sufficiently generalised to make predictions outside of the values encountered during the training process (Ying, 2019). Similarly, machine learning algorithms have limited applicability outside of the contexts in which they were trained, particularly if the correlations upon which they are based are context-specific and the discrepancies with other settings are accurately modelled by the selected features, or if there is a dearth of data for other contexts. Despite this, the biggest problem with ML resulted from its greater use in sensitive areas, such as health. This is because there are several questions about the fairness and bias of the models (Mehrabi et al., 2021).

There are different types of machine learning, each one addressing different types of problems and contexts. The most addressed ones are supervised, unsupervised, and reinforcement learning. Despite that, there is semi-supervised learning, which combines both supervised and unsupervised learning, using labelled and unlabelled data for training (Zhu, 2005). In Figure 5 semi-supervised learning is compared to supervised and unsupervised learning in what regards data labelling.

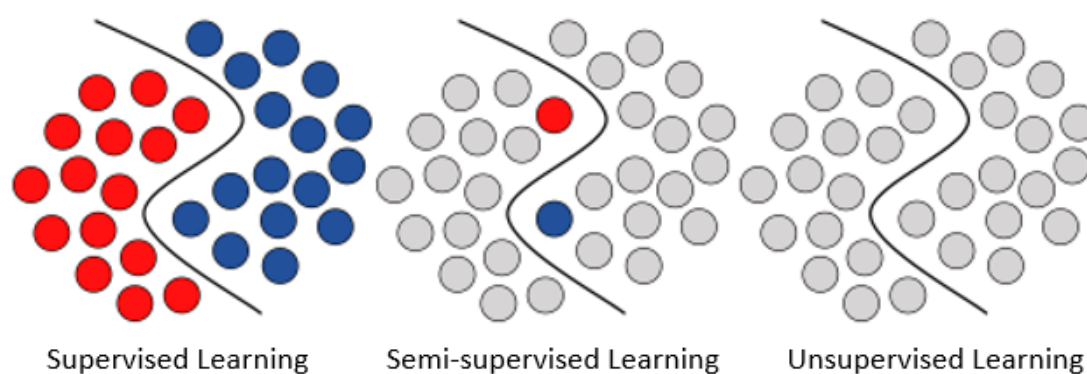


Figure 5 – Comparison between supervised, semi-supervised, and unsupervised learning data labelling (adapted from (Schmarje et al., 2020))

2.3.1 Supervised Learning

Supervised learning is a subfield of ML in which both the input and output data of datasets are known throughout the training process. Supervised algorithms are capable of analysing and forecasting outputs for unknown input data, based on the knowledge obtained from the

training dataset (Ahmad and Chen, 2020). It is the most addressed type of ML, mainly due to its simple and quick application to a wide range of domains (Kulkarni et al., 2021).

Supervised learning tasks may be generated using one of two separate types of output labels: discrete data (e.g., recognising the appliance that is generating a specific load pattern) or continuous data (e.g., estimating energy prices). Problems that deal with the discrete target data are described as classification problems, while if the target data is continuous it is considered a regression problem (Ahmad and Chen, 2020).

There are several algorithms to employ supervised learning. The most addressed ones are described as follows (Saravanan and Sujatha, 2019) (Gianey and Choudhary, 2018):

- Linear and Polynomial regressions: are used to estimate actual values using continuous variables as input;
- Logistic regression: it is a classification technique that is used to approximate discrete values from a collection of independent variables;
- Support vector machines (SVM): each property is mapped into a vector in n-dimensional space using a classification or regression algorithm. The data is divided into groups by tracing a space whose dimension varies with the number of characteristics;
- Decision trees: frequently employed in classification issues, this technique divides the population into homogenous groupings based on the most salient characteristics;
- Random forest: terminology used to refer to a collection of decision trees. To assign a new object a class based on its characteristics, each tree assigns a rating and vote to the class. The class with the greatest number of votes is chosen;
- K nearest neighbours (KNN): it's a straightforward algorithm that saves all possible situations and ranks them based on a majority vote from their K closest neighbours. It may be used for classifying and regression;
- Naïve bayes: Bayes' theorem-based classification technique. The Naive Bayes classifier assumes that the existence of one characteristic in a class is unrelated to the existence of any other attribute;
- Artificial neural networks (ANN): similar to the way a human brain performs, neural networks use communication between simple computing cells (called neurons) to generate predictions.

Traditional supervised learning algorithms have limitations in terms of their capacity to interpret unstructured natural data and also to process high volumes and variety of data. For decades, developing a system capable of pattern recognition required meticulous engineering and extensive domain knowledge, in order to construct a model capable of extracting suitable representations (Zhou et al., 2017). The emergence of the concept of Deep Learning (DL) has revolutionized this domain of AI. A deep neural network (DNN) may express progressively complicated tasks by adding additional hidden layers and units to each layer. DNNs may learn as a supervised algorithm, where the training data contains the target attribute to predict or

classify, or as unsupervised, where training data is created automatically from unlabelled data with small human interaction. (Zhou et al., 2017). Fundamentally, the goal of DL is to deliver solutions to issues that are too complicated for humans to handle without their assistance. With the advancement of technology that enable Deep Learning approaches and innovation in the field, a variety of models have been suggested, each with a distinct application and purpose. Table 5 contains a summary of the deep learning networks (Pouyanfar et al., 2018).

Table 5 – Deep Learning models comparison. Adapted from (Pouyanfar et al., 2018)

Network architecture	Key Points
Recursive Neural Network	Uses a tree-like structure; Preferred for NLP
Recurrent Neural Network	Good for sequential information; Preferred for NLP & speech processing
Convolutional Neural Network	Originally for image recognition; Extended for NLP, speech processing, and computer vision
Deep Belief Networks	Unsupervised learning; Directed connections
Deep Boltzmann Machine	Unsupervised learning; Composite model of RBMs; Undirected connections
Generative Adversarial Network	Unsupervised learning; Game-theoretical framework
Variational Autoencoder	Unsupervised learning; Probabilistic graphical model

In the scope of the smart grid, supervised learning is mainly used in energy consumption forecasting (Seyedzadeh et al., 2018) (Ramos et al., 2022). Despite that, many other applications are possible. In (Ebrahimian et al., 2018) is proposed an energy prices forecast with the use of a three-stage neural network. In the domain of fault detection, in (Jan et al., 2021) is proposed a distributed sensor fault detection and diagnosis framework using support vector machines, a fuzzy deep neural network, and an auto-encoder, that shows that multiple ML models can be combined to enhance the accuracy in predictions. Another commonly addressed domain is load scheduling, and in (Kaur et al., 2021) is proposed a price-based load scheduling for optimal IoT control in smart homes, using Adaptive Boosting, a statistical supervised algorithm. Other topics in the context of the smart grid addressed with supervised learning are: (i) energy resources optimization (Mota et al., 2021), adaptive control (Kolluri and De Hoog, 2020), distributed generation units sizing (Purlu and Turkay, 2021), data-breach detection (Cui et al., 2020), and EV scheduling (Shahriar et al., 2020).

2.3.2 Unsupervised Learning

In contrast to supervised learning, unsupervised learning treats all objects in the dataset identically, with no difference made between the training and testing sets. This occurs because the target values are unknown from the beginning of the training process. As a result, the major goal of these types of algorithms is to independently find hidden patterns or data groupings

(Shalev-Shwartz and Ben-David, 2014), and also to detect anomalies in a dataset (Meira et al., 2022), for instance, to predict security breaches.

Unsupervised Learning is divided into 3 major topics: clustering, dimensionality reduction, and association rules. Clustering is the most researched topic in unsupervised learning, with the goal of dividing a finite, unlabelled dataset into a finite number of groups, named "clusters", based on data similarities (Xu and Tian, 2015). The typical result of a clustering algorithm is a scatter plot indicating the formation of clusters/groups, as seen in Figure 6 (Varghese et al., 2018). Dimensionality reduction is a technique for lowering the dimensionality of characteristics while retaining the data. In a truncated or low-dimensional dataset, critical characteristics persist even if a particular pattern disappears (Zebari et al., 2020). Association rules are typically used for knowledge discovery and pattern discovery, in which associations are created among data characteristics (Wu and Wang, 2020).

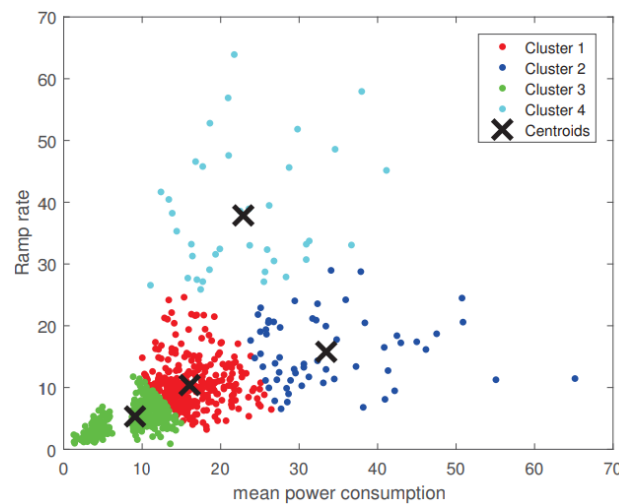


Figure 6 – Clustering example. Extracted from (Varghese et al., 2018)

Unsupervised algorithms can also be applied to consumers categorization or segmentation (Rajabi et al., 2020). Electricity customer categorization might theoretically follow segmentation standards based on commercial categories of activity. However, the load patterns of clients engaged in the same sort of activity might vary significantly (Chicco, 2012). As such, classifications based on activity type and business codes are often inefficient at capturing the individual characteristics of power usage. The distinction can then be made between a few macro-categories (e.g., residential, industrial, commercial).

Clustering algorithms are often employed in power and energy systems to conduct an in-depth analysis of customers. Clustering enables customers' aggregation (Iria and Soares, 2019b), for instance, to enhance their participation in DR (Lin et al., 2019). Multiple clustering algorithms, including K-Means, are explored in (Teichgraeber and Brandt, 2019) to optimize power systems via the identification of relevant periods. Additionally, as described in (Nepal et al., 2019), it is feasible to examine electricity consumption trends and optimize power system design (Kotzur et al., 2018). Coefficients relating to intra-group homogeneity and inter-group heterogeneity,

such as Elbow and Silhouette, are typically used as assessment measures to determine the ideal number of clusters.

2.3.3 Reinforcement Learning

Reinforcement Learning (RL) is a type of learning in which an agent/learner interacts with their environment through trial and error. In contrast to other machine learning techniques, the agent is not instructed on the appropriate course of action to pursue. Rather than that, the agent explores the environment to maximize its future rewards (or, statistically, the total of expected rewards), which is typically the quest for the achievement of a goal/objective (or a target space) represented numerically by a large reward. (Recht, 2019).

The agent is placed in an environment and provided with an observation, such as the system's condition. The agent is then free to choose from a predefined set of actions that, when done, alter the environment. It will be rewarded if it performs something positive; if it does not, it may also receive a penalty (Shin et al., 2019). In this way, the agent explores/exploits the environment and can learn a good strategy, in RL called a policy, by maximizing the rewards. The policy indicates the likelihood of an action occurring in a certain state. In a conventional RL environment, the model is a Markovian Decision Process (MDP), and the rewards should also be Markovian. MDP's objective is to provide a mapping of ideal actions for each environment state. MDP is based on Markovian property, which considers just current events, and the prediction of the upcoming state is decoupled from previous states (Naeem et al., 2020). In a summary, this means that the rewards only depend on the current state, not on how the agent got there.

In some cases, pursuing instant rewards is counterproductive, as the long-term rewards may be greater for another action. If the agent chooses a lesser immediate reward, it may receive a bigger cumulative reward in the long run. If given sufficient time, the agent can learn this policy (Han et al., 2021). Additionally, a discount function can be utilized to communicate the relative importance of future rewards to those received at the moment. Specific RL approaches employ a value function to decide whether activities produce desirable outcomes in the future, as opposed to immediate feedback rewards. The main goal of learning is to maximize the expected discounted cumulative reward.

After training, the agent's policy dictates which actions will be taken in response to the observed state of the environment. It is available in two alternatives: deterministic and agnostic. The latter returns a stochastic policy, in which a distribution (action, state) specifies the likelihood of the agent doing a certain action given a specific state. On the other hand, the deterministic policy function maps a given state to an action (Sutton and Barto, 2018).

The exploration vs. exploitation dichotomy is a recurrent subject in RL and artificial intelligence more broadly. "Should we exploit acquired information?" (i.e., should we take a known, high-reward path), or "Should we explore uncharted states in quest of a more effective new policy?" (Arulkumaran et al., 2017). The combination of the two significantly boosts the agent's learning

performance. One probable explanation emerges: To begin, an agent must investigate the greatest possible number of states, followed by utilising obtained knowledge to obtain superior results once the agent is confident of having thoroughly studied the universe. It is extremely difficult to determine whether sufficient exploration has been conducted in uncertain/dynamic environments (Padakandla, 2021).

The exponential growth of deep learning eventually extended to reinforcement learning, which had a beneficial effect on its potential applications. Deep Reinforcement Learning (DRL) combines the sensing capability of deep learning with the decision-making capability of RL (Botvinick et al., 2019). Deep learning extracts information about the goal observation from the environment and delivers state information about the current environment. The RL algorithm then transfers the current state to the associated action and assesses values in terms of expected return (Arulkumaran et al., 2017). Figure 7 presents the comparison between a RL and a DRL algorithm (i.e., Q-Learning vs. Deep Q-Learning). A continuous interaction process transforms decision-making behaviour into a step-by-step process (François-Lavet et al., 2018). DRL resolves typical problems in RL by completing complex tasks with less prior knowledge, owing to its capacity to learn abstraction levels from data (Agostinelli et al., 2018).

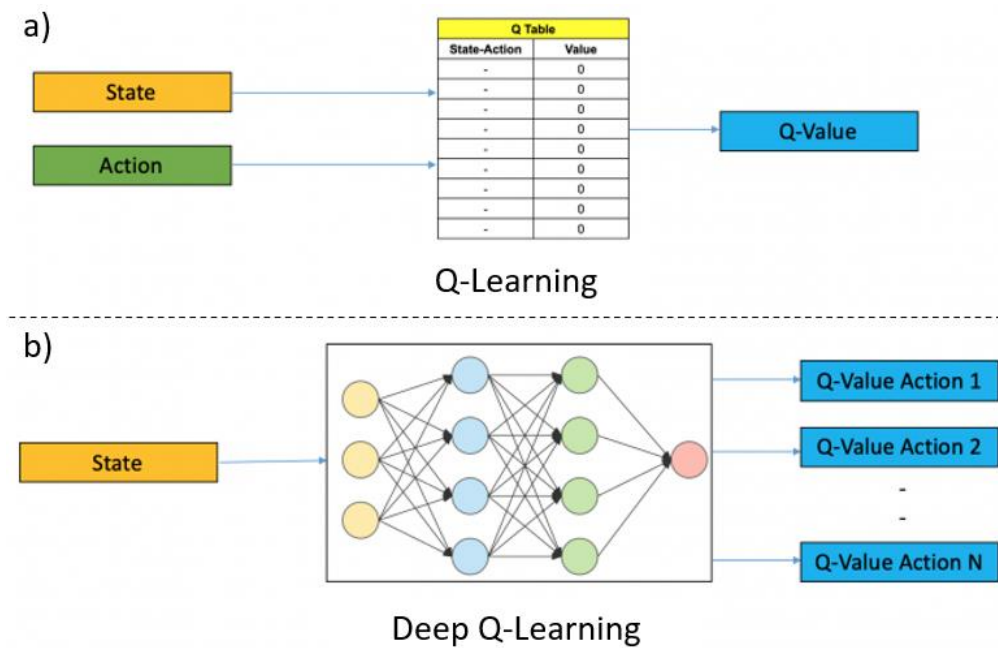


Figure 7 – Comparison between a) Q-Learning, and b) Deep Q-Learning. Adapted from (Choudhary, 2019)

In the domain of smart grids, RL and DRL have been widely used to solve several complex problems. It can be used for smart building energy management, both with model-based and model-free algorithms, as proposed in (Yu et al., 2021). The conclusions extracted from (Yu et al., 2021) is that, for this specific task, model-based algorithms are more practical than model-free, as the majority of the model-free algorithms are not yet implemented in practice due to a long exploration time and a high exploration cost. Besides that, it can be used to model and

optimize the participation in DR programs (Vázquez-Canteli and Nagy, 2019) and P2P (Chen et al., 2022), electricity markets simulation and strategic bidding (Ye et al., 2020), operational control (Diao et al., 2019), among others (Cao et al., 2020).

2.4 Chapter Conclusions

While MAS developed for the smart grid have well-defined objectives, they are frequently restricted to specific domains (e.g., energy community representation, DR, and P2P simulation). Thus, there appears to be no system in which multiple MAS can coexist and communicate, making it difficult to fully represent the smart grid and its entities.

As a result, it is essential to conceive an ecosystem capable of integrating multiple agent-based systems, with the capacity to model even the most complex interactions and representations present in the smart grid, such as the dynamic relationships created by the provider-client paradigm, in which clients should be able to search for the best service providers for their context, naturally and seamlessly. The majority of the interactions in the smart grid are based on the provider-client paradigm, creating the opportunity to directly apply the ecosystem to that specific domain. Besides being useful to better represent the smart grid, an agent-based ecosystem can be considered as an implicit solution for agents' interoperability, since it allows multiple systems and agents to be developed and deployed under the same standards.

In addition to the capabilities used to represent the smart grid, it is possible to conclude that agent-based systems have deficiencies concerning data and objectives overloading, and also in what regards computational performance. Despite that, multiple techniques have already been proposed to address the agents' overloading and performance. For instance, there is the agent cloning technique, which even though being useful for dividing the agents' tasks, it does not ensure that the agents' objectives are met, much less following the requirements outlined by the entities they represent. Besides that, MAS should consider techniques to balance the computational load of its agents, especially when the agents are hosted in multiple physical hosts with different computational power.

Reinforcement learning is one of the techniques that is intended to be integrated into the ecosystem to address the energy customers' active participation in energy communities due to its ability to introduce the characteristics of intelligence and learning in the agents. Although reinforcement learning is a vastly explored technique, its development for MAS is typically not intuitive and highly dependent on a large number of parameters. Normally, MAS have to be developed from scratch to integrate RL training and simulation, not allowing already developed and deployed agents to enhance their behaviour in real-time. This capability would be a big step towards the application of RL in real-context MAS, where agents represent physical devices.

3 Methods and Materials

This chapter describes the methods/algorithms used in the development of the agent-based ecosystem, and the tools used both to develop it and to test it. Besides that, this chapter presents the technological and social challenges related to the domain of development.

3.1 Materials and Tools

This section describes the material and tools that enable the development of the proposed system. The obtained data is described (in this case from energy communities), the MAS development framework, machine learning libraries that enable an intuitive implementation of sophisticated algorithms, and ontologies established for the modelling of smart grids.

3.1.1 Multi-Agent Systems Development Frameworks

The first step in the development of a multi-agent system is the choice of a platform or development framework, or, in specific cases, agents can be developed from scratch, something that involves other responsibilities and concerns in their implementation. The platform chosen for the development of the system and its agents was the Python-based framework for hEterogeneous Agent Communities (PEAK)⁴ (Bruno Ribeiro et al., 2022). This decision is due to three main reasons: the first is that JADE, the most used platform for the development of multi-agent systems, has already been properly explored and its innovation is already limited since its last version is dated from 2017; the second one is that PEAK is developed over SPADE, which by default uses XMPP to enable the agent's communication, which is a naturally extensible protocol; and the third one is that PEAK is implemented in Python, therefore being possible to extract the full potential of the language in the development of the ecosystem.

⁴ PEAK website - <https://www.gecad.isep.ipp.pt/peak>

SPADE is a framework that is regularly updated, is open-source and has the MIT license. SPADE is a modular framework, which allows its expansion using plugins. It is FIPA-compliant and its communication architecture, despite being different from the mainstream architecture, it ends up being more efficient because it provides many functionalities used for agent communication. This type of architecture allows human-agent interaction more easily, allows a more reliable communication between agents, because if one agent goes offline the conversations are not lost, and adds the ability to find the other agents more easily without the need for a facilitator agent. However, SPADE can be improved, because despite providing various tools for MAS development it lacks some fundamental functionalities, for instance, proper management tools for the MAS and better mechanisms for group communication. For these reasons, PEAK was used since it covers a lot of the SPADE’s limitations.

PEAK, which architecture is shown in Figure 8, is a framework that facilitates and provides every tool needed for a developer to create a MAS and integrate it into an agent-based ecosystem. helps to create any type of agents, create simulation environments, extract data from the MAS and create group chats where agents can communicate with each other easily. Additionally, this component has tools to facilitate the integration of IoT technologies in the MAS, as it is possible to use the Modbus TCP and the HTTP protocols and is flexible to use other protocols as well. The PEAK framework contains a dashboard component, which is a graphical user interface that allows to visualize, interact and monitor the PEAK agents. The dashboard to work must be initially connected to the Directory Facilitator (DF) agent REST API. One of the main features of the Dashboard is the ability to visualize the entire hierarchical architecture of the ecosystem.

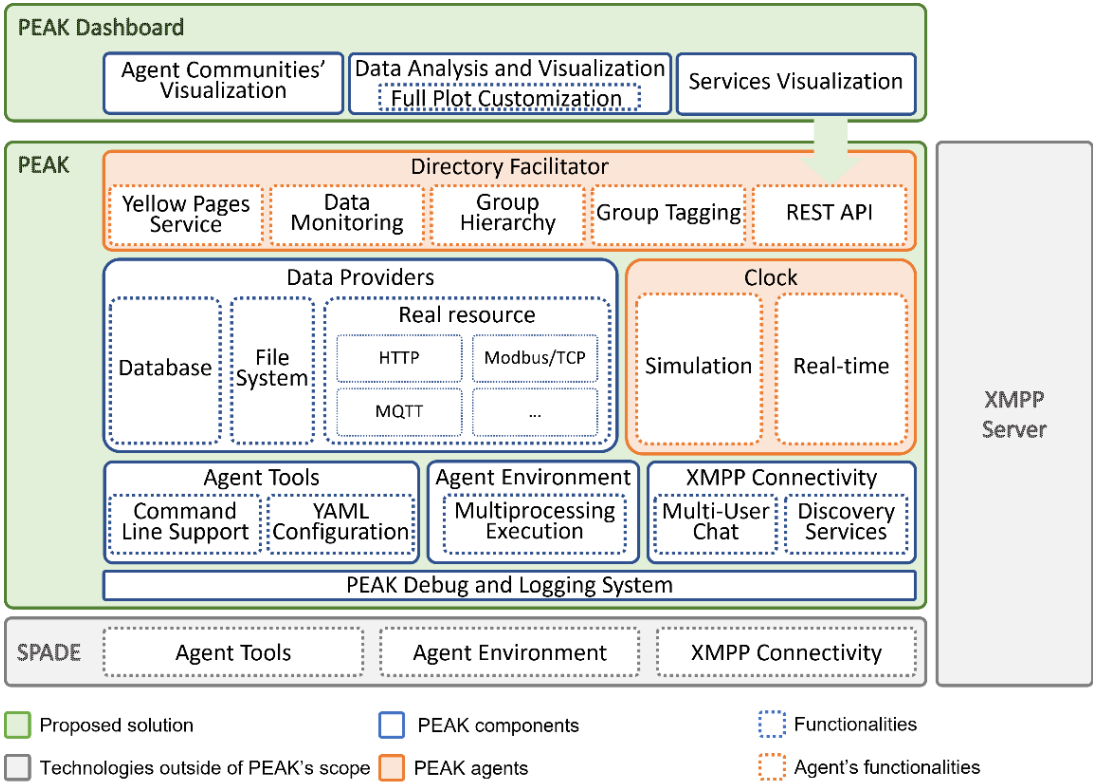


Figure 8 – PEAK framework architecture

The DF agent is crucial to the general maintenance of the ecosystem. This agent is responsible for numerous tasks, with the following being the most important in the context of the proposed ecosystem: (i) registering agents and groups in the ecosystem, allowing the establishment of a multi-level tree-like hierarchy, and (ii) providing a Yellow Pages Service that facilitates the discovery of all registered agents, services, and groups. Tags can be associated with the register to enable a faster and more accurate search.

Since the communications of PEAK agents are carried out through the XMPP protocol, then all of them must be registered in an XMPP server. The chosen server was Prosody⁵, which is a recent and modern XMPP server software. It is extremely simple to install and configure and consumes minimal system resources. Additionally, it is quite simple to edit and change, as well as test with new protocols. It is one of the most efficient servers, as it can operate on low levels of RAM. Another advantage of Prosody is that many of its components are modular, allowing for the addition or removal of functionality in the application (Aryanto et al., 2021). Additionally, it can be connected to other XMPP servers on a network to function as a multi-server system. It is extremely efficient and secure; all communication is encrypted. The most important modules in Prosody in the context of this work are “Bosh” and “WebSockets” since they allow communication over HTTP, which is critical for the proposed model since in this way they enable direct communication between the platform's graphical user interface and the agents.

To be able to regulate interactions between agents and also to improve the capacity to identify inaccuracies in them, it is critical to have tools that allow for the recording and monitoring of communications in real-time. For this task, the used tool is Pidgin⁶, a chat client that enables logging into several chat networks at the same time, including XMPP groups. Pidgin integrates a number of these chat networks' capabilities, including file transfers, away messages, friend icons, custom smileys, and typing notifications. Numerous plugins additionally extend Pidgin's capabilities beyond the standard features.

3.1.1 Python and Machine Learning Libraries

To develop and implement the models proposed in this work, multiple libraries and modules are used, in order to complement each other. This type of library saves developers time and work, besides allowing them to concentrate on applying the models to their systems.

In terms of data analysis and general software development, several libraries were used. Of these, two are particularly useful when it comes to data representation and processing: Pandas (Reback et al., 2022) and NumPy (Harris et al., 2020). Pandas is a library which provides fast, versatile, and expressive data structures meant to make working with “relational” or “labelled” data both easy and natural. It is highly suited for many various forms of data, such as tabular data and any other form of observational/statistical datasets. NumPy is a library for scientific computing that provides a multidimensional array object, various derived objects, such as

⁵ Prosody website - <https://prosody.im/>

⁶ Pidgin website - <https://www.pidgin.im/>

masked arrays and matrices, and a variety of routines for fast array operations, namely mathematical, logical, and sorting.

In what regards code analysis two libraries were used: Pylint⁷ and Pysa⁸. Pylint performs code analysis without actually executing it. It checks for mistakes, enforces a coding standard, searches for code smells, and makes suggestions for refactoring the code. Pylint is extremely flexible and allows the development of plugins to provide custom inspections (for example, for internal libraries or an internal rule). Pysa - Python Static Analyzer, produces a different type of code analysis, named taint analysis. It operates by tracing data flows from their origin to their destination. For instance, it monitors flows in which user-controlled request data flows into a function call, resulting in a remote code execution vulnerability.

Regarding machine learning, multiple libraries have been deeply explored and used. Scikit-learn is a Python module that integrates a diverse set of cutting-edge machine learning methods for solving supervised and unsupervised issues on a medium scale (Pedregosa et al., 2011). This package aims to make machine learning accessible to non-specialists through the use of a general-purpose high-level language. TensorFlow is a machine learning module that operates in large-scale heterogeneous environments, with a major focus on deep learning models (Abadi et al., 2016). TensorFlow represents computation, shared state, and the operations that modify that state using dataflow graphs. In what regards RL development, the environment design has a big impact on the problem's definition, and on the model's performance. Gym is a "*toolkit for developing and comparing reinforcement learning algorithms*" (Brockman et al., 2016) which was created by OpenAI⁹. Their primary goals were to advance the area of reinforcement learning by establishing a uniform benchmark and standardising conditions that facilitate the implementation of publications in real-world systems. Lastly, the Ray RLLib library (Liang et al., 2017) provides the implementation of several RL algorithms, and its main highlight is the direct integration of multi-agent RL algorithms. If the environments where the algorithms are applied are OpenAI Gym-compliant, then the integration between the two libraries is straightforward.

To finalize, it was a library to model and represent the physical energy network. Pandapower is a Python-based, BSD-licensed power system analysis tool aimed at the automation of static and quasi-static analysis and optimization of balanced power systems (Turner et al., 2018). It provides services such as power flow, optimal power flow, and state estimation.

3.1.2 Datasets

Even though multi-agent systems are not always data-driven models, data are required for the agents to learn from experience and to find the optimal context given their characteristics. As such, four datasets, each one with specific objectives, were collected to be then used in the creation and development of the agents.

⁷ Pylint website - <https://pylint.pycqa.org/>

⁸ Pysa website - <https://pyre-check.org/docs/pysa-basics/>

⁹ OpenAI website - <https://openai.com/>

The first acquired dataset was studied in (Pereira et al., 2021), and it is based on the one presented in (Silva et al., 2019). The dataset contains information on 996 customers, considering 5 full days, with a granularity of 15 minutes. The information about the dataset features is described in Table 6. This dataset is very useful for the characterization of customers since it contains a large number of features that can be essential for differentiating them, considering other topics than the historical profiles of consumption and generation. The customers represented in the dataset do not have a connection among them, and thus, they can be grouped as several aggregation entities, for instance, energy aggregators, retailers and communities.

Table 6 – Description of the first used dataset’s features

Feature Name	Description	Unit
Consumption Profile	Energy consumed during each period	kWh
Generation Profile	Energy generated during each period	kWh
Consumption Type	Type of customer, i.e., domestic, small commerce, medium commerce, large commerce, or industrial;	-
Generation Type	Type of generation of the customer, i.e., photovoltaic, wind, biomass, or hydrogen;	-
Consumption Tariff	Energy consumption tariff applied	EUR/kWh
Generation Tariff	Energy generation tariff applied	EUR/kWh
Flexibility	Consumption flexibility for each period	%
Contracted Power	Power defined in the consumption tariff contract	kVA

The second dataset contains an energy community with 96 real customers’ profiles, half of them being consumers, and the other half being prosumers. It was used in (Pereira et al., 2022a). The dataset uses real consumption and generation profiles from (Working Group on Intelligent Data Mining and Analysis (IDMA), 2022) and (Hebrail and Barard, 2012). The energy flexibility, of each customer, was generated using a normal distribution with a mean of 15 % and a standard deviation of 5 %. Regarding energy prices, the Iberian Electricity Market (MIBEL) market prices were used (OMIE, 2022). The dataset has a granularity of 15 minutes, and it represents a week with seven days (i.e., 672 periods).

The third dataset, published in (Pereira et al., 2022b), was conceived with the main objective of testing the mobility functionality proposed in this work. The dataset has 30 customers: ten residential, ten small commerce, five large commerce, and five industrial customers. The combination of several energy customer types allows the creation of a dataset with different types of consumption profiles, generation, and flexibility, and, therefore, different values of participation in DR events. The residential profiles of the considered customers use the data available in the (Working Group on Intelligent Data Mining and Analysis (IDMA), 2022). The values represent a week period using 15 minutes reading periods. All the values are represented in kWh.

The last used dataset represents a realistic energy community, and it is available in (Goncalves et al., 2022). The data describes an electrical energy community, containing photovoltaic (PV) production profiles and customer consumption profiles, desegregated by individual appliances used. A dataset of a residential community was constructed based on real data, where sample consumption and photovoltaic generation profiles were attributed to 50 residential households and a public building (municipal library), a total of 51 buildings. The data concerns a full year. Despite the overall power consumption of these houses being desegregated into the consumption of 10 commonly used appliances using real energy profiles, in the scope of this work, only the aggregated data were used. Also, in the scope of this work, only the residential households' data were used.

Besides the use of datasets, the agents must have the capacity to acquire data from other sources. For instance, the agents are able to acquire data in real-time, with connections to real sensors and external devices, to create a more robust and representative model of their entities. In this way, the agents must have the capability to access external data and interpret it correctly. Missing data is created using simulation engines and manual creation. The manual creation is used to carry out controlled tests of specific agents to evaluate their individual performance in the system.

3.2 Methods and Models

In this section are explored the chosen methods, algorithms, and paradigms for the development of A4SG and its functionalities. The addressed topics are the customers' aggregation algorithm, MAS organizational paradigms, and the RL algorithm used to optimize participation in P2P energy markets.

3.2.1 Customers' Aggregation

The customers' aggregation is essential to optimize the mobility functionalities proposed in this work, to reduce the possible options of mobility of an agent depending on its objectives and context. The goal is to find to most similar agents to the one using the mobility functionality and use that information to give the best mobility options to an agent.

The chosen algorithm to perform the customers' aggregation/segmentation was the K-Means clustering. In the term K-Means, the "K" refers to the number of clusters. Typically, the value of "K" is unknown and must be determined by the developer. Each cluster has a centroid, which is often calculated as the mean of the cluster's feature vectors (Sinaga and Yang, 2020). Each data point is assigned to a cluster using the K-Means clustering technique based on the cluster centroid that is closest to the point. Because centroids cannot be determined directly until clusters are created, the developer sets "K" initial centroids values at the start of the clustering process. After clusters have been established, the actual centroid values are determined.

The evaluation of clustering algorithms is performed using techniques that determine the ideal number of clusters to a dataset (or neighbourhoods, in the case of density-based clustering), such as Elbow, Silhouette, and Davies-Bouldin Index. The Elbow technique takes the total Within-cluster Sum of Squares (WSS) into account as a function of the number of clusters that are acceptable for a dataset (Marutho et al., 2018). When the function curve forms an elbow, the optimum number of clusters is obtained. The Silhouette Score (SS) is a criterion for cluster cohesiveness (Choi et al., 2019). It computes not only the similarity of a point to the rest of its group but also the dissimilarity of the same element to the points of other groups. Thus, a value between -1 and 1 is produced, where 1 indicates that groups have a high degree of cohesiveness and -1 indicates that individuals of the same group do not conform to one another (Lengyel and Botta-Dukát, 2019). The Davies-Bouldin Index (DBI) uses a positive correlation for the ‘within-class’ measurement and a negative correlation for the ‘between-class’ measurement (Xiao et al., 2017). DBI is used as a metric because the clustering validation contains two main categories: external and internal validation.

Before the application of clustering algorithms, typically, there is a step where occurs dimension reduction. This enables clustering to be applied to a dataset with a smaller dimension, which improves prediction accuracy by reducing the number of features and making them easier to classify (Huang et al., 2019). Principal Component Analysis (PCA) is an algorithm for dimensionality reduction. PCA provides a roadmap for reducing a complex data set to a lower dimension to reveal the simplified structures that frequently underlie it, with minimal effort. PCA maps n-dimensional data onto a d-dimensional subspace while minimizing the sum of squared errors or maximizing the variance (Huang et al., 2019). Figure 9 exemplifies the application and possible results of PCA.

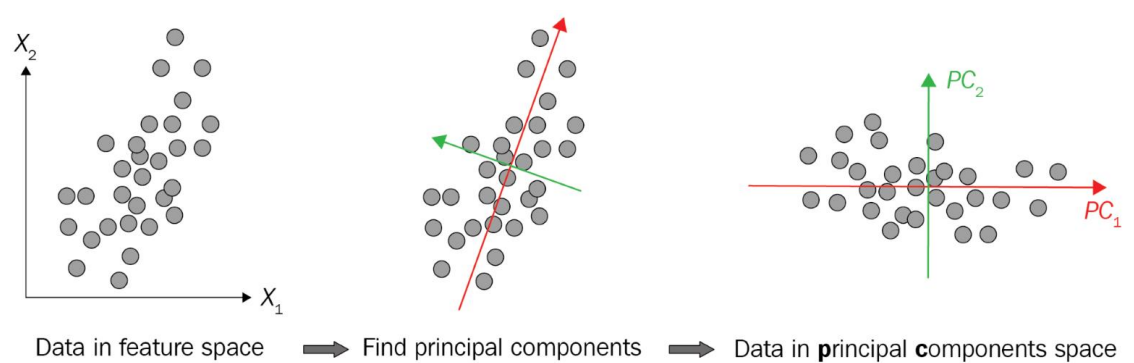


Figure 9 – Dimensionality reduction exemplification. Extracted from (Karim, 2019)

3.2.2 Multi-Agent System’s Organizational Paradigms

Multiple organizational paradigms are considered in A4SG with the purpose of coordinating the different agent-based systems, groups, and agents in a unified solution. The organization types were chosen based on the goal of each group and the communications and interactions that their agents can perform.

The ecosystem organization can be seen as a society of MAS (Horling & Lesser, 2004), whose agents can communicate and interact openly in pursuit of their individual objectives, and in certain cases group/community objectives. In addition to the communication that can be carried out among agents in the same MAS, the proposed solution also enables communication among agents within different MAS. For the proposed society to be viable, it must be governed by rules and norms that its agents must comply with, to guarantee the proper operation of the ecosystem. These rules can validate the mobility of agents, their branching, and their connections to other agents, among others.

The organizational type associated with the relationship between MAS and agent communities are holarchies, as they are multi-levelled, grouped hierarchies (Horling & Lesser, 2004). In other words, this organization establishes that one entity or group contains other groups that, while sharing certain characteristics with the group that contains them, are different in their own right, with objectives that may or may not be related or complementary. Therefore, in the context of A4SG, a MAS can contain multiple agent communities.

In each agent community there may also be other groups, also agent communities, but with more specific objectives, which, for instance, can bring together agents who register to serve and participate in specific types of services. These groups follow the paradigm of congregations, which generally are long-lived, utility-directed groups (Horling and Lesser, 2004). In the context of this system, these groups, among other possible applications, have the main objective of facilitating the discovery of agents that perform a particular service that the customers seek. Taking the example of the DR services, it is possible to have multiple agent communities, to serve different types of DR (i.e., such as real-time pricing and direct load control), in an agent community which is composed of all the participants of the DR events.

3.2.3 Energy Peer-to-Peer Trading Markets Participation Optimization

The integration of reinforcement learning in agent-based systems is a state-of-the art approach to integrate learning and intelligence characteristics into agents. As such, in the context of this work, it is proposed a reinforcement learning model to enhance customers' participation in P2P programs.

Traditional reinforcement learning algorithms are tabular, that is, all the values obtained from the training are stored in tables, which brings critical drawbacks. Firstly, the tabular structure constrains the algorithms to situations with few states and actions, since the state space of the majority of real-world situations is too big to be stored on a standard computer. Second, when a tabular structure is used, the method is unable to use similarities between states to exchange information. To circumvent these constraints, it is usual to replace the tables with function approximators that learn to map between the features characterising the state of the environment and the values of the estimated function. A deep neural network is the most frequently utilized function approximation for reinforcement learning. Their capacity to estimate non-linear functions and extract meaningful characteristics from raw inputs enables

generalisation to previously unexplored states. One of the most used and explored algorithms is Q-Learning, but it only supports discrete observation and action spaces, which prevents it from being used originally in many smart grid situations. Its adaptation deals with continuous spaces, which involves the discretization of actions and observations, which can result in the loss of valuable information, preventing optimal results. Therefore, two RL algorithms will be explored, the Deep Deterministic Policy Gradient (DDPG) and Twin Delayed DDPG (TD3).

DDPG is an algorithm that simultaneously discovers a Q-function and a policy. It employs off-policy data and the Bellman equation to learn the Q-function, then employs the Q-function to learn the policy. This method is closely related to Q-learning and is similarly motivated: if you know the optimal action-value function $Q^*(s, a)$, then in any given state, the optimal action $a^*(s)$ can be found by solving

$$a^*(s) = \arg \max Q^*(s, a). \quad (1)$$

DDPG interleaves learning an approximator to $Q^*(s, a)$ with learning an approximator to $a^*(s)$, and it does so in a way which is specifically adapted for environments with continuous actions.

As with many RL algorithms, DDPG training may be unstable and highly dependent on finding the optimal hyperparameters. This is because the algorithm consistently overestimates the Q values of the critic network. Over time, these errors may drive the agent to reach a local optimum or develop forgetfulness for prior experiences. TD3, proposed in (Fujimoto et al., 2018), solves this problem by concentrating on decreasing the overestimation bias. TD3 is an algorithm that tackles this problem by proposing three crucial techniques:

- Using a pair of critic networks: TD3 tends to underestimate Q values. This underestimating bias is not a concern because low values are not propagated through the algorithm, unlike high values. This technique provides a more stable approximation, hence enhancing the algorithm's stability;
- Delayed updates of the actor: TD3 allows the definition of the delay periods to update the policy (and target networks) as a hyperparameter, thus updating it less often;
- Action noise regularization: when calculating the targets, clipped noise is added to the action. This makes that higher values are preferred for more robust actions.

Both algorithms are used to train agents in the context of P2P markets so that there is a comparison component and a possibility of mobility between models.

3.3 Technological and Social Challenges

This section describes the technological and social implications of this dissertation. It addresses topics from data privacy, protection, and security, to ecosystem and its agents security, as well as several ethical considerations related to the application of AI algorithms in the smart grid.

3.3.1 Data Privacy and Protection

The amount of data generated and made available by devices and sensors has been increasing exponentially in recent years, especially with the emergence of IoT technology (Rahman and Asyhari, 2019). As that, protecting data privacy and complying with privacy policies is of utmost importance, especially when handling sensitive personal information. Considering the events of recent years (i.e., massive personal data leaks by several companies, to have some return, usually financial), customers of the most diverse services have some prudence in adhering to data-driven technologies (Beckett, 2017).

The European Union introduced and adopted in 2018 the General Data Protection Regulation (GDPR), a comprehensive legislation body that has had an enormous impact on how personal data is collected, stored, processed, and shared (Voigt and dem Bussche, 2017). The legal enforcement of GDPR is also transforming how digital solutions, applications, and systems handle sensitive data. For instance, the need for explicit consent for the use of data, the right to timely receive all data collected for oneself, or the right to completely delete personal data are specified in this law, forcing technical solutions to be provided following the regulations. Furthermore, with the introduction of this regulation, energy customers are also more demanding in terms of their data protection and privacy.

In the scope of the smart grid, data privacy and protection is addressed mainly in smart metering, enabled by the Advanced Metering Infrastructure (AMI), and in the network's communication layer (Ferrag et al., 2018). Given the data collected by smart meters in a smart grid context, privacy concerns become critical to the success of the initiative's aims. In the AMI, privacy extends beyond anonymity to include the invisibility of particular domestic appliances' functioning conditions (Uludag et al., 2015). However, an inevitable result of such a volume of data is the ease with which personal information can be extracted for possible abuse or misuses, such as behavioural inferences, and deduction of particular habits or activities. (Triantafyllou et al., 2020). For instance, residential appliances can be classified and monitored, in addition to allowing the study of the behaviour of their owners through the appliance load data, with non-intrusive methods. (Devlin and Hayes, 2019).

Participation in DR and P2P programs is expected to be of the most important elements of next-generation power systems (Aggarwal et al., 2021). However, active participation in these initiatives can reduce the privacy of customers, as they share private information, from which can be drawn insights from their behaviour. The most usual is the study of competitors' bidding strategies, and this information can be used in the future to harm customers, both in DR (Balli et al., 2018), and P2P (Son et al., 2020) participation.

In the ecosystem to be developed, the customers must have full control over their data, from how it is obtained to its storage. In this way, they can, for instance, select the maximum period of historic data (relative to energy consumption, generation, flexibility, etc.) that should be stored in the system (e.g., 6 months, 1 year, 3 years, etc.). However, customers should be warned that with a smaller amount of history, their agents' adaptation to the context, and the

study of their behaviour will be reduced, something that can harm their participation in services. This situation should also be addressed when participating in services where bids involving energy and/or money (e.g., P2P markets) are carried out. If the customers choose not to keep their participation history, their agents will lose the ability to learn from past experiences.

3.3.2 Agents and Data Security

The digitalization of the energy grid brings great benefits to all the stakeholders involved in it, but also creates weaknesses in the system. Using IoT devices enables real-time monitoring on a whole new level and is very convenient in smart grids, but the communications have multiple vulnerabilities that can be explored (Kimani et al., 2019). Because control and monitoring are carried out via Internet-based protocols and open-source software, the smart grid may be extremely appealing to attackers as a critical infrastructure. For instance, an attacker could assault electrical devices by altering the real-time balance of energy output and consumption through the use of falsified data generated by the appliances (Gunduz and Das, 2020).

A multi-agent system involves, by default, a lot of communication between agents, and consequently, a lot of data to circulate among different software and physical hosts. But this concept is taken to another level when the system involves mobile agents, which, in addition to transporting their data, must also know its code and execution status, when they change their host. Therefore, the security in MAS that involves mobile agents is mainly based on: (i) protecting the agent platform, (ii) protecting the mobile agents, and (iii) protecting the agent network (i.e., physical hosts from the MAS) (Bagga and Hans, 2017).

When an agent moves from one host to another, it is normal that the destination machine (i.e., its owner, or the entity that is responsible for its management,) to have some doubts about the agent's intention, as it will execute there and may cause damage to the host. Mobile agents can transport with them obscure malign codes causing harm to other agents or leading to the agent host platform's disruption (Acharya et al., 2021). To protect the host, a pre-analysis of the agent's code can be performed (Celik et al., 2019). On the other hand, also the hosts can be dangerous to the agents, and as such, constant monitoring must be carried out to ensure that the agents that are hosted in a specific machine have the best possible conditions for execution, possible with the use of a Trusted Platform Module (TPM) (Shaik, 2018). This verification must also be carried out when the host registers in the system, and in all mobility processes, creating a more robust mobility functionality since this is a problem that is aggravated by the presence of external/independent agents in the agents' ecosystem.

In the situation of service participation, in many of them, large amounts of data have to be shared between machines, with the risk of the data being used for purposes other than those initially defined, threatening both security and data privacy (Bhattarai et al., 2019). This occurs mainly because the data transitions from a completely distributed pattern to a centralized one in the service's execution. Two solutions can be explored in the development of the ecosystem:

- Data encryption: with a private key, only known to the system. In this way, even if the data is accessed, it will not be interpretable.
- Code-shipping: allows services to move their agent to the host where the participants are located, although this may not be possible for all services. This solution is viable to the customers who do not permit their data to flow to physical hosts other than theirs. Figure 10 represents the main difference between data-shipping and code-shipping, where it is possible to observe that in the code-shipping the service participants data do not flow between physical hosts.

The frameworks used to develop the agents' ecosystem (SPADE and PEAK) integrates several security features such as authentication based on username and password, and especially SSL tunnel encryption in the communication between agents.

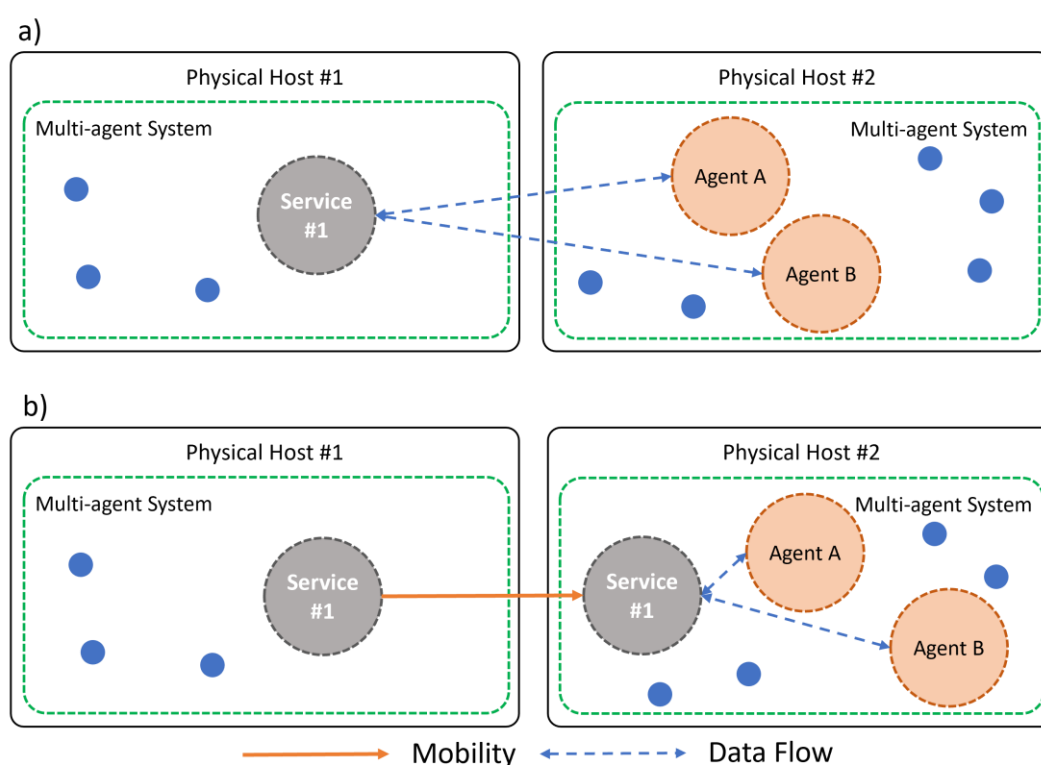


Figure 10 – Comparison between (a) data-shipping, and (b) code-shipping

3.3.3 Ethical and Social Issues

In addition to issues associated with the privacy and security of customers and their data, ethical and social issues come to have great weight in the development of AI techniques/models and, consequently, in the representation of the smart grid. In what regards AI systems, the European Commission proposed the ethical principles to build trustworthy AI (European Commission, 2019b):

- Human agency and oversight: AI systems should enable equitable societies by supporting human agency and fundamental rights, rather than limiting, decreasing or misguiding human autonomy.
- Robustness and safety: Trustworthy algorithms must be reliable, consistent, and robust enough to deal with failures and/or inconsistencies during the AI system's life cycle.
- Privacy and data governance: Citizens should have complete control over their personal information, and data related to them should not be used to harm or discriminate them.
- Transparency: It is necessary to secure the traceability of AI systems.
- Diversity, non-discrimination, and fairness: AI systems should take into account the whole range of human abilities, skills, and requirements, as well as ensure accessibility.
- Societal and environmental well-being: AI systems should be used to promote positive social transformation as well as environmental sustainability and responsibility.
- Accountability: Mechanisms for ensuring responsibility and accountability for AI systems and their consequences should be put in place.

The prominence of considering social science input when designing applications for AI systems has been noted by several authors (Sloane and Moss, 2019) (Miller, 2019). This is also extremely important in smart grid applications. One reason for this is that, even if the most advanced multi-agent systems, game-theoretic tools, and AI systems in general, are employed to automate decision-making processes considering the customers' historical profiles, people's energy practices may deviate dramatically from those predicted by these models (Robu et al., 2019). Consequently, this creates distrust on the part of the customers of these systems, and to counter this trend, customers should play a more active role in the development of models, being able, for instance, to give feedback on the actions taken.

Another point related to the automation of these systems, and of the decision processes carried out by them, is the fairness involved in these same decisions. In general, AI is being employed in power systems to make increasingly complex judgments, normally not addressed by traditional systems. For instance, when designing a smart contract in an aggregator or a local energy market, the contract will specify who gets energy first (e.g., which electric vehicle owner can charge first and who must wait (Qureshi et al., 2021)), or, in the case of excess renewable energy, which photovoltaic panels should be given priority to sell their energy or may need to reduce their generation (Moradmand et al., 2021). Therefore, the importance of inputs from multiple disciplines is highlighted by the fact that using AI systems to make decisions clearly involves not only economics but also social and ethical issues in the way systems are created and developed. One of the solutions explored is the use of incentives in the design of these systems. That is, the customers who have a higher rate of participation and understanding of the system (e.g., give up instant charging in their electric vehicle to respect the decision process), receive incentives, such as cheaper energy at home (Robu et al., 2019).

However, the fact is that many customers are not aware of the technologies that are developed to benefit them and the electricity grid itself, and as such, they are quite reticent to adhere to the suggested programs. Something normal, such as their participation, can have an impact on comfort and finances (Shakeri et al., 2020). As such, customers should be treated as stakeholders of the smart grid, and be integrated into all the initiatives that are suited to their context (Le Ray and Pinson, 2020). A big step towards this is to improve the connection between the domains of smart grids, AI, and Human-Computer Interaction (HCI) (Faheem et al., 2018). That includes, for example, designing novel ways to interact with power systems, such as intuitive interfaces (Pereira et al., 2021) and devices that enable consumers to specify their energy preferences, understand their own energy consumption patterns (Wei et al., 2018), and search for tariffs that best fit their energy context or/and needs (Vom Scheidt et al., 2019).

3.4 Chapter Conclusions

In this chapter, the materials and methods used to conceptualize and develop the proposed ecosystem are described. The chosen programming language and multi-agent frameworks are crucial topics, as all the developments are dependent on them. Modularity makes them particularly suitable for this task. In the case of Python, there are a large number of open-source libraries that can be used to improve ecosystem capabilities. On the other hand, PEAK and SPADE can be utilized to address technological and social challenges, primarily due to their integrated data privacy and security tools.

Regarding the data used in this study, it adheres to the Horizon 2020 guidelines, which state that the data should be findable, accessible, interoperable, and reusable (FAIR), as all datasets are well-described and open. This makes it easier for other developers to reproduce this work, in order to compare their work or improve what has been developed thus far.

Initiatives by the European Union and the United Nations to create secure and equitable projects have increased the significance of the technological and social challenges identified in this chapter. With the introduction of automatic software development toolkits, it is simpler to create and deploy software, typically without regard for the customers' privacy, security, or ethical concerns. All of these issues are considered in this project to enable the development of systems with a sustainable foundation.

4 Agent-based Ecosystem for Smart Grid Modelling – A4SG

As previously stated, there is a high number of MAS developed in the context of the smart grid. However, the MAS proposed in the literature are heterogeneous and mostly closed, i.e., communications are only possible inside the same MAS, with no efficient methods for interaction and communication among multiple systems and their agents. The ecosystem proposed in this work aims to intelligently manage multiple agent-based systems and all the agents that compose them. The proposed ecosystem, named Agent-based ecosystem for Smart Grids modelling (A4SG), uses a new perspective toward MAS applied to smart grids.

The A4SG can integrate, simultaneously, multiple MAS, and a MAS can contain multiple agent communities (ACOM), creating a multi-level hierarchy inside the ecosystem. In the scope of A4SG, an ACOM represents a group created in order to help with the ecosystem's management, or groups managed by an entity, e.g., an aggregator, an energy retailer, or a forecast service provider. The members of each ACOM are agents that share an objective (e.g., to participate in DR programs), that might be achieved competitively or collaboratively. Thus, in this ecosystem, an ACOM can be the representation of multiple types of groups, such as:

- Aggregation entities at the physical level, such as energy communities, or microgrids;
- Aggregation entities at the contract level, such as aggregators, retailers, or virtual power plants;
- Services forum, such as services for energy balance, DR, forecast, or resources optimization.

Each ACOM may contain further ACOMs inside, with more specific objectives that are necessarily related to the objective of the ACOM above it in the hierarchy. Each ACOM is characterized by several attributes, defined at the time of its creation, that identify it towards the ecosystem and facilitate its discovery by the agents. This characterization includes specifying the ACOM's type (i.e., one of the specified in the list above), primary objective, and

target audience, among other characteristics. Each ACOM has an ACOM main agent responsible for group management, described in the A4SG Agents subsection.

As explained before, the proposed ecosystem considers multiple organizational paradigms to coordinate and manage both the groups and their agents. For the proposed society to be viable, it must be governed by rules and norms that its agents must comply with, to guarantee the proper operation of the ecosystem. These rules can validate the mobility of agents, their branching, and their connections to other agents, among other types of validations. For instance, there is an internal MAS to the ecosystem (i.e., A4SG Representation Community (ARC) – represented in Figure 11), to allow the ecosystem's customers to register agents and host them there. However, customers can register their hosts to the A4SG ecosystem, enabling the reception of A4SG agents, on a free or paid basis. These rules and norms can be directly defined in the ecosystem, and each host/group owner or manager can define different rules.

A4SG's communication is FIPA-compliant, meaning that it is possible to use the standards provided by FIPA to standardize and add meaning to the messages sent by agents, thereby enhancing the agents' interoperability. In addition to the message's body and its content, it is also possible to include metadata in each communication. Thus, it is possible to personalize the inclusion of additional information in messages. For instance, it is possible to add ontologies and references to ontologies, so that the message recipient not only receives but also comprehends the message.

The A4SG ecosystem is currently designed and implemented considering smart grid business models, but one of its greatest benefits is its seamless adaptability to other domains. The rules and the norms of other business models can be easily implemented in the society and in each group that composes it, besides the ecosystem being able to deal with different types of agents to represent other entities with different requirements and restrictions, which can be developed from scratch by the customers and entities.

4.1 Architecture

The A4SG's architecture is one of its defining characteristics, as it supports MAS and ACOMs with a multi-level, scalable and high-performance design. The architecture consists of four layers: CECOS, Common Services, A4SG Agents, and Customer/Entity Side. Each layer has specific objectives, and the ecosystem can only be deployed and used, when all layers are combined, conceiving a robust architecture. Figure 11 represents the architecture of the A4SG, (i.e., without specifying services and aggregation entities types).

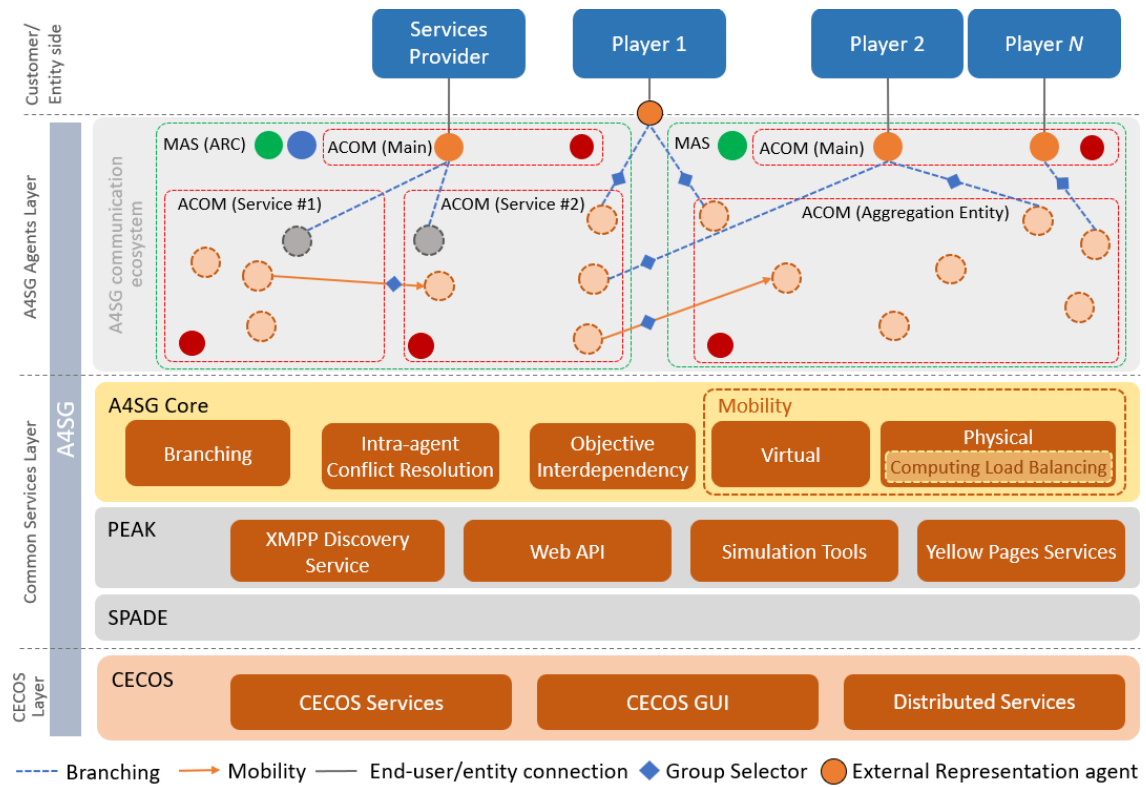


Figure 11 – Representation of the A4SG architecture

4.1.1 Citizen Energy Communities Operator System Layer

In the bottom layer of the architecture, the proposed A4SG uses the Citizen Energy Communities Operator System (CECOS) as a graphical user interface. CECOS is a control interface that was proposed and developed by the author of this work, that provides multiple services to its customers, such as tariffs management and price-elasticity calculation. The CECOS components produce the CECOS layer of A4SG.

CECOS was target of three main stages of development, until the last version was complete, in order to support decentralized systems:

- Centralized CECOS, only supporting city energy communities - a web-based platform, which data flow is represented in Figure 12, allows the community's operator to analyze customers' historical data and execute multiple services regarding the community's management, published in (Pereira et al., 2021). In this stage, the platform was divided into four main services, as shown in Figure 12. The arrows specify the normal procedure of the operator from the moment he starts using the platform until all services have been successfully executed. Besides that, the platform data flow is described, and each step is explained.

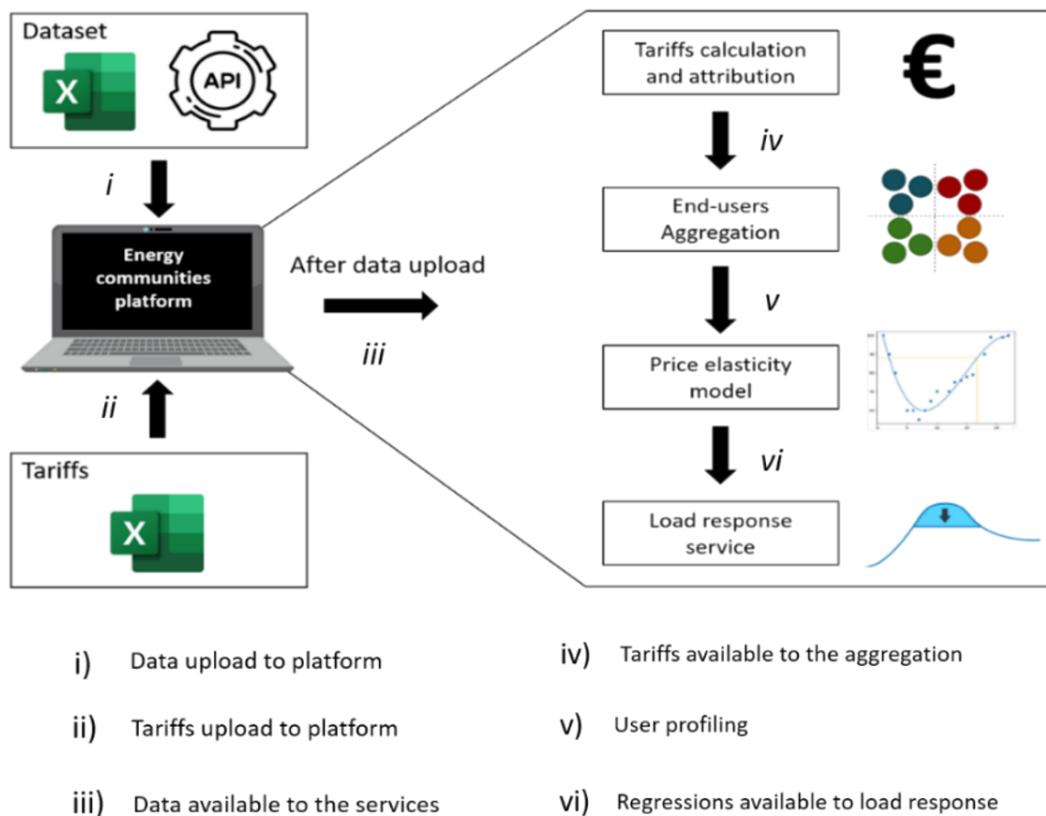


Figure 12 – Centralized CECOS data flow (Pereira et al., 2021)

- Decentralized CECOS, supporting multiple aggregation entities – as there were some limitations in the first version of CECOS, such as low scalability to other service providers and the ability to manage only citizen energy communities, in (Pereira et al., 2022c) it is proposed a significant upgrade in the concept, functionalities, and capabilities of CECOS. In the second version, CECOS no longer represents an individual and isolated energy community. Figure 13 represents the upgraded version's architecture. This upgrade allows CECOS to be used in a collaborative way where multiple instances of CECOS, one per aggregation entity, can interact, transact information, and have customers being part of more than one CECOS instance. The new upgraded version also allows the concept of service providers that can publish their services in CECOS instances. This enables the use of, for instance, energy forecasting service providers to promote and publish their services in multiple aggregation entities.
- Distributed CECOS – In the last version of CECOS, the platform can handle communication with agents through XMPP messages. In this way, it is possible to have a direct connection between customers and agents, which allows, for instance, to start simulations, define the evaluation equations for mobility, or even modify the agent's knowledge base (e.g., change the source agent data). Despite these changes, the services proposed in previous versions are still available to agents.

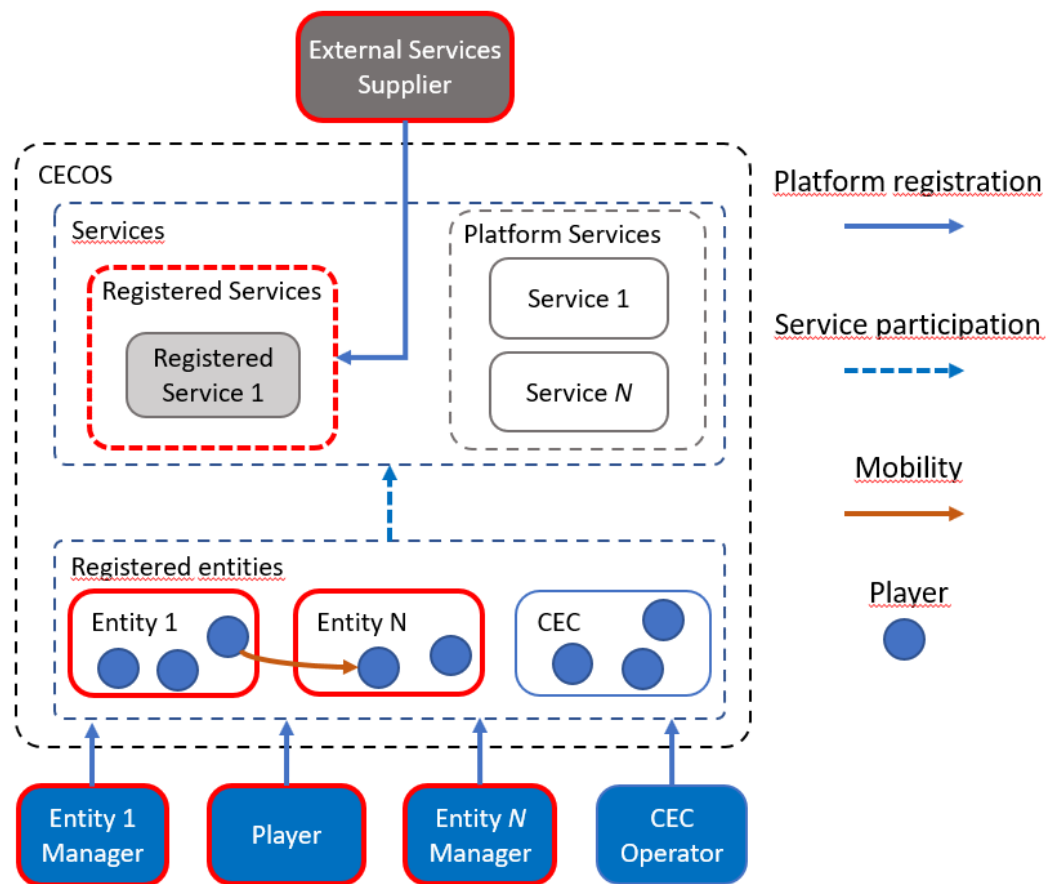


Figure 13 – Decentralized CECOS architecture (Pereira et al., 2022c)

4.1.2 Common Services Layer

The Common Services Layer is composed of the services and functionalities that are available to all the ecosystem's agents. As such its components are the SPADE and the PEAK frameworks, and the A4SG Core, the component of the ecosystem that enables access to all the proposed services, and functionalities. These services and functionalities are presented, respectively, in Section 4.3 and Section 4.4.

The A4SG ecosystem, as previously stated, is developed on top of two other frameworks, PEAK and SPADE. Therefore, the A4SG solution uses third-party XMPP servers to handle the communications and clients' groups. Starting with SPADE, it is compliant with the FIPA standards and has an agent communication channel (ACC), which is a component that manages all the communications among agents in the platform by establishing a connection through an XMPP server. This server redirects all messages from the sender to the receiver without user intervention. This framework integrates multiple security features, such as username/password-based authentication for each agent and SSL tunnel encryption in ACC. On top of SPADE, the PEAK framework provides several features for the development of robust multi-agent systems and agents. It provides a variety of XMPP capabilities to the agents that SPADE does not, including participation in Multi-User Chats (MUC) and communication through

Publisher-Subscriber (PubSub) nodes. Besides that, it allows the distributed execution of agents, unlike SPADE which runs the agents in the same process but on different threads.

4.1.3 A4SG Agents Layer

In the A4SG Agents Layer, the ecosystem accommodates all the registered physical and virtual hosts, agent-based systems, and agents. In this layer of the ecosystem is designed an open world of communication among different MAS and agents and the entire business logic of the ecosystem is represented, including the groups' hierarchy and the deployed agents.

Although the ecosystem has a well-defined and robust architecture, this layer is completely dynamic, since its structure depends on the groups and agents present in the system at a given moment. For instance, the creation and deployment of a new MAS or ACOM will affect the group's architecture and hierarchy but will have no negative impact on the ecosystem's overall management as well as on other groups.

The A4SG Communication Ecosystem is one of the components of the A4SG Agents Layers. This component is essential for ensuring that all agents can communicate with one another without the need for third-party tools to create and send messages. Multiple XMPP servers are supported, allowing for both client-to-server and server-to-server connections. This increases the number of possibilities for agents registered in A4SG, as they can contact a greater number of agents, for instance, connected to another XMPP server.

Also represented in this layer is the ARC, which is a MAS made available by the ecosystem to host the agents that compose it. This MAS represents a straightforward and convenient method for deploying the ecosystem's customers agents.

4.1.4 Customer/Entity Side

This layer represents all the users registered in A4SG, including energy customers and smart grid entities. It also represents the customers' connections to their representation agents, which can be either physical (e.g., a connection to a real energy storage system) or virtual (e.g., accessing data through an external API). Multiple communication protocols, such as Hypertext Transfer Protocol (HTTP) and Modbus Transmission Control Protocol (TCP), can be used to establish these connections.

The A4SG customers can use their own machines to host their agents as shown with the external representation agent that is deployed in the Customer/Entity side layer of Figure 11, but they can also use ARC to take advantage of its computational power and for the convenience of not having the agent executing in their hosts. Customers can also configure their machines as general hosts of A4SG, enabling the deployment of agents from other customers.

4.2 Agent Types

The A4SG agents, as previously stated, are developed based on the agents supplied by the PEAK framework, which themselves derive from the SPADE agents. Each of them has properties that are valuable to the agents based on them, as represented in Figure 14. SPADE agents create a direct connection to an XMPP server to enable communications between various agents, in addition to enabling each agent to execute asynchronously. In turn, PEAK agents offer several XMPP functionalities that SPADE does not support, such as communication through Multi-User Chat and PubSub. Furthermore, it provides a distributed execution of the agents and a direct integration with datasets/drivers to access and update the agent's data.

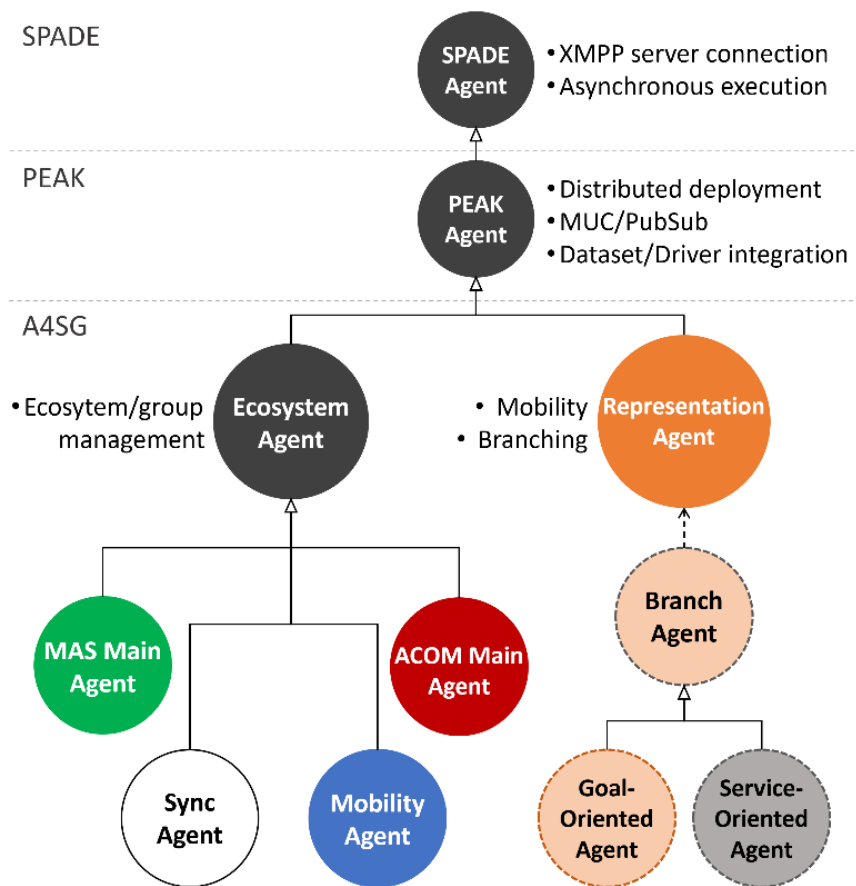


Figure 14 – A4SG agent types and a brief description

4.2.1 Ecosystem Agents

The ecosystem agents are a main component of the A4SG and are generated by the system core in response to the ecosystem's requirements. These agents are essential to the ecosystem's management as a whole and to the administration of each created group. There are different variations of this type of agent, according to their responsibilities, being: MAS main agents, ACOM main agents, synchronizer agents, and mobility agent.

The MAS main agents are responsible for managing a specific multi-agent system, and also for accessing information from the physical host where it is located. Therefore, they play a crucial role in validating mobility and branching operations, and in deploying and registering agents in the ecosystem and in the host they represent. As such, this type of agent is able to retrieve information about the host's physical machine (e.g., CPU utilization, RAM consumption, and disk storage capacity) and the environment in which it is executing (e.g., installed software, and available packages and libraries). This type of information is essential to determine whether a host is capable of executing an agent's software for the processing of the mobility and branching functionalities.

The ACOM main agents are similar to the MAS main agents, except they represent the entities that coordinate a particular group or operate as management agents. For instance, in a group with the clients of an aggregator, an agent of this type is responsible for representing the aggregator itself, whereas, in a group created for a service provider to deploy a forecast service, the ACOM main agent serves merely as the group manager.

The synchronizer agent is a type of agent developed on top of the PEAK framework's simulation tools. It is capable of running simulations in a single group using Multi-User Chat communication or in several groups concurrently using the PubSub protocol. This agent enables the configuration of many simulation parameters, including the number of periods to simulate, the granularity of the periods, and the duration of each period in real-time.

The mobility agent is responsible for managing the necessary communications for both types of mobilities available in A4SG (i.e., physical and virtual). When an agent wishes to move to a different host or aggregation entity, it sends a message to the mobility agent, which responds with a list of mobility options. Using the Group Selector component (described in Section X.X), the mobility agent must be able to analyze and evaluate the different mobility options available to an agent in a personalized manner.

4.2.2 Representation Agents

The representation agents, as the name indicates, have as their main purpose the representation of energy customers and entities in the A4SG. They can also represent individual IoT devices or energy resources and loads that want to engage in A4SG (i.e., devices and resources disassociated from a customer).

The main component of these agents, which defines them, is their knowledge base, composed of the agent's behaviours, information about the customer or entity the agent represents, the data obtained from the resources owned by the entity it represents, and the information to retrieve those data.

This type of agent has the ability to move to other physical hosts or aggregation entities (i.e., MAS and ACOMs), and to execute a branching process, creating other agents to extend and

represent itself. Therefore, the multiple representations of these agents can be in multiple physical and virtual hosts simultaneously.

These agents can be hosted outside the scope of A4SG (i.e., on hosts that are not registered in the ecosystem), thus being considered independent or external agents. However, even though they are external, they have access to all the features and functionalities of the ecosystem, and they can also take advantage of the hosts registered in A4SG with the branching functionality to improve the efficiency of the execution of their agents.

4.2.3 Branch Agents

The branch agent is a type of agent created by a representation agent to facilitate the accomplishment of its objectives. They inherit a part of their representation agent's knowledge base, considering the purpose for which they were created, and the information required to accomplish that purpose. This type of agent is dependent on its representation agent, and therefore only exists as long as the representation continues to execute; otherwise, its execution stops. The branch agents maintain a direct line of communication with their representation agent to report the current status of goal accomplishment. The procedure for generating branch agents is elaborated on in the section on the branching functionality – Section 4.4.3. In A4SG there are two types of branch agents, with different types of purposes: goal-oriented and service-oriented agents.

The goal-oriented agents are created to achieve a clear objective, defined in their creation. The goal-oriented agents must inform their representation agent of each decision they are about to make, so that the agent can validate it and verify, for instance, that there is no conflict or dependency between concurrently pursued objectives. In Figure 15 is exemplified a an energy customer representation agent and its respective branch agents, with the following objectives: (i) participation in DR programs, (ii) subscription to demand forecast services, and (ii) participation in local energy markets are examples of goal-oriented agents.

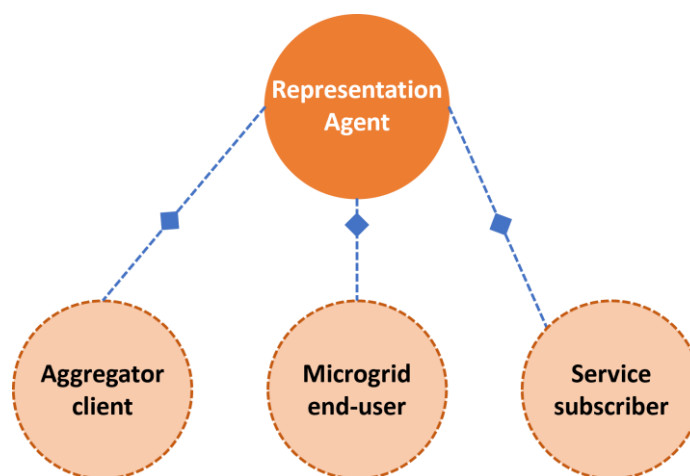


Figure 15 - Example of the representation of a customer, considering the agents' branching

To enable the virtual mobility between groups of the ecosystem, fully described in Section 4.4.2, goal-oriented must be able to evaluate themselves based on the objective they are pursuing. This evaluation can be directly implemented in the goal-oriented agent or inherited from its representation agent, considering a satisfaction criterion that, for instance, evaluates customer's comfort and energy costs.

The service-oriented agents, on the other hand, provide services to other agents. These services may be public or private, and free or paid. The agent's service can be embedded within its code or in an external API; in this case, the service-oriented agent acts as a gateway agent. These agents may be created, for example, to provide forecasting, optimization, and tariff management services. In contrast to goal-oriented agents, service-oriented agents are not required to evaluate themselves; however, they are required to provide information about the service they provide so that goal-oriented agents can evaluate them.

4.2.4 Agents Characteristics in A4SG

The agents developed for A4SG, despite being able to be further developed by other customers and entities that wish to participate in the ecosystem, already integrate the necessary characteristics and functionalities to be considered an agent. Table 7 presents the agents' characteristics proposed in (Weiss, 1999), and their integration into A4SG agents.

Table 7 – Integration of agents' characteristics into A4SG agents

Characteristic	Brief Description	Implementation in A4SG agents
Agility	Analysing and identifying new opportunities	The representation agents can create branch agents for new objectives when the agents themselves or the entities they represent identify a new objective to achieve
Autonomy	Ability to control their actions	All A4SG agents are autonomous since they can make their own decisions without any interaction from their owners
Character	Present a certain personality	The agents can integrate personalized methods of self-evaluation, as well as execute custom behaviours
Flexibility	If your tasks do not need to be predetermined	Due to their perception of the environment, agents are able to adapt and seek new objectives or make decisions that are not directly programmed
Intelligence	Linking the characteristics of learning and autonomy	All A4SG agents are capable of adopting autonomous behaviours, and learning can be integrated via personalized evaluations or reinforcement learning
Learning	Gaining experience with the result of actions and adapting decisions futures	The agents can either learn and evaluate their actions to consider, for instance, mobility, or learn to optimize their participation in services through reinforcement learning
Mobility	Ability to move between machines depending on the perception of the environment	The proposed mobility functionalities allow agents to move to different physical hosts or to other groups within the same physical host

Characteristic	Brief Description	Implementation in A4SG agents
Persistence	How much an agent exists over long periods	Branch agents can stop their execution if their objective is not continuous and has been accomplished. However, representation agents are completely persistent, and their execution can be stopped only if the entity they represent commands it
Pro-Activity	The ability to aim for a goal	In A4SG all the agents have well-defined objectives, working proactively to achieve them
Reactivity	Ability to react and adapt to changes in the environment	The agents are able to adapt and react based on the data they obtain from their environment (for instance, via sensors)
Sensory Capacity	Sensitivity to the perception of the environment in which it operates	The agents can have direct connections to real sensors or read their data through an API
Social Ability	Ability to communicate with other agents in the system	A4SG follows the society's paradigms, where each agent can communicate and share knowledge with the other agents of the ecosystem

4.3 Tools and Services for A4SG

The tools and services presented in this section of the dissertation were developed by the author within the scope of this work, however, they contribute primarily to facilitate the general operation of the ecosystem.

4.3.1 Static Code Analysis

Static code analysis is a technique for debugging computer programs that examine the code without executing the program. The procedure helps guarantee that the code complies with industry standards by providing knowledge of the code's structure. In the context of A4SG, it is used every time an agent is ready to be executed and when the code of an agent is transferred between different machines.

This service is essential to ensure the operational status of the ecosystem and its agents, and, as such, it has two main objectives:

- Prevent compilation errors – Compilation errors are becoming less frequent as more and more programming assistance is introduced into development environments, but in fact, a minor mistake can lead to critical errors that prevent the software from executing. Therefore, Pylint was used to ensure that the code is executable and functional. This library can recognize syntax and semantic errors, as well as customized errors, such as ecosystem internal rules.
- Find security vulnerabilities – In the smart grid, due to the increasing dependency on digitalization and IoT, various security incidents such as unauthorized access, malware

attack, and data breach, have grown at an exponential rate in recent years. As a result, it is becoming increasingly vital to address security concerns, and while the XMPP server used in the ecosystem implements security mechanisms, the agents' source code is entirely open for development. Using the Pysa library, it is determined whether or not the developed Python code employs certain internal frameworks that are designed to prohibit access to an agent's data based on technical privacy policies. Pysa is also very useful in the detection of XSS and SQL injection.

In A4SG the static code analysis is an automated process so that the ecosystem maintains a functional operation, and its security is reinforced, therefore allowing A4SG to be more robust and fault tolerant. Figure 16 presents the flowchart of this service.

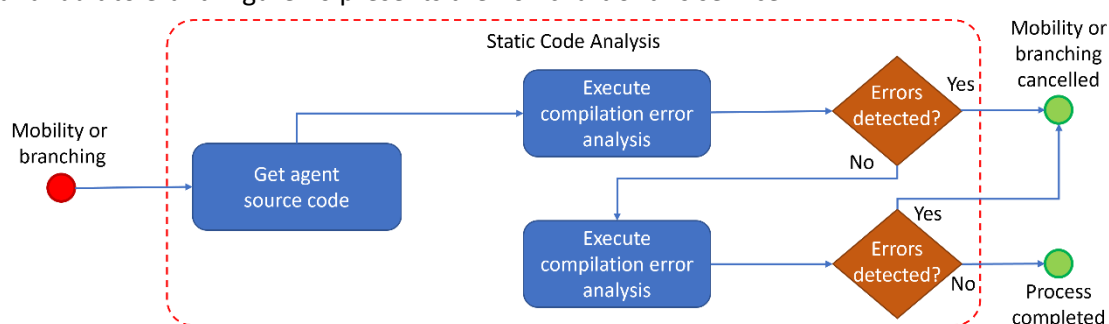


Figure 16 – Static Code Analysis flowchart

4.3.2 Translator Interface for Different Communication Protocols

Communication is a crucial aspect of agent-based systems because it enables agents to interact and share information. Thus, it is essential to increase the means by which agents can communicate, with the focus here being on communication protocols. As previously explained, the primary communication protocol of A4SG is XMPP, but others can be crucial, such as when communicating with physical devices. Thus, gateways for MQTT and HTTP communication have been integrated into the ecosystem.

This service behaves differently from the others proposed in this section. This because, this service is represented by agents in the ecosystem as if it were a service provided by an external service provider. When an XMPP server is added to the ecosystem, a service-oriented agent is created to represent this service on that server, functioning as a gateway agent to enable communication in other protocols to the agents registered in that server. This is done to prevent communication failures between servers and ensure the emission and reception of all messages.

To access this service, agents must communicate with the gateway agent defined to the communication protocol that they have access to, which in this case are MQTT and HTTP. These gateway agents will be subscribed to the MQTT topic with their respective names and provide an HTTP API, enabling the reception of MQTT and HTTP messages. Several elements of information are required for these agents to perform the necessary translations, depending on the communication protocol being used. For example, MQTT requires the topics where to

publish the message, HTTP requires the URL to contact the agent, and XMPP requires the agent's name and the server on which it is registered. The translations can be done for each combination of protocols, in any direction, creating a greater range of options: (i) XMPP-MQTT, (ii) MQTT-XMPP, (iii) XMPP-HTTP, (iv) HTTP-XMPP, (v) MQTT-HTTP, and (vi) HTTP-MQTT.

This service represents a scalable method for increasing ecosystem interoperability. Three communication protocols have been integrated so far, but others can be easily added to expand the range of communication options.

4.3.3 Group Selector

The necessity for mobile agents in A4SG stems from two key motivations: (i) the construction of a robust ecosystem in which agents can adapt to their context, and (ii) giving energy customers the ability to seek better solutions to optimize the use of their energy resources. These mobility mechanisms are represented in the ecosystem by a centralized agent (i.e., mobility agent), which is unique inside the ecosystem, in order to guarantee that it knows the existence of each different entity registered in A4SG, and how to contact them. The mobility agent has access to the Group Selector tool.

The Group Selector, represented by blue squares in Figure 11 and Figure 15, is a tool provided by A4SG with a high preponderance in the proposed mobility and branching functionalities. Its primary objective is to evaluate the available groups for an agent to move, either at the time of the agent creation (i.e., branching) or when the agent requests mobility. These groups may be physical hosts, MAS, or ACOMs.

Regarding the physical hosts evaluation, the priority is to ensure that the machine provides the necessary software for the agent to execute properly. Therefore, a machine that is not able to execute an agent, is automatically excluded from the mobility options. Other criteria that consider information about the machine, such as CPU usage, free RAM, and free disc memory, can be considered depending on the objective of the mobility.

In what regards the evaluation of MAS, ACOMs, and their respective services, the evaluation parameters are more dynamic, since they vary based on the agent's objectives and the type of aggregating entity under consideration. Moreover, the customers of the A4SG may choose to evaluate objectives not initially addressed during the development of the ecosystem, making this evaluation more difficult and complex than the evaluation of physical hosts. As a result, in those situations, a function or equation must be submitted to the Group Selector. Some services may have embedded evaluation tools, since, normally, the used metrics do not differ much. For instance, in the case of the forecasts, the customer can pass a customized function or equation, or choose one of the available metrics: mean absolute error, mean absolute percentage error, root mean squared error, or coefficient of determination

4.4 Novel Functionalities for Multi-Agent Systems

In this section, all the functionalities proposed in this work, which constitute a significant part of the contributions and innovation it proposes, are described in depth. These functionalities have as their main objective to integrate intelligence in the agents, as well as in the ecosystem itself. The proposed functionalities are summarized and described in Table 8.

Table 8 – Overview of proposed functionalities

Functionality	Objective	Usage Benefits
Physical Mobility and Computing Load Balancing	Allowing agents to move between physical hosts considering the convenience of their host, and the balance between hosts in what regards to computing load	Enable customers to deploy agents in evenly loaded physical hosts
Virtual Mobility	Allowing agents to move between agent-based systems (i.e., multi-agent systems and agent communities) in a personalized way	Addresses the dynamic interactions present in the smart grid, where the customers can find the best services for their profiles
Branching	Allowing representation agents to create branch agents to achieve new objectives in a distributed way	Enables agents to not overload their selves in terms of data and objectives
Objective Interdependency	Creating direct communication channels between dependent and interdependent ways	Less communication in the XMPP server, as the representation agents lose the responsibility of facilitator between branch agents
Intra-agent Conflict Resolution	Resolving the possible conflicts between multiple branch agents that represent the same entity	Manage multiple objectives from the same entity to comply with contracts, maximize customer's comfort and minimize energy costs

4.4.1 Physical Mobility and Computing Load Balancing

The proposed ecosystem allows the integration of several physical hosts (e.g., servers, computers) as well as virtual hosts (e.g., virtual machines). For this same reason, it is important to consider and address the distribution of agents by the available hosts, both for the convenience of the entities represented by the agents and for the cause of balancing the computational load between the hosts. This ability in the ecosystem is made possible by the developed physical mobility.

The physical mobility to a different host, such as a different server or computer, allows an agent to stop its execution in a specific host in order to be deployed in another host, without changing its knowledge base. This mobility is executed through a weak mobility, that saves and restores only the runtime values of the agent's properties, unlike strong mobility that also saves data from the agent's runtime state, for instance, the execution stack (Mitrovic et al., 2011). This type of mobility is either requested by the agent (i.e., representing its entity in the request),

which selects a new host as the destination, or by the ecosystem's computing load balancing functionality.

If the mobility is requested by the entity that the agent represents, for instance from the CECOS platform, the process is relatively straightforward. In this situation, the entity itself can select the new host, and if the process is validated, the agent's execution is then moved to the new host. This type of mobility can be convenient for the A4SG's customers to execute their agents on hosts other than their personal computer or server, as the agent's continuous operation becomes safer.

On the other hand, physical mobility can be triggered by the Computing Load Balancing functionality. In related works, this functionality is normally referred to as a load balancing module, and it enables the distribution of the computing load throughout the ecosystem's hosts. This distribution considers multiple attributes of the physical hosts, and these are considered in the following order:

1. RAM usage;
2. CPU usage;
3. Million Instruction Per Second (MIPS);
4. Free disc memory.

The Computing Load Balancing functionality uses the above-mentioned parameters to distribute the workload equally on available resources (i.e., in the context of A4SG – physical hosts). The functionality iterates through the parameters sequentially, in the defined order, and selects the host with the best value from the considered parameter to execute the agent. If two or more hosts tie in a given parameter, the subsequent parameter is used to determine the agent's destination. If all the parameters have equal values, one of the hosts is chosen randomly. In order to make the evaluation more objective, the values are not assessed directly; rather, they are fuzzified into an interval of integer values, from 1 to 10, before being used. This is done to ensure that minimal alterations in parameter values do not have a significant impact on the distribution of agents. In these situations, it is a responsibility of the MAS Main agent that is representing a particular physical host to obtain these parameters from the machine. Therefore, this type of agent is implemented in order to be able to provide this type of data when other agents are requesting physical mobility.

The main objective of this functionality is to provide continuous service to the agents, especially in case of failure of any physical host by provisioning and deprovisioning the application instances along with proper utilization of resources. In addition, load balancing aims to minimize the response time for tasks and improve resource utilization, which enhances system performance at a lower cost. In this way, it is ensured that no machines are overloaded, and therefore, the agents can execute in optimal environments considering their context. In Figure 17 is exemplified the use of the Computing Load Balancing functionality, where it is possible to observe agents exiting overloaded machines.

Agents and the entities they represent can benefit from this type of mobility, especially when employed in a real-world context. The physical mobility can help entities that own agents to save money, for example, by enabling them to move agents to a host with lower hosting fees or energy costs. Moreover, it provides dynamic access to profitable and reliable services, as well as the capacity to relocate agents to the most cost-effective host, such as one that maximizes the use of renewable energy, so enabling the practice of green computing (Saha, 2014).

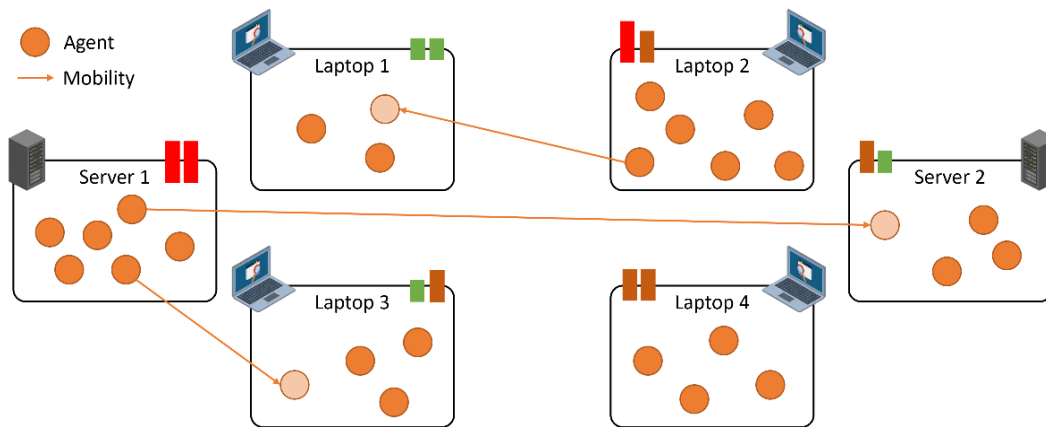


Figure 17 – Exemplification of the Computing Load Balancing functionality

4.4.2 Virtual Mobility

The architecture of the A4SG Agents Layer is mostly composed of groups, these being MAS or ACOMs, taking into account their level in the hierarchy, and the specificity of their objectives. These groups can represent various smart grid entities, such as aggregators, energy retailers, energy communities, or even microgrids. With the evolution of the smart grid, these entities have become increasingly important, and with more advantages to provide to energy customers. As such, some of these entities stand out for having more benefits, or for adapting better to some types of customers than to others, introducing the concept of competitiveness in these groups. A significant part of these groups' interactions is based on a provider-client model. This model is utilized, for instance, by aggregators, energy retailers, and distribution systems operators (DSO). And while some of these interactions are static, as, for instance, it will be challenging for a customer to change the DSO, the most frequent and complex interactions are dynamic. These dynamic interactions allow customers and entities to navigate the system by signing contracts with various entities over time. The dynamic characterization of the smart grid increases its modelling complexity but also supports the usage of MAS, and the proposed virtual mobility in A4SG, to model this type of system.

The Virtual Mobility functionality allows A4SG agents to move to other MAS or ACOMs on the same physical host. However, it can be combined with the physical mobility to allow mobility to a group present in another physical host. The virtual mobility, not deeply explored in the literature, in the context of the smart grid, enables energy customers to search in a personalized manner for the entities that best match their profile. Depending on the type of aggregation

entity or service being evaluated, A4SG agents and the entities they represent can have customized evaluation functions (i.e., functions developed by the customers). This aligns this methodology with the most current European Union strategies, which emphasize customer empowerment, i.e., giving the customers authority in the decision-making. Despite that, there can be a greater abstraction in this evaluation, as it can be performed through the Group Selector service, which allows using default functions to evaluate the agents' participation. Figure 18 represents the virtual mobility in the context of A4SG.

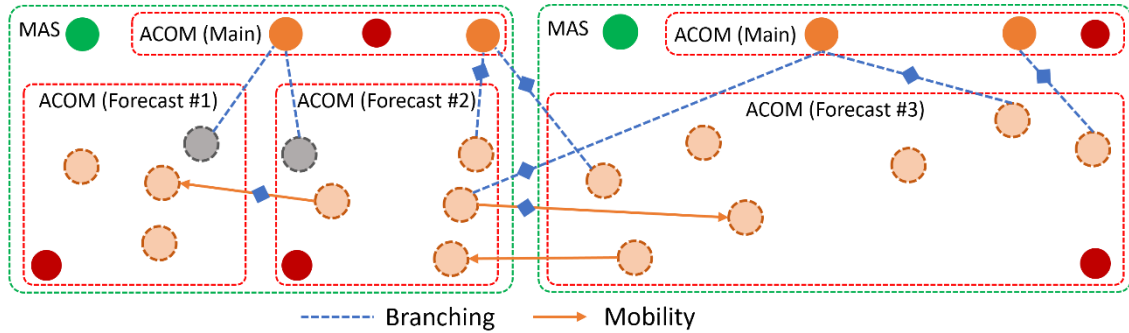


Figure 18 – Exemplification of the Virtual Mobility functionality

The evaluation that triggers virtual mobility is continuous and occurs at intervals determined by the agent. This is because, depending on the aggregation entity or service being evaluated, it may make sense to analyze, for instance, on a weekly or monthly basis. In the case of aggregators, a weekly assessment may be appropriate, whereas a retailer may benefit more from a monthly evaluation in order to have a monthly cost overview. In other circumstances, such as resource optimization or even demand forecast services, a daily evaluation may be appropriate.

The options to be evaluated by each agent must be related to the agent's objective, and therefore, the agent communicates with the mobility agent to make a request, taking into account its objective. The mobility agent, in turn, processes the request and its data, returning the options available in the ecosystem. As the ecosystem evolves and more aggregation entities and services are registered in A4SG, this process becomes less efficient because more evaluations will be required, since this process follows a brute-force approach, where all the options are evaluated. Therefore, it is required to develop a method for optimizing this procedure. The solution found is filtering the mobility alternatives before evaluating each one.

To support the filtering of the mobility options in a smart grid context, it was used the clustering technique, which allows the identification of similar data, splitting it into a finite number of groups. In this process, the ecosystem's customers will be grouped in order to determine which ones are similar based on the various parameters of their knowledge base. The division into clusters is determined by the parameters chosen, which in turn are determined by the mobility objective. For instance, it makes sense to consider flexibility when moving between aggregators, whereas it makes more sense to consider only consumption and generation profiles when moving between forecast services. This procedure, with the flowchart represented in Figure 19,

involves multiple steps, including dimensionality reduction, support algorithms, clustering, results visualization, and evaluation of the filtered mobility options.

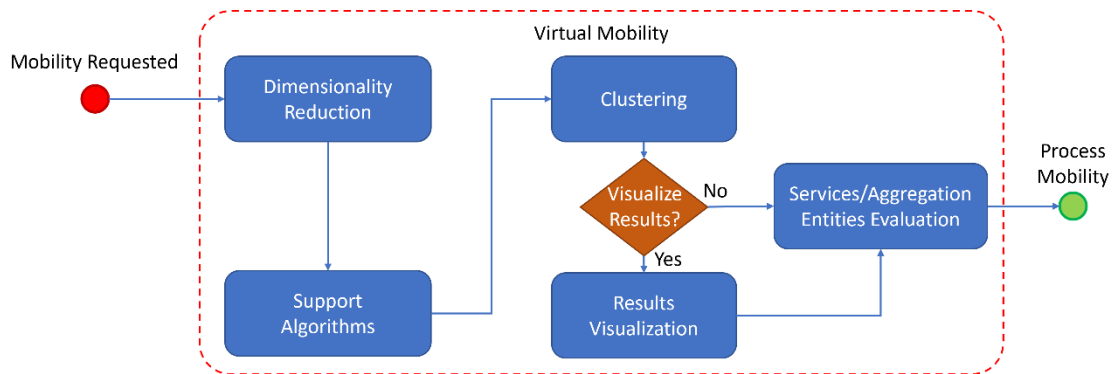


Figure 19 – Virtual Mobility functionality - options filtering flowchart

The agents may contain a great amount of data that is required for mobility. However, clustering does not perform well when there are numerous distinct parameters. Consequently, the first step of the model, dimensionality reduction, which applies the Principal Component Analysis (PCA) algorithm, is used to transform the data to a lower dimension. It employs the evaluation through the Davies-Bouldin Index (DBI) to find the best number of data dimensions to enhance the clustering results, where the best number of dimensions is the n-dimension with the lowest DBI score. DBI requires the list of n-dimensional points contained within the dataset, as well as the classification assigned to each point by the clustering algorithm used to calculate the evaluation between these parameters.

The second step is the application of the support algorithms, which in the context of this work are the Elbow and Silhouette algorithms, which are used to determine the optimal number of data clusters. This is a crucial component in this context, as it will determine which customers in the ecosystem are similar, thereby triggering the evaluation of services or aggregation entities in which comparable customers participate. In the next step, the K-Means algorithm is applied to the reduced data, using the results from the support algorithms to obtain the final clustering results.

The fourth step is optional, as it is not required for the completion of the process; it involves the visualization and analysis of the results. It enables the visualization of clusters in a scatter plot, where the points are plotted based on the reduced dimensions of the data or the original parameters from the agents. The visualization is compatible with both two and three dimensions. This visualization has no direct effect on the results of virtual mobility, but customers can analyze it to determine why certain services or aggregation entities were selected over others, and even manually select a different mobility option.

After the mobility options have been filtered, the process advances to the final step, where the filtered options are evaluated based on the evaluation equation of function from the agents to the objective in question. To process virtual mobility, the option with the highest evaluation score is selected, and if the process is validated, the agents are moved to their new group.

4.4.3 Branching

The concept of agent and objectives is directly connected, especially since in several agent definitions it is said that an agent "interacts through sensors and actuators, trying to accomplish its design objectives" (Wooldridge and Jennings, 1995). Therefore, it is typical for agents and MAS to have objectives that are pursued individually or in group. In the context of smart grids, however, as the number of opportunities to be addressed increases, agents may become overloaded, and therefore the knowledge and processing ability required to accomplish the same objectives may become unsustainable for a single agent.

To address this problem, in A4SG it is proposed a novel functionality, entitled branching. This functionality enables the creation of branch agents to handle new objectives from the representation agent, hence preventing an agent from becoming overloaded with data and objectives. This functionality employs a similar approach to agent cloning techniques, such as the ones proposed in (Shehory et al., 1998) and (Rosaci and Sarné, 2013). However, in contrast to the agent cloning strategies, which prioritize resource optimization and service provision, the branching functionality permits branch agents to pursue diverse objectives in the name of their representation agent. Figure 20 presents the branching functionality flowchart.

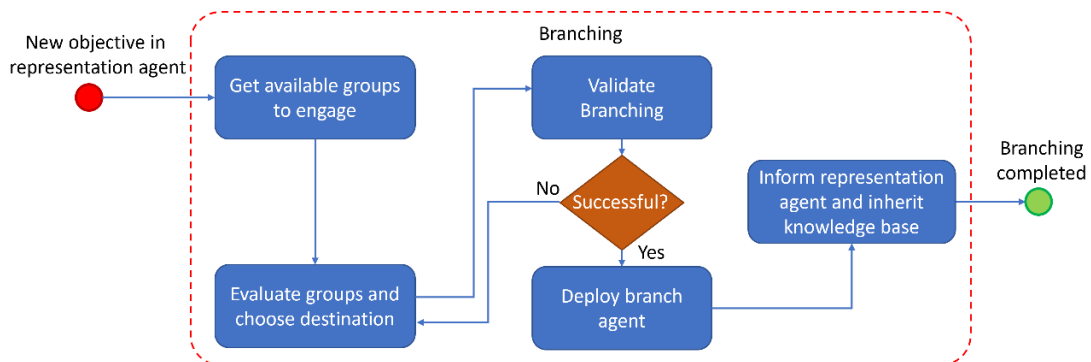


Figure 20 – Branching functionality flowchart

The process of branching begins, if required, by gathering and evaluating all accessible groups to facilitate the agent to achieve its objective. The optimal group is selected, and if the branching is validated, the agent is deployed into the new group. Alternatively, if the branching is not validated, for instance, if the group does not accept more agents, the next best evaluated group is considered. In the last step, the branch agent notifies its representation agent that it is now executing and requests a part of its knowledge base.

With the use of the branching functionality, a representation agent becomes a distributed entity, where each of its branch agents represents it for the accomplishment of a single objective. In this way, by separating the objectives by different agents, the capacity to achieve a single objective and the collection of objectives of a representation agent is increased.

Therefore, in the context of the smart grid, the branching functionality enables agents to participate in multiple aggregation entities, or even subscribe to several different services,

simultaneously. There are two types of branch agents: goal-oriented, and service-oriented, as described in Section 4.2.3. In Figure 15, as previously presented, are exemplified possible branch agents of an energy customer's representation agent. On the other hand, a service provider representation agent can create branch agents to provide multiple services, as presented in Figure 21.

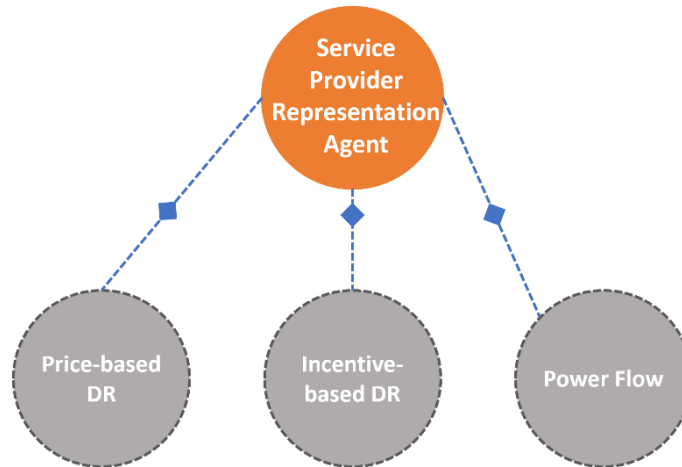


Figure 21 – Example of the representation of a service provider in A4SG, considering the agents' branching functionality

Using the Group Selector service, the branching functionality identifies the optimal aggregation entity or service for an agent, given its context, restrictions, and requirements. Therefore, the branching functionality utilizes both the proposed mobility functionalities (i.e., Computing Load Balancing and Virtual) to determine where to deploy the agent, at both the physical and virtual levels. A significant benefit of integrating the branching and mobility functionalities is that a single agent can simultaneously act in several locations. Although it must be physically hosted on a computer or a server, it can have multiple virtual representations on multiple hosts. In addition, these functionalities enable agents to continuously seek solutions that are better than their current ones, so supporting the dynamic relationships of the smart grid, and an organic flow between groups in the ecosystem.

One advantage of the branching functionality is the locational flexibility in terms of the physical resources where service participation takes place. As previously mentioned, some customers may not want to move their agents to other hosts for data privacy and security concerns, necessitating the branching of services in which they want to participate to the host where their agents are running. In this manner, there is a form of code-shipping in which the original service's code is duplicated on the agent's host. Depending on the type of service and the service provider's definitions, this process may incur fees for customers.

Although this functionality brings significant benefits to the agents and to their participation in the smart grid, it actually introduces potential difficulties in the management of the agents, particularly in their decision-making process. The existence of different objectives, and the issue of their management being carried out by different agents, make the agents' decision-making

process significantly more complex, requiring consideration of additional types of parameters and data that a single branch agent lacks the information to resolve. As a result, two new functionalities were developed to address objectives' dependencies and potential conflicts.

4.4.4 Objective Interdependency

The deployment of a high number of branch agents will assist an entity in achieving all of its objectives more efficiently, with a distributed architecture, that is also personalized for each agent and entity. The reality, however, is that when distributing the decision-making process and assigning an agent to each objective, a branch agent may not have access to all of the information required to achieve that objective. For instance, if there is a branch agent for participation in DR and another for subscribing to forecast services, the agent for participation in DR will require the data obtained by the forecast, and it is not guaranteed that this information can be obtained intuitively and without an exponential number of communications. Figure 22 represents the direct communication channels in the ecosystem before the implementation of the Objective Interdependency functionality.

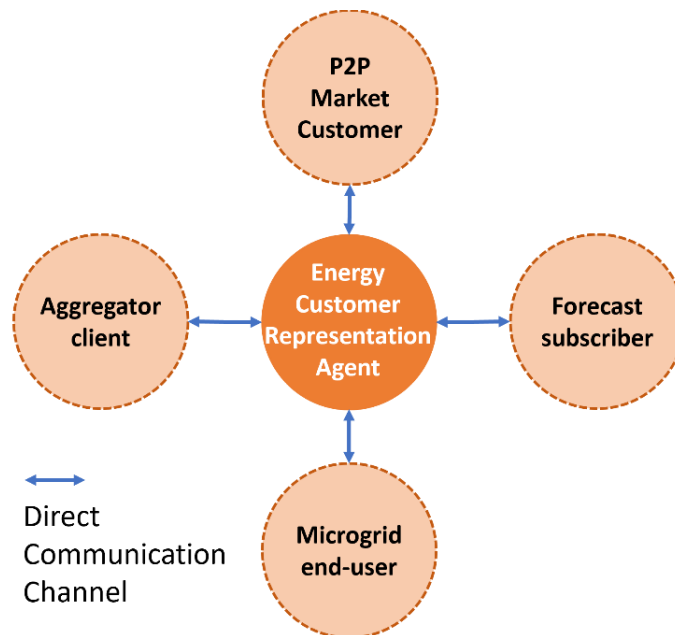


Figure 22 – Communication between branch agents before the implementation of the Objective Interdependency functionality

The objective of the Objective Interdependency functionality is to facilitate the achievement of objectives by facilitating communication between branch agents who can assist each other in either direction. This functionality employs two functionalities for this purpose:

- Create/deploy new branch agents
- Create direct communication channels between agents

When an agent is created (with a focus on branch agents), this functionality verifies the type of data the agent requires to accomplish its objective. If it is only data that would normally be contained in the knowledge base of a representation agent (e.g., consumption profile, generation profile, flexibility, etc.), the functionality does not have any intervention. However, if it requires additional data, such as forecasts or resource optimization, the functionality checks to see if other agents are pursuing those objectives. If there are insufficient agents to provide the required data, the Objective Interdependency functionality will create a new branch agent. However, if an agent has already been created to fulfil the necessary objective or provide the required data this functionality establishes a direct communication channel between the two agents. Thus, there is no longer a need for an intermediary in the communication. Figure 23 represents the direct communication channels in the ecosystem after the implementation of the Objective Interdependency functionality.

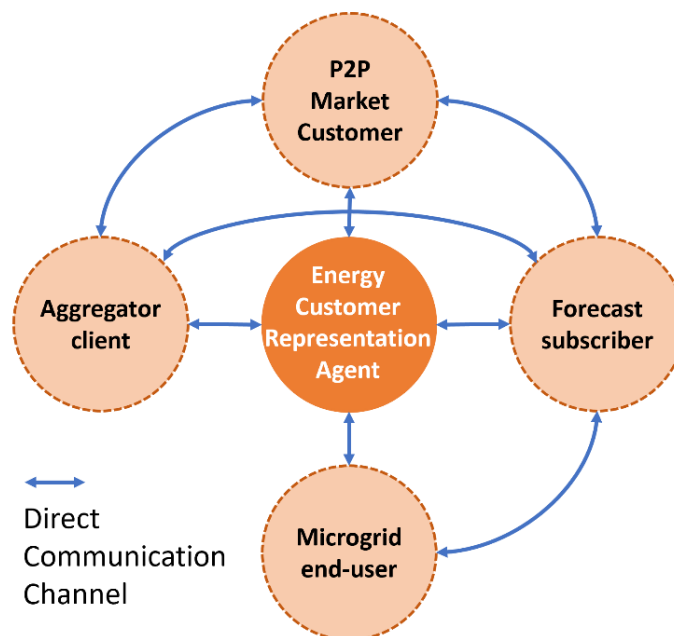


Figure 23 – Communication between branch agents after the implementation of the Objective Interdependency functionality

With the application of this functionality, it is ensured that an agent has the data and services required to achieve its objective in the most efficient manner and with the fewest number of messages to other agents. In this way, the representation agent has the greatest responsibility to manage its entity as a whole, obtaining data from sensors and APIs that make its data accessible, losing its role as a facilitator between branch agents that are dependent (dependence in only one direction) or interdependent (dependence in both directions).

4.4.5 Intra-agent Conflict Resolution

With the diversification of service offerings on the smart grid, energy customers have a greater chance of benefiting from the usage of their energy resources. However, just as there can be

dependent and interdependent objectives, there can also be conflicting objectives. These conflicts may arise for different reasons, namely:

- Contractual situations;
- Usage of the same physical resources;
- Usage of the same information or potentially impact the same parameters.

Consequently, the existence of these conflicts must be resolved so that the collection of branch agents of a representation agent understands what decisions to make in these circumstances. In this way, a method for resolving conflicts on the smart grid is proposed, based on communication between agents and the definition of priorities, represented in the methodology as the priority stack. Figure 24 shows the conflict resolution methodology flowchart, which incorporates multiple steps crucial for its application.

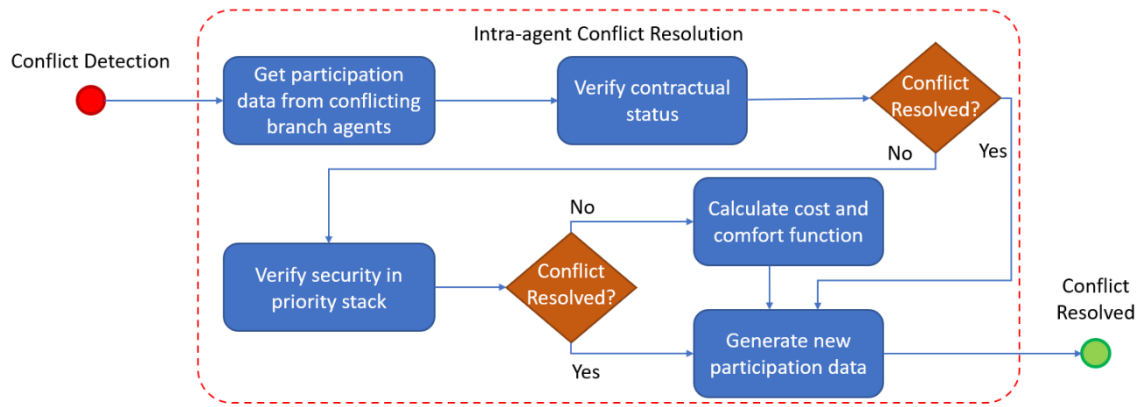


Figure 24 – Intra-agent Conflict Resolution functionality flowchart

Firstly, it is necessary to detect the conflict between agents, and this is done through triggers in the representation agents. The triggers verify if the branch agents may have a conflict in what regards one of the topics of the list above mentioned, and in positive situations, the functionality is triggered. In the following step, the conflicted agents must share their data to start the process. The first verification stage relates to the contracts of the consumers, as the customer's contracts may contain a clause that obligates the customer to participate in some cases. If such a clause exists, it would take precedence in resolving the conflict as the determining factor, and the conflict would be resolved. If the conflict cannot be resolved using simply the contract terms, the functionality proceeds to the following verification phases, where the customer-defined priority stack is applied.

The methodology's priority stack consists of three levels, of which two are defined by the customer represented by the agent in question. The first one, which is mandatory, relates to security. In this case, security refers to both the energy network as a whole, as different requirements must be met to keep the grid operational, and the agent's physical devices (i.e., IoT devices, such as air conditioners, washing machines, and sensors). The second and third levels, in the order determined by the customer, are the level of comfort and the associated

costs. The order is optional because the preferences can vary depending on the entity or customer being represented. Even the same entity or customer may have different preferences depending on their context, considering different factors, such as the time of day. To specify their preferences, the customers and entities have to establish an equation or function to evaluate the various comfort and cost parameters. This customization by the customers enables their agents to make autonomous decisions based on the customer's preferences.

After this step, the conflict is resolved, and as so, the final step is realized, in which the new participation data in the services is generated according to the results obtained by this functionality and is communicated to the respective branch agents.

4.5 Chapter Conclusions

In this chapter, is described the primary contribution of this work, the A4SG. This solution allows multiple multi-agent systems and agent communities to coexist in order to model smart grid entities as individuals and the smart grid as a whole. It has a robust four-layered architecture with clearly defined responsibilities for each layer and its components or services.

In addition to the proposed services and functionalities, the proposed types of agents have distinct responsibilities and are entirely adaptable by entities that wish to employ them. Customization is one of the primary innovations of this solution, as all of the proposed agents and functionalities can be used to increase the quality of the active participation in the smart grid, considering customizable methods for evaluating agents and services. Besides that, both the programming language (Python) and the multi-agent systems development framework (PEAK) used to develop the proposed ecosystem are modular and extensible, allowing the addition of new features that can improve the ecosystem operation.

5 Service-based Agents for Smart Grid Business Models

In this chapter are described the proposed smart grid services integrated into the agent-based ecosystem as service-based agents. With the exception of the integrated demand forecast services, all other services were proposed and developed by the author of this dissertation.

5.1 Energy Demand Forecast

Demand and supply forecasting is a crucial part of the strategic planning of the entire power and energy systems, as the improvement of the forecasts' accuracy can reduce uncertainty and lead to better operational decisions. A significant part of the services and programs offered by the smart grid depend on forecasts because they are planned on an hour-ahead or day-ahead basis, and therefore there are no certain values of the customers' demand.

As energy demand forecast is one of the most addressed domains in the smart grid, with a large number of developed and deployed works, it was not the objective of this study to propose new forecast models. As a result, some state-of-the-art forecast models were incorporated into the ecosystem. There were used forecasts with various technological capacities, processing requirements, and training errors, including the following:

- Baseline – mathematical model, published in (Gomes and Vale, 2022);
- Support Vector Machines – machine learning model, based on the Support Vector Machines algorithm, published in (Gomes and Vale, 2022);
- XGBoost – machine learning model, based on the ensemble algorithm, named XGBoost published in (Gomes and Vale, 2022);
- Neural Network – deep learning model, based on the artificial neural networks algorithm, published in (Ramos et al., 2021)

5.2 Electric Network Analysis

The power flow research in electricity networks is regarded as one of the most frequent and generally addressed issues in power systems studies (Baczynska and Niewiadomski, 2020), as it is a required process in the analysis of the energy grid performance (Montoya Giraldo, 2019). Power flow calculation can be considered a tool that is implemented to obtain, via an iterative methodology, all the voltage final values of an electrical network with a determinate tolerance acceptance to obtain the full operative status (Baczynska and Niewiadomski, 2020).

The proposed power flow model analyzes multiple grid attributes, including energy load, energy supply, energy losses, and energy generation, to detect problems in the grid. Each of these characteristics is broken down into three distinct categories: active, reactive, and apparent. Pandapower is the library used to construct the energy network and manage the power flow. It was used a balanced AC power flow, based on the backward/forward sweep algorithm.

The power flow algorithm is executed with the energy customers' data and outputs the information regarding the network in each period. To maintain the grid in an optimal operation state, two types of issues can be detected:

- Active load above the grid's transformer threshold (i.e., limit to reliable operation).
- Unbalance between supply and demand in the grid.

The verification performed to ensure that the active load values are within the limits of the transformer is described as:

$$P_{Load_p} \leq Transformer_{Limit} \quad (2)$$

where P_{Load_p} is the active load in period p , in kW, and $Transformer_{Limit}$ is the grid's transformer operation limit, in kW. The transformer limit is extracted from the network's data. The community operator must address the periods in which this condition does not apply.

The supply-demand balance is defined by the equality between supply (i.e., energy supply coming from the main grid and energy generation) and demand (i.e., energy load and energy losses). The constraints that verify the balance of the grid in a specific period, in the active, reactive, and apparent components, respectively, are as follows:

$$P_{Load} + P_{Losses} = P_{Supply} + P_{Generation} \quad (3)$$

$$Q_{Load} + Q_{Losses} = Q_{Supply} + Q_{Generation} \quad (4)$$

$$S_{Load} + S_{Losses} = S_{Supply} + S_{Generation} \quad (5)$$

If one of the constraints identified in equations (3), (4), or (5) does not comply, it means that the grid is unbalanced regarding its supply and demand. In case of unbalanced periods, it is possible to apply DR models to target a decrease in the demand.

Another service offered by the power flow analysis is the visualization of the energy grid via the graphical user interface of the CECOS platform. Similar to the power flow algorithm, the visualization is provided by the Pandapower library, which accepts a Python file containing network data as input and generates a network image file. Figure 25 depicts a realistic example of a low voltage network (i.e., used in the second dataset presented in the Datasets Sections).

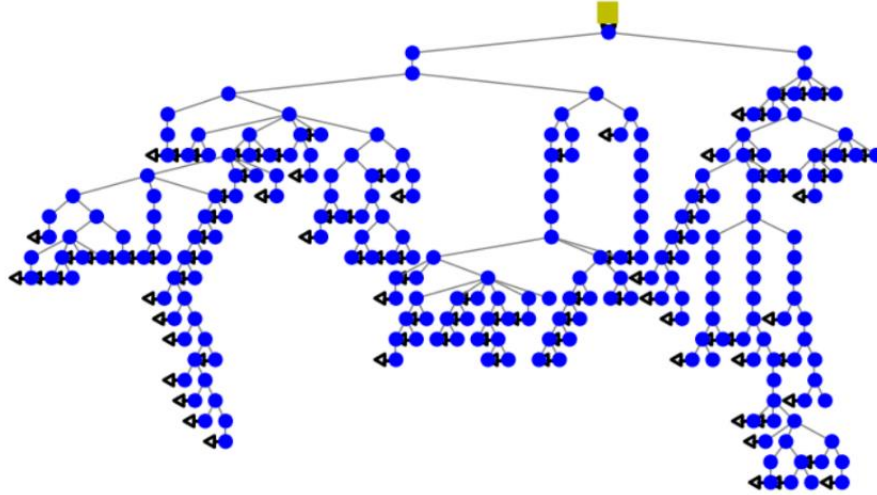


Figure 25 – Physical energy network graphic visualization (exported from CECOS)

5.3 Demand Response

This section describes two DR programs, one price-based and the other incentive-based, that can be implemented independently or combined to solve more complex problems, in what regards to congestion management in the energy grid. These programs were originally designed for use in citizen energy communities, but they are applicable to any group of energy customers.

5.3.1 Price-based Program

The price-based program uses the price elasticity profile of energy customers to set the real-time price for the energy community. Price elasticity (Pe) is defined as the ratio of consumption variation over the price variation, as proposed in (Thimmapuram et al., 2010):

$$Pe = \frac{\Delta C}{C} / \frac{\Delta T}{T} \quad (6)$$

where ΔC represents the difference between the final energy consumption (C), and the energy consumption of the last period, both in kWh. Relatively to ΔT , the same logic is applied, but it represents the difference between the final energy tariff price (T), and the last period's energy price, both in EUR/kWh. This equation, when applied to all data periods, yields points in a

cartesian system. The plotted points can then be used to construct an elasticity profile for each customer, based on linear and polynomial regressions, which can be used to forecast customers' consumption variation based on energy tariffs price variation.

The profiles can be used to predict the consumption variation of the customers, considering a certain price variation (Faria and Vale, 2011). In other words, the community operator asks for a reduction and the price variation is increased until the required variation in the community's demand is reached.

5.3.2 Incentive-based Program

The incentive-based DR uses an asymmetric auction model based on (Faia et al., 2019), to calculate how much the community operator will pay for the flexibility of each customer. Each customer specifies the price, in EUR/kWh, that will be accepted to reduce consumption. The community operator makes an auction call, where the customers will be able to present their bids. These bids are sorted in ascending order of price, so the smaller the price, the greater the possibility for the customers to sell their flexibility. The asymmetric model will be applied, where all the accepted bids required to meet the necessary flexibility amount are paid at the clearing price (i.e., the most expensive accepted bid). Therefore, the cost of the power cut (i.e., consumption reduction) is represented as

$$Cut\ Cost = Reduced\ Load * Clearing\ Price. \quad (7)$$

The maximum reduction applicable by a customer in the incentive-based program is given by

$$MaxReduction_p = Consumption_p * Flexibility_p \quad (8)$$

where $MaxReduction_p$ represents the customer's maximum reduction in period p , in kW, $Consumption_p$ represents the customer's consumption in period p , in kW, and $Flexibility_p$ represents the customer's flexibility in period p , in %.

5.4 Reinforcement Learning for Peer-to-Peer Transactions

Within the context of the smart grid, with the customer empowerment strategy, where decisions are made on the customer-side, it is becoming increasingly important to optimize both the customers' active participation and their energy resources usage. This optimization, which improves customers' participation in the smart grid, benefits both the customers and the grid as a whole, which can use the intelligent behaviours of its customers to improve the energy network management and gain the ability to automate its processes. Reinforcement learning (RL) is the chosen technique for solving this problem. This technique is applicable to agents and agent-based systems and aligns with the primary objective of integrating intelligence into the smart grid's services participation.

The proposed service aims to provide decision support for participants in local energy markets, regarding the amount of energy to be transacted, the price bided/offered for that transaction, and the use of flexibility to counter possible extra costs related to participation in P2P. In this service, the customers' representation agents can use reinforcement learning-based training to improve their participation in P2P markets using simulation environments. For this, two Deep RL algorithms will be used, i.e., multi-agent versions of DDPG and TD3, separately, allowing the choice of agents who will take advantage of them. The RL training in this service focuses on two main blocks, the environment and the agents. The environment incorporates the P2P model used and provides agents with customized observations for each one. The agents receive the observations from the environment, compute the action to take, determined by the policy or exploration mechanism, and then execute the after-market phase. The architecture of the service is shown in Figure 26. As can be seen, although there are several agents, with actions decided by themselves, these are centralized when entering the environment and the P2P market, in order to guarantee the integrity of the environment.

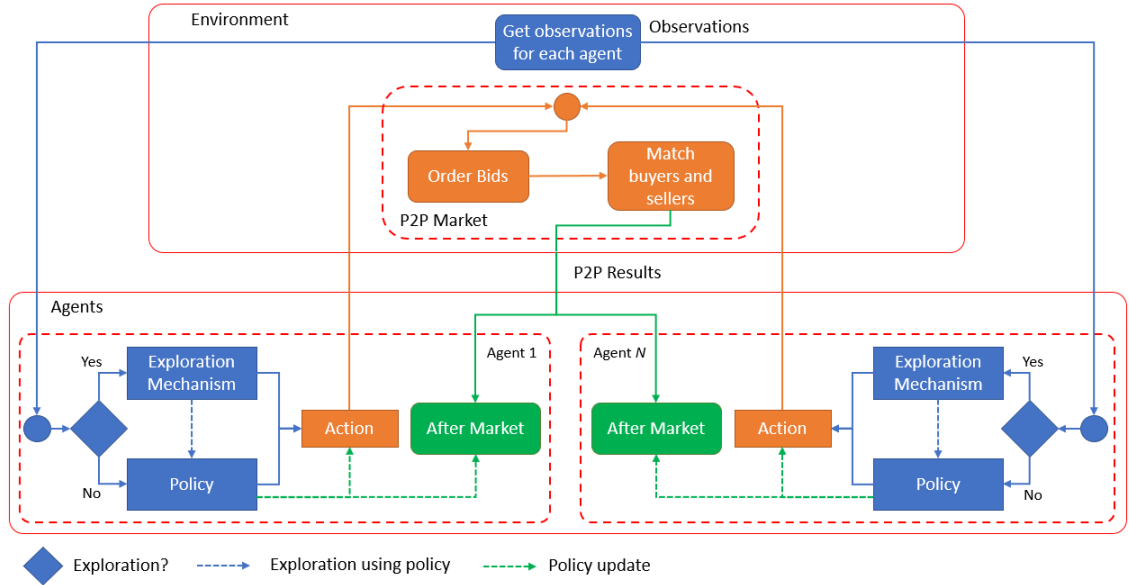


Figure 26 – RL environment and agents interaction in training

Regarding the observation of the state by an agent, it includes important factors for the customers' decision process. The observation for customer p in period t is given by:

$$o_t^p = (Forecast_t^p, Flexibility_t^p, Transactions_{t-1}^p, PeriodTime_t, ToU_t, FiT_t) \quad (9)$$

where $Forecast_t^p$ is the demand forecast of customer p for period t , in kWh, $Flexibility_t^p$ is the forecasted flexibility of customer p for period t , also in kWh, $Transactions_{t-1}^p$ is the list of transactions made by customer p in period $t - 1$ in the P2P market, including information about the price and quantities of energy transacted, and $PeriodTime_t$ provides information about the period of the day represented by period t . ToU_t and FiT_t are, respectively, the price to buy and sell energy to the energy retailer.

The agents' actions are related to the strategy that each one of them develops to participate in the P2P market. Thus, each agent generates two different actions, one regarding the price, and the other regarding the amount of energy to transact, both in the interval $[0, 1]$, representing a percentage value. Thus, the actions of each agent are given by:

$$a_t^p = (aPrice_t^p, aQuantity_t^p) \quad (10)$$

where $aPrice_t^p$ represent the action relative to the price to pay for energy, $aQuantity_t^p$ is the action that indicates the amount of energy to trade in the P2P market, both represented in percentual points, regarding period t and customer p .

Regarding the bid price, the prices offered/asked are limited to the purchase and sale price of energy on the grid. In what regards the energy amount to transact the agents consider the forecast error, and as that, the first step is to determine the potential error in a period. This error is computed using the evaluation metrics of the algorithm for forecasting at the time of testing. If it is the Mean Absolute Percentage Error (MAPE), it must be multiplied by the forecasted value for the period in question; if it is the Mean Absolute Error (MAE), it is used as the error's direct value. After calculating the error, the value of $aQuantity_t^p$ is applied within the forecast's possible range. As such, the price to pay and the amount of energy to be transacted are given by the following equations:

$$BidPrice_t^p = aPrice_t^p * (ToU_t - FiT_t) + FiT_t \quad (11)$$

$$Error_t^p = \begin{cases} MAPE * Forecast_t^p, & \text{if Metric} = MAPE \\ MAE, & \text{if Metric} = MAE \end{cases} \quad (12)$$

$$BidQuantity_t^p = aQuantity_t^p * ((Forecast_t^p + Error_t^p) - (Forecast_t^p - Error_t^p)) + (Forecast_t^p - Error_t^p) \quad (13)$$

where $BidPrice_t^p$ represents the price to pay in the P2P market, in EUR/kWh, $Error_t^p$ is the mean error of the forecast of the customer, in kWh, and $BidQuantity_t^p$ is the amount of energy to transact in the P2P market, in kWh, all regarding customer p in period t .

Bearing in mind that in this service is used an hour-ahead market, there is a need for energy forecast models to carry out the market, and not real demand values. Therefore, the true impact can only be measured in the period after the transactions, when the real values of consumption and generation are known. Thus, in period t the data related to the transactions carried out are stored, while in period $t + 1$, interactions with the grid to buy or sell energy are executed, and the reward for period t is calculated.

Regarding the calculation of the reward, it is directly linked to the savings made by the customer. The first step is to calculate the cost or profit of buying or selling the energy to the grid (i.e., $CostGrid_t^p$), where the actual demand of the customer is multiplied by the corresponding market price, represented in Equation (7). Then, the next step is to calculate the money transacted in the P2P market (i.e., $CostMarket_t^p$), that is given by the summation of price multiplied by energy transacted in each deal of the market, as represented in Equation (8).

$$CostGrid_t^p = Demand_t^p * \begin{cases} Price_t^{Buy}, & \text{if } Role_t^p = Buyer \\ Price_t^{Sell}, & \text{if } Role_t^p = Seller \end{cases} \quad (14)$$

$$CostMarket_t^p = \sum_{i=0}^N (TransactedEnergy_i * Price_i) \quad (15)$$

Even with market transactions, the interaction with the grid to buy/sell energy from/to the grid is almost inevitable. This because, using the forecast as a basis for the amount of energy to be transacted, there will always be errors, even if small, that make this interaction mandatory. The amount of energy to buy/sell from/to the grid (i.e., $EnExtra_t^p$) is given by equation (9) and is the difference between the real demand and the amount of energy traded in the market. In order to try to reduce the cost of interacting with the grid, when it is necessary to buy, that is, when a buyer does not transact enough energy in the market, or when a seller transacts more energy, flexibility is used to reduce costs. Equations (10) and (11) describe the process of calculating how much flexibility is needed, and the cost associated with this interaction in the after-market phase. In Equation (10) the amount of flexibility is given by the minimum between $Flexibility_t^p$ and $EnExtra_t^p$, both regarding customer p in period t , in kWh. On the other hand, the result of equations (9) and (10) are used to calculate the cost of buying/selling energy to the grid in the after-market phase. The flexibility is used only in the periods that demand the buying of more energy from the grid.

$$EnExtra_t^p = \sum_{i=0}^N (TransactedEnergy_i) - Demand_t^p \quad (16)$$

$$UsedF_t^p = \min(EnExtra_t^p, Flexibility_t^p) \quad (17)$$

$$CostExtra_t^p = \begin{cases} EnExtra_t^p * FiT_t, & \text{if } Role_t^p = Buyer \text{ AND } EnExtra_t^p \geq 0 \\ EnExtra_t^p * FiT_t, & \text{if } Role_t^p = Seller \text{ AND } EnExtra_t^p < 0 \\ (EnExtra_t^p - UsedF_t^p) * ToU_t, & \text{if } Role_t^p = Seller \text{ AND } EnExtra_t^p \geq 0 \\ (EnExtra_t^p - UsedF_t^p) * ToU_t, & \text{if } Role_t^p = Buyer \text{ AND } EnExtra_t^p < 0 \end{cases} \quad (18)$$

Finally, the reward is calculated by measuring the impact of participation in P2P in reducing costs or increasing profits, so there is a differentiation in the formula for sellers and buyers. The equation that gives the reward is then given by:

$$r_t^p = \begin{cases} CostGrid_t^p - CostMarket_t^p + CostExtra_t^p, & \text{if } Role_t^p = Buyer \\ CostMarket_t^p - CostGrid_t^p - CostExtra_t^p, & \text{if } Role_t^p = Seller \end{cases} \quad (19)$$

It is important to note that this service was conceived and developed on top of an abstract RL service, developed by the author of this dissertation, in order to facilitate the application of this field of machine learning to multi-agent systems. As previously stated, it is already common practice to apply RL to multi-agent systems, but agent-based systems must be developed intentionally to integrate RL from scratch. This service fills this gap in the state of the art, where

it is possible to integrate RL in previously conceived and deployed agent-based systems, to incorporate intelligent participation in services using RL models.

The environment is a crucial component of RL models, and it is typically the bottleneck in terms of development simplicity, as it is created using complex data that requires intensive processing. In the proposed abstract service, the RL environment is completely abstract, and customers are only required to select the environment's principal attributes, which in this case are the action space, the observation space, and the state transition function. Both the action and observation spaces can be conceived with the different types of available data (discrete or continuous), and it is possible for these spaces to be constructed automatically, with the service identifying the types of data being processed for each space. In addition, in some instances, the state transition function is not required, such as in time-dependent environments where agents' actions can have a direct effect on the environment but are not permanent across time periods.

This service also addresses the possibility of using actual environments to train the models. The service enables the use of simulated contexts, in which training is accelerated and agents interact with simulation-developed environments, but also enables the application of RL in real environments, with real energy customer data and physical devices, in the case of smart grids.

5.5 Chapter Conclusions

This chapter's primary objective was to describe the services that were conceived/adopted, developed, and integrated into the ecosystem as service-based agents. These services consist of energy demand forecasts, power flow-based energy network analysis, DR programs, and a reinforcement learning model for P2P participation.

Energy demand forecasts have a significant impact on the majority of other smart grid services, which are typically executed on an hour-ahead or day-ahead basis; therefore, it is crucial to integrate this type of service into the ecosystem. To test the virtual mobility between forecast services, the integrated services are state-of-the-art models with distinct error profiles. The energy network analysis and DR programs are integrated services that support a congestion management method.

The final integrated service is an optimization model for P2P market participation based on reinforcement learning. This model is developed on top of an abstract reinforcement learning service applied to multi-agent systems. As it allows for the intuitive development of reinforcement learning models, without the need to deal with data pre-processing and with the creation of observation and action spaces, it fills a gap in the existing literature. Using this service, the focus is on the definition of the action's value, state transition function, and rewards.

6 Case Studies

In this chapter, the developed case studies for testing and validation the proposed ecosystem and its functionalities are presented. Table 9 presents the descriptions of the case studies, including their main objectives.

Table 9 – Proposed case studies summary

Case study name	Main objective
Smart Grid Services Integration in Citizen Energy Community Operator System	Integrating multiple smart grid services in CECOS, in order to validate the platform's scalability and intuitiveness in the execution of the services
Customer-Aggregator Fitting for Demand Response Efficient Participation	Simulating ten weeks of participation in DR, allowing agents to move between aggregators to enhance energy customers participation and the customer-aggregator fitting
Optimize Participation in P2P Markets Using Reinforcement Learning	Training energy customers' agents with reinforcement learning models to optimize their participation in P2P markets
Smart Building Participation in P2P Markets Considering Energy Storage Systems	Enabling a smart building to participate in P2P markets using reinforcement learning models, considering mobility in forecast services to enhance the participation

6.1 Smart Grid Services Integration in Citizen Energy Community Operator System

This case study's primary objective is to evaluate CECOS's ability to integrate multiple smart grid services and provide them to energy communities. It consists of two main parts: (i) integrate tariffs management, customers aggregation, and price elasticity services, and (ii) integrate a congestion management methodology, with a power flow algorithm, and a price-based and incentive-based DR program. Each part of the case study employs a distinct dataset, being, respectively, the first and second datasets described in the Datasets section (i.e., Section 3.1.2). This case study is presented in two publications:

- [Journal] **Helder Pereira**, Luis Gomes, Pedro Faria, Zita Vale, Carlos Coelho (2021) “Web-based platform for the management of citizen energy communities and their members”, published in Energy Informatics in 2021, doi: 10.1186/S42162-021-00155-7
- [Conference] **Helder Pereira**, Ricardo Faia, Luis Gomes, Pedro Faria, Zita Vale (2022) “Incentive-based and Price-based Demand Response to Prevent Congestion in Energy Communities”, presented in 2022 IEEE International Conference on Environment and Electrical Engineering (EEEIC), doi: 10.1109/EEEIC/ICPSEUROPE54979.2022.9854648

In the case study’s first part, to enable the execution of the tariff calculation and attribution model, the developed platform demands the input of an file with a set of available consumption and generation tariffs. The used tariffs were generated considering hourly prices based on the MIBEL market. The tariffs file contains five consumption and five generation tariffs. After the tariffs are loaded to CECOS, it is possible to calculate the best tariff, by type (consumption or generation), for each one of the customers. After the calculations are finished, the operator can visually analyze the original and the most profitable tariff to each customer, through a comparative graph, shown in Figure 27. It is also possible to check if the best-calculated tariff is better than the original, considering the cost or profit.

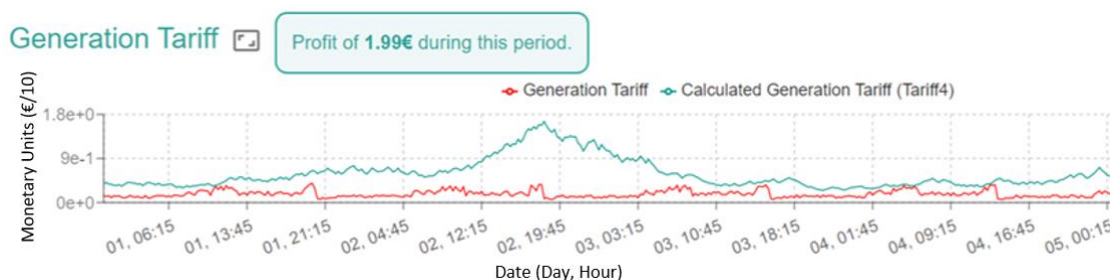


Figure 27 – Most profitable tariff comparison with the original tariff (exported from CECOS)

For the aggregation service, multiple parameters combinations were studied. The chosen ones were consumption profile, consumption type, generation profile, generation type, and contracted power. To examine the details of the aggregation service, the results for the combination generation profile and generation type are analyzed individually. The optimal number of clusters obtained by the Elbow method was 3, and by the Silhouette method was 8. In Figure 28, it is possible to analyze the generated evaluation graphs by those methods.

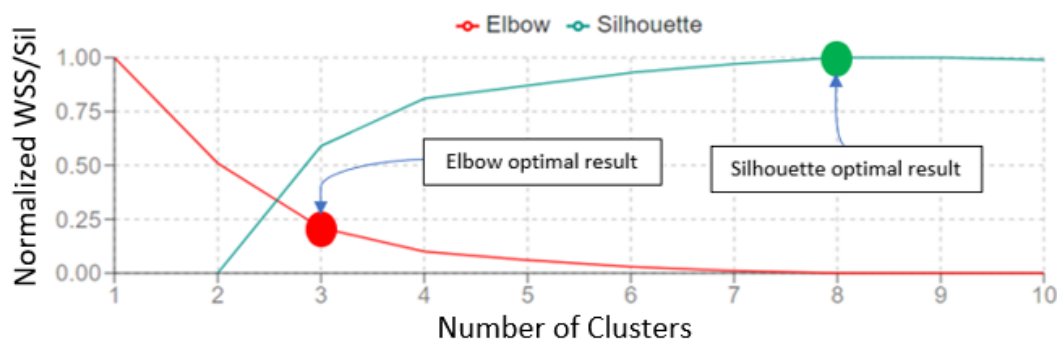


Figure 28 – Customers’ aggregation evaluation metrics (exported from CECOS)

Considering the Elbow method result as optimal, the data will be divided into three different groups. In Figure 29 it is possible to analyze the created clusters. The first cluster is composed totally of the customers without energy generation (generation type is None). The second cluster contains the customers with wind-generated energy, and the last one contains all the other customers. It is an interesting separation because it does not split any of the customers with the same generation type into more than one cluster, being helpful for future analysis. However, it also is an expected division, considering that the non-producers (i.e., consumers) have an exclusive group, and the wind generation is in majority on the dataset (254 customers). On the other hand, if the considered evaluation method is Silhouette, the optimal number of groups is eight. It is an expected number since there are eight different generation types on the dataset. With eight clusters, it is made a perfect division, considering that each one of the clusters is exclusive to one generation type. Both considered number of clusters provides a good division of the dataset. The chosen number of groups will always depend on the context and the goal of the division since there are multiple applications for both obtained cluster numbers.

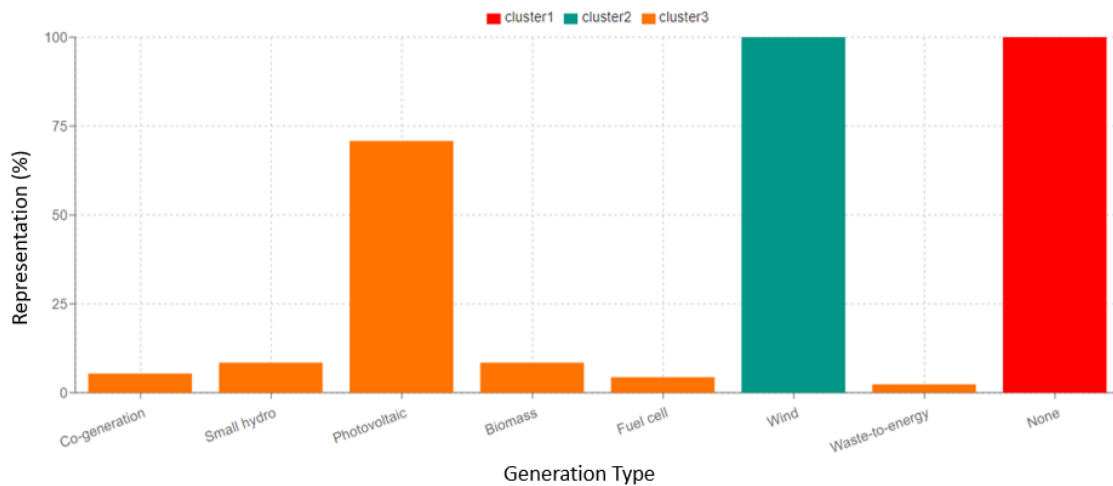


Figure 29 – Customers’ aggregation results for three clusters (exported from CECOS)

In what regards the price elasticity service, the first step is to calculate the price elasticity for each period of the dataset, to each of the customers. Then, all the calculated points are plotted in a scatter plot, where it is possible to analyze the elasticity of each customer. Executing the service, all the resulting regressions have negative coefficients. It is an expected result since the consumption and tariff price should have opposite directions. This because, if the customers’ profiles are explored, it is normal to decrease energy consumption when there is a price increase, and the same happens for the opposite situation. As expected, none of the linear regressions achieved a greater score than the quadratic regressions. Even so, the linear regressions in the graph were maintained for purposes of comparison and testing. The average score for linear regressions was 0.498, and for quadratic was 0.688.

After this part, is made the transition to the second part of the case study, where are integrated the proposed grid analysis model and DR models. Besides using the previously described data, in order to enable the use of power flow algorithms, it is used a realistic low voltage distribution network, composed of 236 buses and 235 lines, supplied by an 85 kW medium voltage

connection with a transformer, visually represented in Figure 25. This part of the case study uses the described data to (i) execute the power flow algorithm, (ii) predict congestion issues, and (iii) launch DR programs. The results compare the use of the proposed DR programs (i.e., incentive-based vs price-based), and the combination of both.

Executing the grid analysis model, the results show that there are 31 periods (i.e., 4.6 % of total periods) above the grid's transformer limit. Despite being a significant number of possible failures, the values above the threshold are not very high, varying from 0.546 kW to 17.571 kW. The periods above the threshold are used to launch the proposed DR programs.

The programs were executed independently, and their results for days 2 and 3 (to facilitate the reading) are summarized in Table 10. The table presents, for all the considered periods, the reductions achieved by each DR program, as well as the strategy to achieve the pretended results. For example, the necessary increase in energy price is calculated for the price-based program, whereas the clearing price of the market is determined for the incentive-based.

Table 10 – Demand Response simulation considering power flow results for days 2 and 3

Period (Day, Time)	Needed Reduction (kW)	Price-based			Incentive-based		
		Average Reduction (kW)	Price increase (%)	Number of participating customers	Average Reduction (kW)	Clearing price (EUR/kWh)	Number of participating customers
2, 08:00	6.164	6.328	2.10	70	6.164	0.098	31
2, 21:00	10.218	10.354	3.34	79	10.218	0.131	36
2, 21:15	17.571	-	-	-	-	-	-
3, 08:00	3.161	3.197	1.21	60	3.161	0.040	24
3, 21:15	1.089	1.105	0.59	31	1.089	0.034	10
3, 21:30	11.211	11.623	3.51	78	11.211	0.138	54
3, 21:45	9.900	9.942	3.26	77	9.900	0.110	44
3, 22:00	4.060	4.103	1.46	62	4.060	0.074	15

Relative to the price-based program, it is noticeable that the customers in this community are meaningfully reactive to energy tariff changes. Therefore, the necessary price increase to achieve the pretended reductions was never very high. In this program, the reductions are not exactly the necessary ones. This happens because the reduction is the dependent variable since the one that is controlled and independent is the price variation. It can be positive, to not keep the transformer at the exact load limit it can handle.

The required reductions were exactly met in all studied times, except for a specific period (day 2, 21:15), that required both programs to be executed. The asymmetric market's implementation allows an accurate reduction in each period, as the last accepted offer is decreased to the power required to cover the reduction needs. It is also possible to observe, that the cut cost in each of the periods is strongly correlated with the aggregated flexibility of consumers, since the greater this flexibility, the more easily the necessary reduction is achieved, and consequently, the cost is lower.

Analysing the DR programs outcome, it shows that one of the intervals does not follow the same process as the others. Neither the price-based nor incentive-based program could achieve the needed decrease to prevent the detected issue. In these instances, the incentive-based program might be applied first, and then the price-based. Customers would be able to sell their flexibility first and then lower their consumption due to the tariff increase. Considering this, the asymmetric market pool reduced 16.668 kW, leaving a 0.903 kW gap. The price-based program was used to produce the remaining reduction, increasing the price by 4.05 % and reducing 0.927 kW. The two programs working together prevented a problem that neither could handle alone.

6.2 Customer-Aggregator Fitting for Demand Response Efficient Participation

The goal of this case study is to illustrate the use of A4SG and its architecture's performance when tested in a real-world scenario of DR participation. The goal is to engage all the available customers in DR initiatives that are appropriate for them. For that, 30 customers are initially assigned to one of the three aggregators available within the ecosystem. Over time, if the customers are not satisfied with the results, they will request to be moved to another aggregator, and so on, until they achieve a high flexibility fitting rate. The profiles of the considered customers use the data available in (Pereira et al., 2022b). Having different types of customers means having different types of consumption profiles and flexibility, and therefore, different needs regarding the demand reductions made by DR events. Three aggregators with different profiles of needed reductions are simulated. The 3 defined aggregators have different profiles, and as such, they will attract different types of customers to their DR programs. One of the aggregators will have a profile that is more appealing to residential customers, another to industrial customers, and the other to customers who fall somewhere in the middle. These differences will motivate customers' mobility. This case study resulted in one scientific paper:

- [Journal – Under review] **Helder Pereira**, Luis Gomes, Zita Vale “Agent-based Ecosystem with Branching and Mobility Mechanisms to Support Agent Communities: a Smart Grid Application”

To evaluate the weekly performance of the customers in DR programs, the following equations are considered:

$$Part_i = \begin{cases} 1, & \text{if } Flex_i \geq MinP_i \\ 0, & \text{if } Flex_i < MinP_i \end{cases} \quad (20)$$

$$TotalPart = \sum_{i=0}^N \text{Min} \left(\frac{Flex_i}{MaxP_i}, \frac{MaxP_i}{Flex_i} \right) \times Part_i \quad (21)$$

$$Flexibility \text{ Fitting Rate} = \frac{TotalPart}{N} \quad (22)$$

where, $Flex_i$ represents the flexibility, in kW, of the customer in the period of the event i , $MinP_i$ represents the minimum participation per customer of the DR event i , $MaxP_i$ represents the maximum paid participation in kW, and N represents the number of DR events from the aggregator in the evaluation period.

Equation (20) determines if a customer can participate in the aggregator's DR event i . In other words, it evaluates if the customer's flexibility throughout the event period is more than or equal to the aggregator-defined minimum participation. If the consumer does not meet the minimum participation requirement, the event's parcel is set to zero, contributing negatively to the flexibility fitting rate. Equation (21) calculates the sum of the quotient between the customer's flexibility and the maximum paid participation in each of the DR events. To ensure that the evaluation is reliable and that unfavourable evaluation periods for different causes (i.e., excess and lack of flexibility) do not cancel each other out, the minimum between two parcels is determined, to ensure that the participation in a given event is always between 0 and 1. Equation (22) calculates the flexibility fitting rate of a customer in relation to an aggregator, and consequently, to its DR events. This evaluation is given by the quotient between the result of Equation (21) and the number of the customer's current aggregator DR events in the evaluation period. The Flexibility Fitting Rate is a value in the interval $[0, 1]$ (i.e., a percentual value). If the flexibility fitting rate is in the interval of $[0, 0.5]$, the customer will move, necessarily, to another aggregator, without considering the current one as an option. If it is in the range $]0.5, 0.7]$, the mobility will be considered, but the current aggregator will also be included as a possible destination, meaning that the mobility result can be the aggregator where the customer already is. If the flexibility fitting rate is above 70 %, the customer will remain in the current aggregator during the following week.

The simulation is executed for ten weeks, where, at the end of each week, each of the customers will evaluate their participation in the respective aggregators, with the use of the proposed flexibility fitting rate. At the initial week, the customers distribution to the aggregators is executed randomly. The simulation is processed in one-hour periods, making a total of 120 periods per week (excluding weekends). As such, the simulation will have a total of 1200 periods. In each of the weeks, each aggregator will launch between one and five DR events. The branching functionality will be used in the initial phase of the simulation to allow customers to subscribe to the aggregators they are initially associated with. Using this functionality, the number of agents related to customers will be 60 in total, 30 of which will be representation agents and 30 will be goal-oriented agents (i.e., where the objective is to participate in aggregators). In addition, there will be multiple ecosystem agents deployed, including a MAS main agent, 3 agents to represent the aggregators (ACOM main agents), a mobility agent, and a synchronizer agent to guide the simulation. Figure 30 represents the considered agents and groups, although the figure does not show all the amount of representation agents, to facilitate its reading and understanding. In this figure are represented the aggregator's groups and their agents. In this case study, the mobility functionality is used at the end of every week of simulation, depending on the respective evaluations of the agents, with the support of the Group Selector component, to send the best mobility options to each agent.

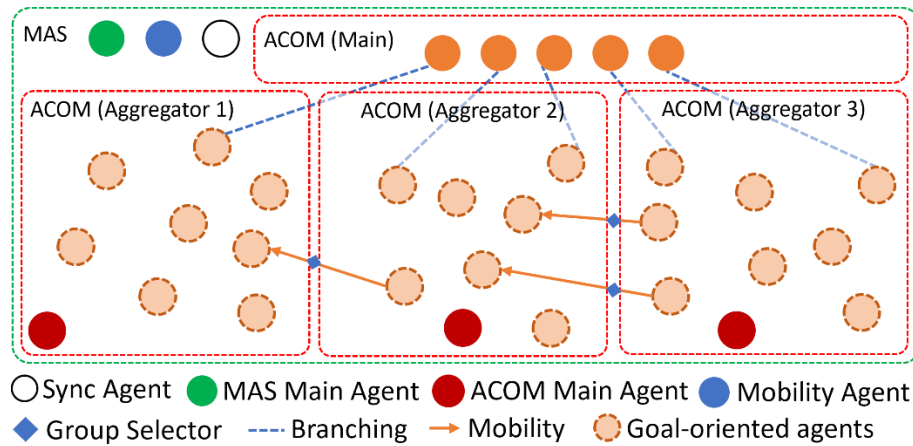


Figure 30 – A4SG Agents Layer architecture considering the case study structure, between week 9 and 10 of simulation (i.e., agents and groups)

The results from the simulation, regarding the number of mobilities and the types of customers in each aggregator, are summarized in Table 11. In this table, for each aggregator, are described the types of customers that compose them in each of the weeks, besides the number of customers that moved in and out at the beginning of each week. Observing the table, it is possible to see that the amount of mobility decreases over the weeks, which means that customers' assignments to aggregators are converging and evolving to the aggregators best suited to their consumption and flexibility profiles.

In addition, it is possible to see that Aggregator 1 is mainly composed of residential and small commerce customers, thus being good for smaller levels of flexibility. On the other hand, Aggregator 3 is primarily composed of industrial customers, as it requires members with a significant consumption reduction capacity. Aggregator 2 serves intermediate needs, and as a result, a sizable portion of its members are large commerce customers.

Table 11 – Customers' description and mobilities for each aggregator per week

Week	Aggregator 1					Aggregator 2					Aggregator 3				
	Number of Customers (+ In; - Out)	Types of Customers				Number of Customers (+ In; - Out)	Types of Customers				Number of Customers (+ In; - Out)	Types of Customers			
		R	SC	LC	I		R	SC	LC	I		R	SC	LC	I
1	9 (+9; -0)	4	2	2	1	9 (+9; -0)	3	1	2	3	12 (+12; -0)	3	7	1	1
2	9 (+2; -2)	5	3	1	0	11 (+4; -2)	4	3	1	3	10(+3; -5)	1	4	3	2
3	10 (+2; -1)	5	4	1	0	12 (+2; -1)	4	5	1	2	8 (+1; -3)	1	1	3	3
4	12 (+2; -0)	7	4	1	0	11 (+2; -3)	3	4	2	2	7 (+2; -3)	0	2	2	3
5	14 (+3; -1)	7	7	0	0	7 (+0; -4)	2	2	2	1	9 (+3; -1)	1	1	3	4
6	16 (+2; -0)	9	7	0	0	5 (+0; -2)	1	2	1	1	9 (+1; -1)	0	1	4	4
7	16 (+1; -1)	8	8	0	0	6 (+2; -1)	2	2	2	0	8 (+1; -2)	0	0	3	5
8	15 (+0; -1)	8	7	0	0	7 (+1; -0)	2	3	2	0	8 (+0; -0)	0	0	3	5
9	17 (+2; -0)	9	8	0	0	5 (+1; -3)	1	1	2	1	8 (+1; -1)	0	1	3	4
10	18 (+1; -0)	10	8	0	0	6 (+2; -1)	0	2	3	1	6 (+0; -2)	0	0	2	4

R = Residential; SC = Small Commerce; LC = Large Commerce; I = Industrial

In general, from the aggregators' perspective, it is less vital to know who their customers are than it is to ensure that the required consumption reductions are accomplished through their flexibility. Thus, it is critical to verify that the aggregators have the capacity to meet the objectives of their DR programs. Figure 31 depicts the percentage of events in which the required reductions were achieved over the course of the simulated weeks. It is possible to observe that the trend is positive, in the sense that the number of DR events that are successfully completed goes up over the weeks. This is important for both aggregators and customers, because financially it is essential that the necessary reductions are achieved, so that they can be rewarded for their participation. From the customers' perspective, it is not only essential to be able to participate in the events, but it is critical that their aggregators reach the required reduction, in order to be paid for participating.

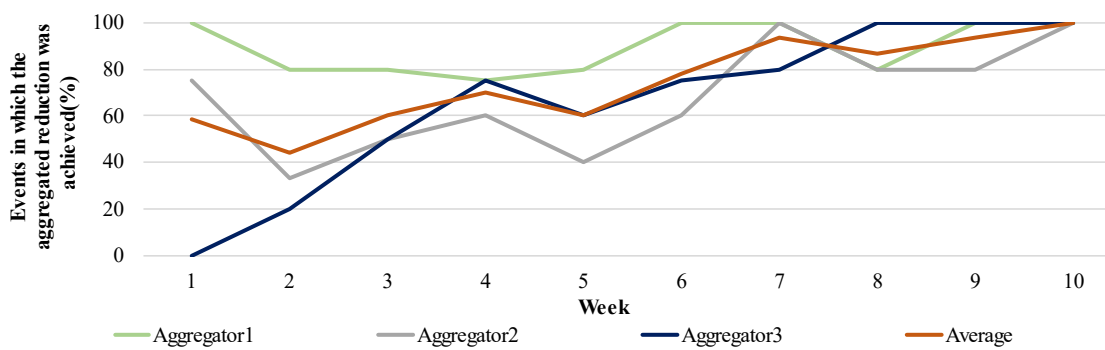


Figure 31 – Percentage of events in which the aggregated reduction target was achieved for each aggregator per week

With regard to the individual participation of customers, its assessment was carried out with the application of Equation (22). Figure 32 shows the area between the minimum and maximum flexibility fitting rate for each of the simulation weeks, as well as the average flexibility fitting rate of all the customers (which is calculated as an arithmetic average of the flexibility fitting of each of the customers). It is possible to conclude that the three studied metrics (minimum, average, and maximum) have an upward trend across the simulation, demonstrating the advantage of using a mobility functionality that considers the profiles of the customers and their association to the available aggregation entities. The increase in these metrics can mean several advantages, both for customers and for aggregators, for instance, from a financial point of view.

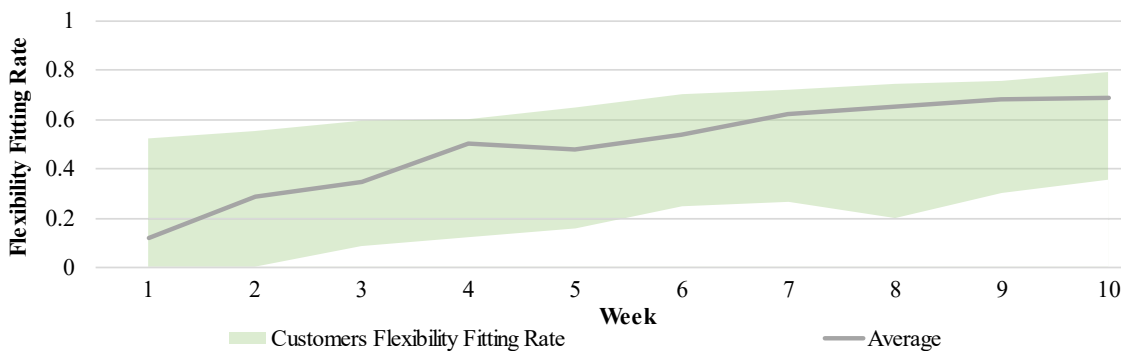


Figure 32 – Customers' flexibility fitting rate area (minimum and maximum) and average

Analysing the positive evolution in customers' flexibility fitting rate, it reveals a correlation with the amount of mobilities performed inside the ecosystem. As expected, the higher the customers' flexibility fitting rate, the lower the number of mobilities in the week. Taking into account the evaluation metrics and the ranges of values used to determine whether a customer should consider moving to another aggregator or not, the decrease in the number of mobilities over time is a positive indicator, meaning that the customers are converging to the aggregators that offer the type of participation that is most suited to their profile.

In contrast to previous customer selection methodologies, which aim to find the best customers for one aggregator based on their profile and context, the methodology employed in this case study considers multiple aggregators and prioritizes customer benefits. For instance, in the works proposed in (Silva et al., 2020), (Martinez-Pabon et al., 2017), and (Liang and Ma, 2022) customers are evaluated taking into account whether they are eligible to participate in DR, whether their flexibility profile is suitable to the type of DR launched, and how robust and reliable is their participation in different contexts.

Comparing the proposed methodology with the work developed in (Silva et al., 2020), there is a possibility of an increase in the percentage of events in which the aggregator achieves the necessary demand reduction for the event to be successful, since the defined aggregators in the developed case study resulted in 100 % of events achieving the aggregated reduction and in the referenced work these results were only 73 %. These results support the notion that focus and benefit the customers also have favourable consequences for aggregators, as they strengthen the context between customers and aggregators. This type of methodology, where the customer makes the decision, is consistent with the European Union's smart grids strategy, which focuses on customer empowerment. Moreover, as this case study demonstrates, placing the decision on the customer side can be advantageous for the smart grid entities.

6.3 Optimize Participation in Peer-to-Peer Markets Using Reinforcement Learning

In order to test the proposed methodology, a case study was developed using an energy community of 50 customers (Goncalves et al., 2022). The objective was to validate whether the two reinforcement learning algorithms used allow the A4SG ecosystem's agents to improve their participation in the P2P energy transactions market. This case study resulted in two scientific papers:

- [Journal] **Helder Pereira**, Luis Gomes, Zita Vale, (2022) "Optimizing participation in local energy market through deep reinforcement learning in an agent-based ecosystem", accepted for publication in Energy Reports
- [Journal] **Helder Pereira**, Luis Gomes, Zita Vale, (2022) "Peer-to-Peer Energy Trading Optimization in Energy Communities using Multi-Agent Deep Reinforcement Learning" accepted for publication in Energy Informatics

Of the 50 customers of the energy community, 20 will be sent to training with TD3, 20 to training with DDPG, and 10 will remain without training, in order to have a strong component of comparison between these three aspects. The distribution of the customers is random. The flow and steps of the case study are shown in Figure 33. The objective is to (i) simulate a P2P week without training, (ii) train the agents with the proposed RL models, and then (iii) compare the simulated market week with the agent's participation in the market with the trained policies. From the point of view of the A4SG ecosystem, the agents will be distributed among the different training ACOMs, despite being together in the energy community ACOM.

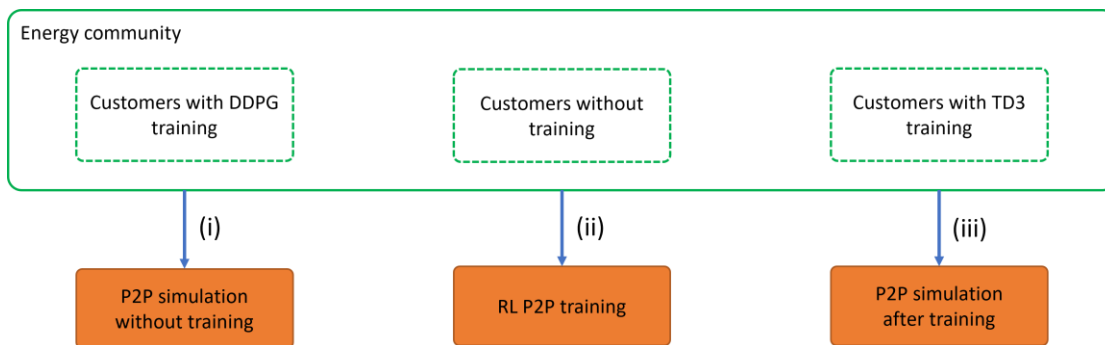


Figure 33 – P2P participation optimization case study steps

In order to compare the performance of P2P agents with and without training, three negotiation profiles were used prior to RL training to generate offers:

- Greedy profile: agents adhering to this profile will attempt to trade between 80 % and 100 % of the forecasted demand at a price between 20 % and 80 % within market limits;
- Safety profile: where agents attempt to negotiate between 90 % and 110 % of the forecast, while in terms of price, they use the limits between 60 and 100 %;
- Cheaper profile: where agents try to seek deals at a lower price. These contracts aim to trade between 80 % and 100 % of the forecast but utilize only between 0 % and 50 % of the price limits.

The most effective method for assessing the performance of reinforcement learning algorithms is to examine agent rewards across training iterations. An increase in the value of rewards indicates that agents are performing better at the task for which they are undergoing training. However, in competitive contexts such as the P2P market, it is common for some agents to achieve better results than others, as strategies are not shared and each agent seeks to achieve the best results for himself, which in most cases results in poorer outcomes for the others. Figure 34 depicts the average reward received by each agent for each algorithm throughout the training episodes. In general, the TD3 algorithm enabled better learning for most agents, whereas roughly half of the agents in the DDPG were very close to their initial values. In addition, agents with TD3 saved an average of 0.05 EUR per period, whereas, in the DDPG, this value was close to 0.03 EUR.

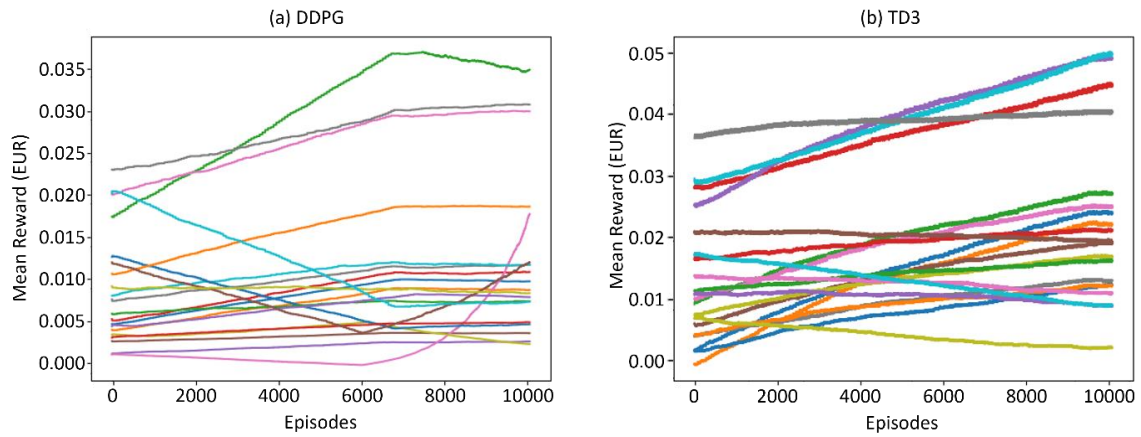


Figure 34 – Mean reward for each agent per training episode: (a) DDPG, and (b) TD3

From the perspective of the general community, there are greater differences in terms of rewards between the two algorithms. The minimum, mean, and maximum are depicted in Figure 35 for the agents that participated in the training of each algorithm. On the one hand, the TD3 has a much higher maximum reward value than the DDPG, with a value of 0.47 compared to 0.34 for the DDPG. Regarding the average value, the behaviour of both was quite similar, as the difference was only 0.02; however, the TD3 again held the advantage. On the other hand, as far as the minimum value is concerned the DDPG has the advantage, and based on the analysis of the graph, there appears to be a greater balance between all market-participating agents. This indicates that, with the use of DDPG and a competitive strategy, a balance has been reached between the customers' profits.

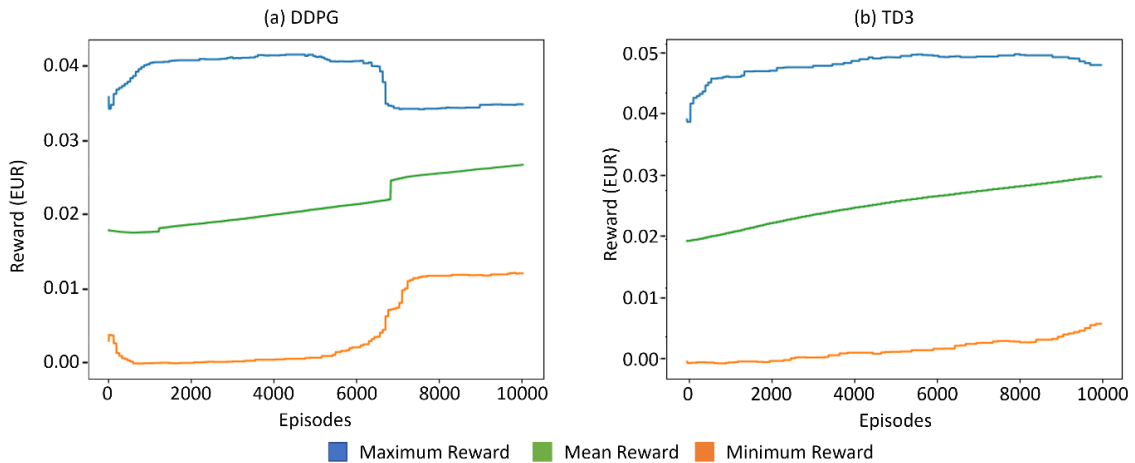


Figure 35 – Maximum, mean, and minimum reward per episode: (a) DDPG, and (b) TD3

Considering the overall savings of the community and the fact that the rewards are positive for all agents who participated in the training, it is possible to conclude that costs were reduced, i.e., the customers saved money through the training that enabled them to develop P2P participation strategies. Figure 36 depicts the results of participating in a P2P week before and after training, broken down by agents who trained with DDPG, agents who trained with TD3, and agents who received no training. As can be seen, the most significant difference is between

agents with and without training, with agents with trained policies having a significant advantage in the week following the training. From a total profit of 13.29 EUR to a loss of 3.78 EUR, untrained agents incurred a loss. Comparing trained agents, those with DDPG saved 15.99 EUR (i.e., from a cost of 7.38 EUR to a profit of 8.61 EUR, representing savings of 217 %), while those with TD3 saved 28.66 EUR (i.e., from a cost of 36.64 EUR to a cost of 7.97 EUR, representing savings of 78 %). These results demonstrate that agents trained with the TD3 algorithm realized greater cost savings. Analysing the results, in what regards the percentage, the DDPG had better results, but this is due to the low cost previously associated with the customers who trained with this algorithm, going from having costs to profit.

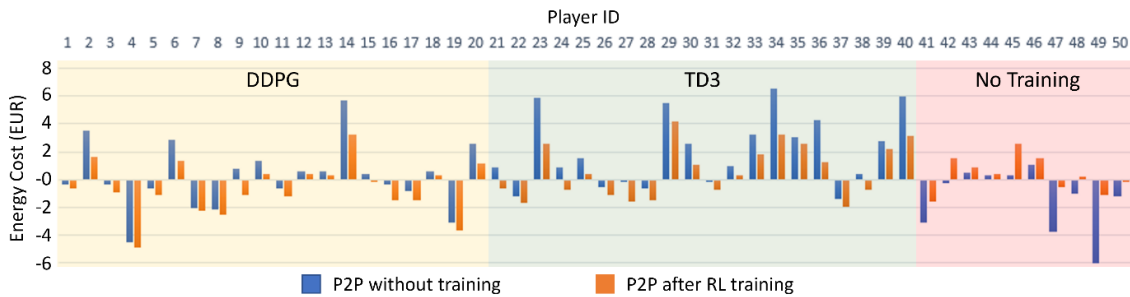


Figure 36 – Comparison of the energy costs during the week before and the week after the RL training, for both RL algorithms and the agents without training

The final metric used is the amount of energy traded on the P2P market. As a matter of fact, it can also be used as an evaluation metric for the participation of agents in the market as a community, since the goal is to maximize the energy transacted to create greater sustainability within the energy community while interacting with the utility grid as little as possible. Figure 37 shows the energy exchanged per period in the week preceding and following RL training. As can be seen, in the vast majority of periods following training, the amount of energy transferred increased by 40 %, from 245.11 kWh to 342.84 kWh, getting much closer to the maximum value throughout the week (i.e., 387.39 kWh).

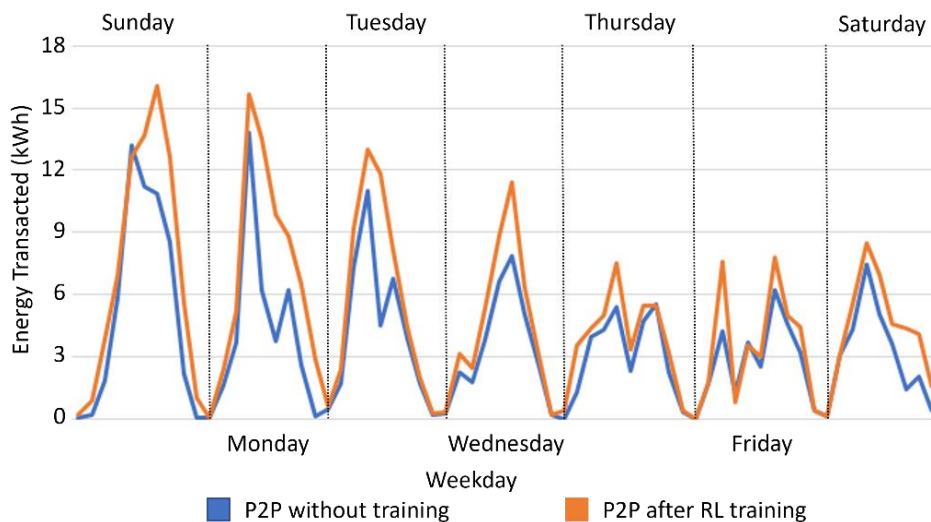


Figure 37 – Energy transacted in the P2P market before and after the RL training

The results obtained are highly positive, particularly when comparing the performance of trained and untrained customers. It is apparent that a customers trained with one of the RL models can make use of the intelligence it provides to employ the best strategies in each situation to minimize their energy costs. In addition, since training is conducted competitively with other customers, it is possible to perceive multiple strategies and prepare the customer for various market circumstances. The proposed methodology considers the forecast error to deal with uncertainty, being an improvement when compared, for instance, to the model proposed in (Qiu et al., 2021), where forecast errors are not considered. Also, this case study used a realistic dataset, with real measurements, for a community of 50 agents, contributing to the testing of the scalability of the proposed ecosystem and used algorithms. The dataset used is a public dataset that can be used by other authors to compare results (Goncalves et al., 2022).

6.4 Smart Building Participation in Peer-to-Peer Markets Considering Energy Storage Systems

This case study considers an energy community with a total of 20 energy prosumers, where: (i) energy community members are able to search for better demand hour-ahead forecast services, with the main objective of lowering their forecast errors, and (ii) using the forecasting results obtained from the first part of the case study, the community members can participate in a P2P market. The main objective is to understand if the decrease of forecast errors can increase positively the P2P participation. The results of this case study are part of two scientific papers:

- [Conference] **Helder Pereira**, Bruno Ribeiro, Luis Gomes, Zita Vale (2022) “CECOS: a Centralized Management Platform Supported by Distributed Services to Represent and Manage Resources Aggregation Entities and its End-users in a Smart Grid Context”, presented in “11th Symposium on Control of Power and Energy Systems” (CPES 2022), doi: 10.1016/J.IFACOL.2022.07.054
- [Journal – Under review] **Helder Pereira**, Bruno Ribeiro, Luis Gomes, Zita Vale “Smart grid ecosystem modelling using a novel framework for heterogenous agent communities”

In the considered energy community, 6 prosumers will have energy storage systems (ESS). Although the energy community members’ energy profiles were created by using external data sources (i.e., (Goncalves et al., 2022) and (Working Group on Intelligent Data Mining and Analysis (IDMA), 2022)), the ESS are actual units available in GECAD, creating a hybrid simulation that will use simulated data and physical ESS. The 6 energy community members that have ESS, will be able to use them in an after-market phase to reduce costs related to the purchase and sale of energy on the grid. The 20 members will be integrated into an ACOM representing the energy community in A4SG, with access to a P2P energy trading market. Taking advantage of the functionalities provided by A4SG, a three-week simulation will be carried out where customers can adapt to reduce the costs inherent to their energy resources. In this scenario, the agents are able to move between different forecast services, with different error profiles.

The three considered forecasts are: (i) a baseline forecast, only using mathematical models, (ii) a support vector machines (SVM) model, and (iii) an artificial neural network (ANN) model.

For customers' participation in P2P energy transactions, the bids (i.e., energy amount to be transacted and maximum/minimum price accepted) are generated from pre-trained reinforcement learning policies, using the algorithms integrated for the case study presented in Section 6.3 (i.e., DPPG and TD3).

At the beginning of the simulation, different groups and agents are created, both to guarantee the integrity of the ecosystem and of the simulation itself, as well as to represent energy community members and service providers. Table 12 presents the groups and the agents that are part of each one of them. In this case study there are two MAS: (i) in ARC there are the forecast groups, for the energy forecast service providers, represented by ACOMs, and (ii) in the Community MAS there is the ACOM to represent the energy community and its members. Each of the MAS has a Main ACOM to host representation agents, while the other ACOMs contain branch agents. Besides that, an ACOM main agent is needed in each one of the ACOMs to manage the groups and to communicate information about the ACOM to other agents, such as the objective of the group and the events or services history. As it is possible to conclude with the observation of Table 12, all groups, with the exception of the forecast groups, maintain their agents from the beginning to the end of the simulation. In the case of forecast groups, the members are dynamic, as energy customers will move between them, taking into account the performed evaluation.

Table 12 – Ecosystem structure to support the case study: groups and agents

Group	Ecosystem Agents	Representation Agents	Branch Agents
ARC	1 MAS Main Agent 1 Mobility Agent	-	-
Main-ARC	1 ACOM Main Agent	20 Customers 2 service providers	-
Forecast Baseline	1 ACOM Main Agent	-	1 Service-oriented (Forecast) <i>N</i> Goal-oriented (Customers) in each group
Forecast SVM	1 ACOM Main Agent	-	
Forecast ANN	1 ACOM Main Agent	-	
Community MAS	1 MAS Main Agent	-	-
Main-Community MAS	1 ACOM Main Agent	-	-
Energy Community with P2P	1 ACOM Main Agent	-	1 Service-oriented (Market Operator) 20 Goal-oriented (Customers)

The first situation to study is the mobility of agents across the forecast groups, and to see the effect that this mobility has on the error associated with the energy forecast services. As can be seen in Table 13, the final distribution of the agents is quite different from the initial one (i.e., random distribution), indicating that the agents were able to perform mobilities during the simulation to pursue better energy forecast results. At the end of the first week, the baseline

forecast lost more than half of its customers, and by the end of the second week, it was without any agents, proving that this was the worst service available to agents. Regarding the other two services, the one that uses an ANN as an algorithm is, in general, better than the service that uses an SVM. Even so, some of the agents stayed on the SVM forecast service provider until the last week, as it was the service that best adapted to their consumption and generation profiles. The community MAE dropped from 0.741 at the beginning of the simulation to 0.389 at the end.

Table 13 – Forecast services groups' customers and errors for each week of simulation

Week	Baseline Forecast		SVM Forecast		ANN Forecast	
	Number of Customers	Average MAE	Number of Customers	Average MAE	Number of Customers	Average MAE
1	8	1.036	6	0.650	6	0.438
2	3	0.627	6	0.513	11	0.401
3	0	-	5	0.459	15	0.366

One of the great advantages provided by PEAK, and used in A4SG, is the possibility of generating a chart with the groups that are part of the ecosystem so that it is possible to analyze them more graphically and intuitively. Figure 38 represents the chart from this case study. In this figure are represented the groups in the first and third weeks, showing the differences caused by the mobility functionality in the ecosystem hierarchy. Figure 38.a side shows the groups in the first week, and Figure 38.b shows the same groups in the third week. As expected, the chart shows the mobilities that were processed in the ecosystem during the simulation, reflecting the information provided in Table 13.

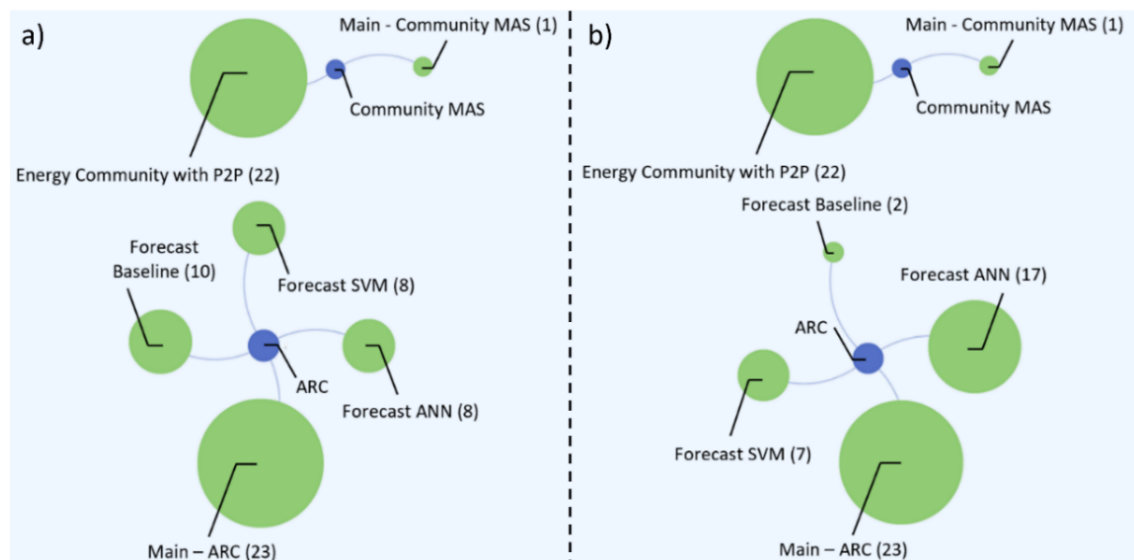


Figure 38 – Ecosystem structure (extracted and adapted from the PEAK graphical interface) representing the names and the number of agents of agent communities: a) agent communities at week 1 showing the distribution of agents among the forecast providers, and b) agent communities at week 3 showing the distribution of agents among the forecast providers.

Figure 39 illustrates data for a single day of the week and a comparison of forecasts for the one of the energy community members, in order to better illustrate the impact of forecast mobility. The considered prosumer passed by all the forecast services, starting at the baseline, moving to the SVM, and finishing in the ANN forecast. As can be seen, the initial forecast is significantly out of step with actual consumption. The SVM forecast, despite already showing improvements, identifies numerous peaks that did not occur in reality, whereas the ANN forecast allows for a substantial increase in the accuracy of the consumption forecasts.

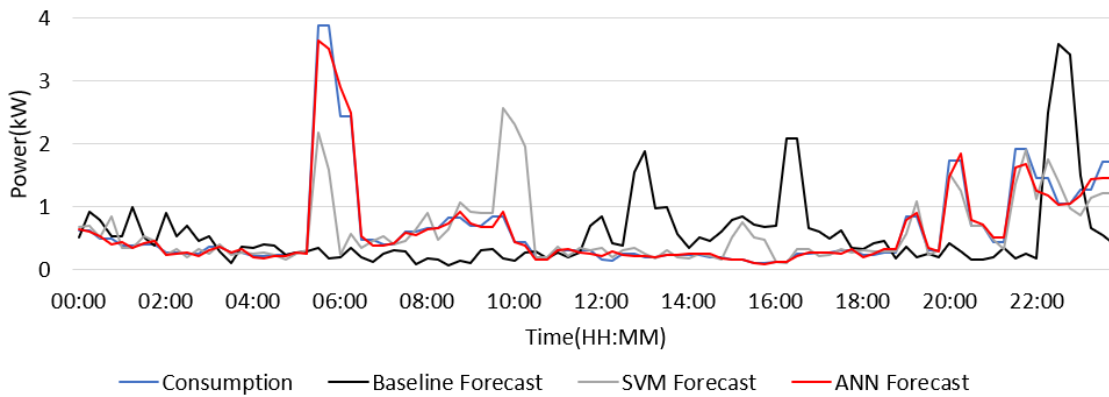


Figure 39 – Forecast services comparison for one of the studied energy community members

For these mobilities to have a positive effect on the energy customers, they must be used so that customers' participation in P2P markets is more effective and helps them to reduce energy costs. Figure 40 represents this analysis. In this figure, the second and third weeks are compared with the previous week for each customer in terms of energy costs. As it is possible to observe, most of the customers benefited from the mobility they performed, due to the decrease in energy forecast error. Only two of the energy community members were harmed (member 3 and member 7), that is, 90 % of the members benefited from the change of their forecast service provider to a lower error forecast. Overall, comparing the third week of simulation with the first, there were savings of 10.62 EUR in the community.

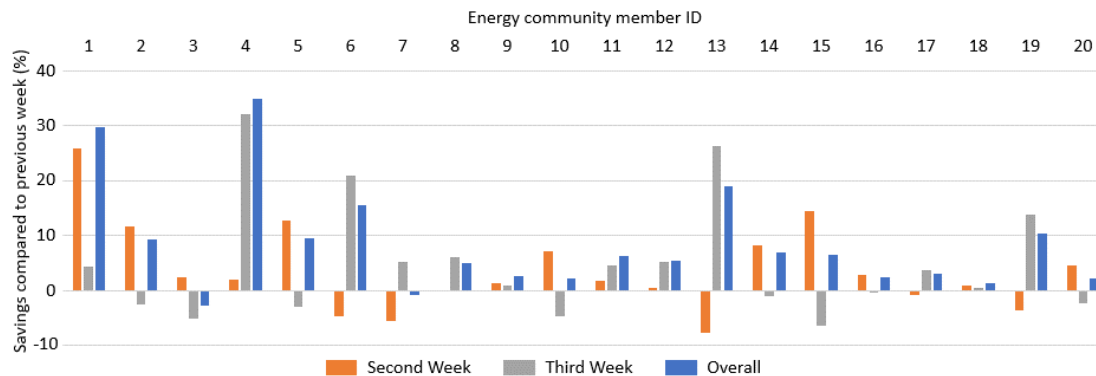


Figure 40 – Energy costs in the second and third weeks compared to the previous week

Another factor that was influenced by the mobility of the agents is the use of ESS. Since ESS are used after the peer-to-peer market, to minimize the buying and selling of energy from and to

the grid, the more accurate the bids for peer-to-peer are, the less the use of ESS will be required. Figure 41 represents a comparison of the ESS load (state-of-charge) between weeks for each period. As it is possible to observe, the ESS were less used with the advance of the weeks, and there was a 22 % decrease in ESS usage comparing the final with the initial week.

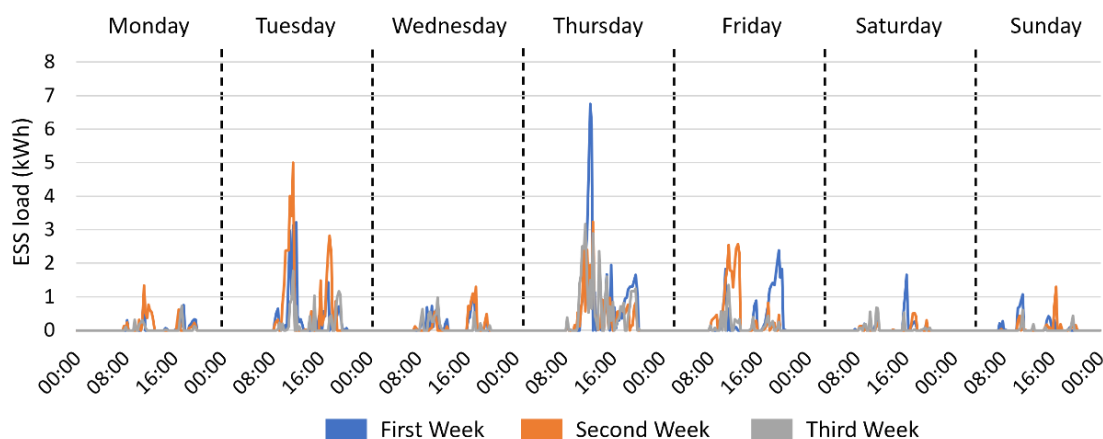


Figure 41 – ESS load (state-of-charge) comparison between weeks for each studied period

To understand the benefits that this methodology provides to energy customers, two additional scenarios are simulated to serve as a baseline to assess the full scenario. The first scenario employs neither the P2P market nor the mobility proposed by A4SG, while the second scenario employs P2P but without mobility. To compare the three simulated scenarios, Table 14 shows the data regarding all the scenarios. The greatest difference, as expected, is the introduction of the P2P market, especially with intelligent bid strategies provided by the reinforcement learning policies. Mobility allows for extra energy savings, proving that it can help energy customers to adapt to a context more advantageous to them, which is the reduction of the energy forecasting model to be considered for the P2P.

Table 14 – Scenarios comparison considering P2P and mobility

Scenario	Community Overall Cost (EUR)	Mean Energy Price (EUR/kWh)
Without P2P and mobility	585.49	0.142
With P2P and without mobility	392.34	0.126
With P2P and mobility	377.85	0.119

6.5 Chapter Conclusions

All the case studies presented and integrated into the proposed ecosystem demonstrated positive outcomes in terms of the energy customers' active participation and the smart grid's overall benefits.

The case study involving aggregators and DR was the greater test for the virtual mobility functionality, in which a personalized method for evaluating participation in DR was proposed and used to trigger the mobility of agents. In comparison to a random distribution of customers per aggregator, the mean proposed flexibility fitting rate increased from 12 % to 69 % in a simulation lasting ten weeks. This indicates that the adequacy between the DR events of aggregators and the flexibility profiles of energy customers increased.

In the case studies related to participation in P2P markets (i.e., Optimize Participation in P2P Markets Using Reinforcement Learning and Smart Building Participation in P2P Markets Considering Energy Storage Systems), intelligent participation in P2P markets was able to help energy customers reducing their energy bills. This intelligent participation was made possible through the application of the proposed reinforcement learning service, which was shown to be implementable in a complex smart grid domain dealing with the uncertainty that demand forecasting introduces. In addition, the integration of this type of service is essential for reducing the energy market prices and the CO₂ emissions generated by the customers of an energy community, as their interaction with the grid to buy and sell energy decreases.

In conclusion, the case studies were able to test each of the proposed services and functionalities that were integrated into the ecosystem, as well as the ecosystem as a whole, particularly in terms of agent deployment, communication, and interoperability. Moreover, compared to other types of methodologies, the functionalities incorporated into A4SG place the decision on the customer's side, emphasizing the strategy of customer empowerment.

7 Conclusions

This chapter will focus on the major conclusions drawn from this dissertation. Additionally, the achieved objectives will be described. Several functionalities and capabilities are proposed for future integration in order to contribute to the ecosystem's evolution.

7.1 Main Conclusions

With the completion of the described work, it is possible to conclude that all objectives have been achieved and all research questions have been answered. During the project's development, unexpected obstacles and difficulties emerged that demanded the re-planning of certain options in order to achieve the best results; however, the final results meet the initially defined objectives.

The first objective addressed in this dissertation was to deeply investigate the multi-agent system's state-of-the-art, especially when applied to the smart grid domain, to identify the main gaps of the literature (accomplishing O1 and answering RQ1, defined in Section 1.2). The proposed agent-based ecosystem (i.e., A4SG) presents an advance in the current multi-agent systems literature, filling the identified gaps in Chapter 2.

The proposed A4SG enables agents from multiple multi-agent systems and agent communities to coexist in a unified solution (related to O2, O3, and RQ2). A4SG's distributed solution allows it to circumvent the issue that agent-based systems are highly specialized and have specific objectives. This ecosystem takes a novel approach, providing an open society of agents in which all types of entities can register and take advantage of the ecosystem's functionalities and the services offered by other agents. Besides that, it enables the representation of real energy customers and smart grid entities, providing multiple types of agents to deal with the different parameters that they can handle. The most significant innovation relates to the system architecture and its novel functionalities.

The proposed functionalities have a great impact on the ecosystem operation and bring significant advantages to the smart grid customers. The physical mobility allows agents to move between physical hosts, which, when combined with the computing load balancing functionality, prevents the physical hosts registered in A4SG from becoming overloaded. The virtual mobility functionality covers a big gap in the literature of multi-agent systems and smart grids, as it enables agents to move between agent-based systems considering their context, in order to find the groups that best suit their energy resources, therefore representing the dynamic interactions of the smart grid. The branching functionality enables agents to have multiple representations in the ecosystem simultaneously. This facilitates the participation in multiple aggregation entities and smart grid services. Finally, the objective interdependency and the intra-agent conflict resolution functionalities enable different branch agents to optimize their performance and operation, considering personalized comfort and cost functions. The conception, development, testing, and validation of these functionalities contribute to O4, O5, O6, O7, O9, and RQ3.

In A4SG there is integrated a web-based platform, entitled CECOS, that can enable a more intuitive use of the ecosystem. It integrates multiple smart grid services, such as tariff management and demand response, and it also enables the direct registration of new services. It enables the monitoring of the agents and the data analysis of its resources. As such, a graphical user interface facilitates the connection between customers and agents, besides facilitating the understanding of service results.

The training of reinforcement learning policies for the ecosystem's agents is one of the services integrated into A4SG. This service allows the training of decisions in controlled environments provided by the reinforcement learning approach. Because this service is located in the ecosystem, outside individual agents, already existing and deployed agents can use this service without needing modifications in their source codes, representing a significant advancement when compared to state-of-the-art approaches. The development and integration of this service is related to O8 and RQ4.

To test and evaluate the use of the ecosystem, developed multiple case studies were developed, in which different functionalities and services of the solution are tested. The designed case studies demonstrated the capability of validating the entire solution. The results demonstrated the possibility for the energy customers to reduce their energy bills with the use of reinforcement learning to participate in peer-to-peer transactions or virtual mobility to enhance their participation in demand response programs through aggregators. These case studies contribute to O9 and O10.

As a result of the solution testing and validation, it is possible to conclude that customer empowerment strategies can be advantageous for both the customers and the system as a whole. A4SG's entire methodology is based on the idea of putting the decision-making power in the hands of the customers, based on customized functions and equations that are tailored to the preferences of each individual customer. With this, the entire smart grid can benefit, as each customer can automate the optimization of their resources, thereby contributing , for instance, to the reduction of CO₂ emissions.

The proposed ecosystem, being a combination of multiple complex models and functionalities, facilitates the practical implementation of the new European directives in the area of power and energy systems, namely energy markets, whose wide-scale execution demands novel models. This dissertation produced multiple substantial outputs that can be replicated and applied in other projects. Multiple models and functionalities were conceived, developed, tested, and validated, from which stand out: the reinforcement learning model to improve peer-to-peer participation, the virtual mobility functionality supported by a model of energy customers aggregation based on unsupervised machine learning, and the branching functionality that allows the decentralized pursuit of agent objectives.

This dissertation resulted in 10 scientific publications, from which six have been published or are in the process of being published in journals, and the other four have been presented at scientific conferences. These papers serve as proof that the proposed solution is innovative and fills the identified gaps in the state of the art.

7.2 Future Work

The proposed work has proved its value in achieving all the initially defined objectives in different contexts. Despite that, there are some possible adjustments and upgrades that can be made to improve the possible uses of A4SG.

The integration of multiple agent-based systems in A4SG demonstrated to be a good approach to intuitively improve the interoperability between different systems and agents. However, the definition of a common vocabulary, for instance, through middleware data translators can be a good improvement, that would facilitate other developers to integrate their systems into A4SG. Middleware data translators, unlike the proposed communication protocols translator in this dissertation, which aims to increase the number of agents that receive messages, aim to give a meaningful representation to each agents' message.

The implementation of the abstract reinforcement learning service shown to be easily deployed in a smart grid setting, more especially, peer-to-peer market participation optimization. To rigorously test this service, its implementation in other smart grid domains, such as participation in demand response programs, or more complex tasks, such as energy resources usage optimization, can be carried out. In addition, a more robust test of the service can be conducted in a domain unrelated to the smart grid.

The final future work task, that would have a great impact, is the test and adaptation of the A4SG to other domains. The core concept of the ecosystem, as well as the conception of the different types of agents, is unrelated to a particular domain; as such, it would be a good test of the ecosystem concept, and extremely beneficial if the results are positive.

References

- Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. TensorFlow: A System for Large-Scale Machine Learning. 12th USENIX Symp. Oper. Syst. Des. Implement. (OSDI 16), Savannah, GA: USENIX Association; 2016, p. 265–83.
- Abbas HA, Shaheen SI, Amin MH. Organization of Multi-Agent Systems: An Overview. *Int J Intell Inf Syst* 2015;4:46. <https://doi.org/10.11648/j.ijis.20150403.11>.
- Abrishambaf O, Lezama F, Faria P, Vale Z. Towards transactive energy systems: An analysis on current trends. *Energy Strateg Rev* 2019;26:100418. <https://doi.org/10.1016/J.ESR.2019.100418>.
- Acharya A, Prasad H, Kumar V, Gupta I, Singh AK. Host Platform Security and Mobile Agent Classification: A Systematic Study. *Lect Notes Data Eng Commun Technol* 2021;58:1001–10. https://doi.org/10.1007/978-981-15-9647-6_79.
- Aggarwal S, Kumar N, Tanwar S, Alazab M. A Survey on Energy Trading in the Smart Grid: Taxonomy, Research Challenges and Solutions. *IEEE Access* 2021;9:116231–53. <https://doi.org/10.1109/ACCESS.2021.3104354>.
- Agostinelli F, Hocquet G, Singh S, Baldi P. From reinforcement learning to deep reinforcement learning: An overview. *Braverman Readings Mach Learn Key Ideas from Inception to Curr State* 2018:298–328.
- Ahmad T, Chen H. A review on machine learning forecasting growth trends and their real-time applications in different energy systems. *Sustain Cities Soc* 2020;54:102010. <https://doi.org/10.1016/J.SCS.2019.102010>.
- Akyol B, Haack J, Carpenter B, Ciraci S, Vlachopoulou M, Tews C. Volttron: An agent execution platform for the electric power system. *Third Int. Work. Agent Technol. Energy Syst.*, Valencia, Spain: 2012.
- Aladdin S, El-Tantawy S, Fouda MM, Tag Eldien AS. MARLA-SG: Multi-Agent Reinforcement Learning Algorithm for Efficient Demand Response in Smart Grid. *IEEE Access* 2020. <https://doi.org/10.1109/ACCESS.2020.3038863>.
- Alduailij Mona A., Petri I, Rana O, Alduailij Mai A., Aldawood AS. Forecasting peak energy demand for smart buildings. *J Supercomput* 2021;77:6356–80. <https://doi.org/10.1007/S11227-020-03540-3/FIGURES/15>.
- Ansari J, Gholami A, Kazemi A. Multi-agent systems for reactive power control in smart grids. *Int J Electr Power Energy Syst* 2016;83:411–25. <https://doi.org/10.1016/J.IJEPES.2016.04.010>.
- Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA. Deep reinforcement learning: A brief survey. *IEEE Signal Process Mag* 2017;34:26–38. <https://doi.org/10.1109/MSP.2017.2743240>.
- Aryanto KYE, Dermawan KT, Putrama IM, Gupta D, Shivankar J, Gugulothu S. Instant messaging using

xmpp. J Phys Conf Ser 2021;1913:012126. <https://doi.org/10.1088/1742-6596/1913/1/012126>.

Asim MN, Wasim M, Khan MUG, Mahmood W, Abbasi HM. A survey of ontology learning techniques and applications. Database 2018;2018. <https://doi.org/10.1093/DATABASE/BAY101>.

Avancini DB, Rodrigues JJPC, Martins SGB, Rabêlo RAL, Al-Muhtadi J, Solic P. Energy meters evolution in smart grids: A review. J Clean Prod 2019;217:702–15. <https://doi.org/10.1016/J.JCLEPRO.2019.01.229>.

Azeroual M, Lamhamdi T, El Moussaoui H, El Markhi H. Simulation tools for a smart grid and energy management for microgrid with wind power using multi-agent system. Wind Eng 2020;44:661–72. <https://doi.org/10.1177/0309524X19862755>.

Baczynska A, Niewiadomski W. Power flow tracing for active congestion management in modern power systems. Energies 2020;13. <https://doi.org/10.3390/en13184860>.

Bagga P, Hans R. Mobile agents system security: A systematic survey. ACM Comput Surv 2017;50:1–45. <https://doi.org/10.1145/3095797>.

Baidya S, Potdar V, Pratim Ray P, Nandi C. Reviewing the opportunities, challenges, and future directions for the digitalization of energy. Energy Res Soc Sci 2021;81:102243. <https://doi.org/10.1016/J.ERSS.2021.102243>.

Balaji PG, Srinivasan D. An Introduction to Multi-Agent Systems. Stud Comput Intell 2010;310:1–27. https://doi.org/10.1007/978-3-642-14435-6_1.

Balli MF, Uludag S, Selcuk AA, Tavli B. Distributed Multi-Unit Privacy Assured Bidding (PAB) for Smart Grid Demand Response Programs. IEEE Trans Smart Grid 2018;9:4119–27. <https://doi.org/10.1109/TSG.2017.2651029>.

Beckett P. GDPR compliance: your tech department's next big opportunity. Comput Fraud Secur 2017;2017:9–13. [https://doi.org/10.1016/S1361-3723\(17\)30041-6](https://doi.org/10.1016/S1361-3723(17)30041-6).

Bhattarai BP, Paudyal S, Luo Y, Mohanpurkar M, Cheung K, Tonkoski R, et al. Big data analytics in smart grids: State-of-the-art, challenges, opportunities, and future directions. IET Smart Grid 2019;2:141–54. <https://doi.org/10.1049/IET-STG.2018.0261/CITE/REFWORKS>.

Botvinick M, Ritter S, Wang JX, Kurth-Nelson Z, Blundell C, Hassabis D. Reinforcement Learning, Fast and Slow. Trends Cogn Sci 2019;23:408–22. <https://doi.org/10.1016/J.TICS.2019.02.006>.

Bradley P, Leach M, Torriti J. A review of the costs and benefits of demand response for electricity in the UK. Energy Policy 2013;52:312–27. <https://doi.org/10.1016/J.ENPOL.2012.09.039>.

Braun P. The Migration Process of Mobile Agents Implementation, Classification, and Optimization. Faculty of Mathematics and Computer Science of the University of Jena, 2003.

Brito J, Campos P, Leite R. An Agent-Based Model for Detection in Economic Networks, Springer, Cham; 2018, p. 105–15. https://doi.org/10.1007/978-3-319-94779-2_10.

Brockman G, Cheung V, Pettersson L, Schneider J, Schulman J, Tang J, et al. OpenAI Gym 2016.

Bruno Ribeiro, Helder Pereira, Luis Gomes, Zita Vale. Python-based Ecosystem for Agent Communities simulation. Springer, Cham; 2022. https://doi.org/10.1007/978-3-031-18050-7_7.

Burger SP, Luke M. Business models for distributed energy resources: A review and empirical analysis. Energy Policy 2017;109:230–48. <https://doi.org/10.1016/J.ENPOL.2017.07.007>.

Cao D, Hu W, Zhao J, Zhang G, Zhang B, Liu Z, et al. Reinforcement Learning and Its Applications in Modern Power and Energy Systems: A Review. J Mod Power Syst Clean Energy 2020;8:1029–42. <https://doi.org/10.35833/MPCE.2020.000552>.

Caramizaru E, Uihlein A. Energy communities : an overview of energy and social innovation. Luxembourg: Publications Office of the European Union; 2020. <https://doi.org/10.2760/180576>.

Cardoso RC, Ferrando A. A Review of Agent-Based Programming for Multi-Agent Systems. Comput 2021, Vol 10, Page 16 2021;10:16. <https://doi.org/10.3390/COMPUTERS10020016>.

Casado-Vara R, Novais P, Gil AB, Prieto J, Corchado JM. Distributed Continuous-Time Fault Estimation Control for Multiple Devices in IoT Networks. *IEEE Access* 2019;7:11972–84. <https://doi.org/10.1109/ACCESS.2019.2892905>.

Celik ZB, Fernandes E, Pauley E, Tan G, Mcdaniel P. Program analysis of commodity IoT applications for security and privacy: Challenges and opportunities. *ACM Comput Surv* 2019;52. <https://doi.org/10.1145/3333501>.

Chen S, Liu CC. From demand response to transactive energy: state of the art. *J Mod Power Syst Clean Energy* 2017;5:10–9. <https://doi.org/10.1007/S40565-016-0256-X>.

Chen T, Bu S, Liu X, Kang J, Yu FR, Han Z. Peer-to-Peer Energy Trading and Energy Conversion in Interconnected Multi-Energy Microgrids Using Multi-Agent Deep Reinforcement Learning. *IEEE Trans Smart Grid* 2022;13:715–27. <https://doi.org/10.1109/TSG.2021.3124465>.

Cherkassy V, Mulier F. *Learning from Data: Concepts, Theory, and Methods*. Wiley; 1998.

Chicco G. Overview and performance assessment of the clustering methods for electrical load pattern grouping. *Energy* 2012;42:68–80. <https://doi.org/10.1016/J.ENERGY.2011.12.031>.

Choi HW, Muhammad Faseeh Qureshi N, Shin DR. Comparative Analysis of Electricity Consumption at Home through a Silhouette-score prospective. *Int. Conf. Adv. Commun. Technol. ICACT*, vol. 2019-February, Institute of Electrical and Electronics Engineers Inc.; 2019, p. 589–91. <https://doi.org/10.23919/ICACT.2019.8701923>.

Choudhary A. A Hands-On Introduction to Deep Q-Learning using OpenAI Gym in Python. *Anal Vidhya* 2019. <https://www.analyticsvidhya.com/blog/2019/04/introduction-deep-q-learning-python/> (accessed October 21, 2022).

Coelho VN, Weiss Cohen M, Coelho IM, Liu N, Guimarães FG. Multi-agent systems applied for energy systems integration: State-of-the-art applications and trends in microgrids. *Appl Energy* 2017;187:820–32. <https://doi.org/10.1016/j.apenergy.2016.10.056>.

Cucurull J, Martí R, Robles S, Borrell J, Navarro G. FIPA-Based Interoperable Agent Mobility. In: Burkhard H-D, Lindemann G, Verbrugge R, Varga LZ, editors. *Multi-Agent Syst. Appl. V*, Berlin, Heidelberg: Springer Berlin Heidelberg; 2007, p. 319–21.

Cui L, Qu Y, Gao L, Xie G, Yu S. Detecting false data attacks using machine learning techniques in smart grid: A survey. *J Netw Comput Appl* 2020;170:102808. <https://doi.org/10.1016/J.JNCA.2020.102808>.

Derhamy H, Eliasson J, Delsing J. IoT Interoperability - On-Demand and Low Latency Transparent Multiprotocol Translator. *IEEE Internet Things J* 2017;4:1754–63. <https://doi.org/10.1109/JIOT.2017.2697718>.

Devlin MA, Hayes BP. Non-Intrusive Load Monitoring and Classification of Activities of Daily Living Using Residential Smart Meter Data. *IEEE Trans Consum Electron* 2019;65:339–48. <https://doi.org/10.1109/TCE.2019.2918922>.

Diao R, Wang Z, Shi D, Chang Q, Duan J, Zhang X. Autonomous Voltage Control for Grid Operation Using Deep Reinforcement Learning. *IEEE Power Energy Soc Gen Meet* 2019;2019-August. <https://doi.org/10.1109/PESGM40551.2019.8973924>.

Dorri A, Kanhere SS, Jurdak R. Multi-Agent Systems: A Survey. *IEEE Access* 2018;6:28573–93. <https://doi.org/10.1109/ACCESS.2018.2831228>.

Dulac-Arnold G, Levine N, Mankowitz DJ, Li J, Paduraru C, Gowal S, et al. Challenges of real-world reinforcement learning: definitions, benchmarks and analysis. *Mach Learn* 2021;110:2419–68. <https://doi.org/10.1007/s10994-021-05961-4>.

Ebrahimian H, Barmayoon S, Mohammadi M, Ghadimi N. The price prediction for the energy market based on a new method. *Econ Res Istraživanja* 2018;31:313–37. <https://doi.org/10.1080/1331677X.2018.1429291>.

Elshaafi H, Vinyals M, Grimaldi I, Davy S. Secure Automated Home Energy Management in Multi-Agent Smart Grid Architecture. *Technol Econ Smart Grids Sustain Energy* 2018;3:4. <https://doi.org/10.1007/s40866-018-0042-0>.

European Commission. Stepping up Europe's 2030 climate ambition: Investing in a climate-neutral future for the benefit of our people. Brussels, Belgium: 2020.

European Commission. European Green Deal. Brussels, Belgium: 2019a.

European Commission. Artificial Intelligence: Commission Takes Forward its Work on Ethics Guidelines. 2019b.

European Commission. Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions: A Policy Framework for Climate and Energy in the Period from 2020 to 2030. Brussels, Belgium: 2014.

European Commission, Directorate-General for Energy. Clean energy for all Europeans. Publications Office; 2019. <https://doi.org/doi/10.2833/21366>.

European Parliament. Directive 2019/944 on Common Rules for the Internal Market for Electricity. *Off J Eur Union* 2019;18.

European Parliament. Directive 2018/2001 on the Promotion of the Use of Energy from Renewable Sources. *Off J Eur Union* 2018;61:128.

Faheem M, Shah SBH, Butt RA, Raza B, Anwar M, Ashraf MW, et al. Smart grid communication and information technologies in the perspective of Industry 4.0: Opportunities and challenges. *Comput Sci Rev* 2018;30:1–30. <https://doi.org/10.1016/J.COSREV.2018.08.001>.

Faia R, Pinto T, Vale Z, Corchado JM. A Local Electricity Market Model for DSO Flexibility Trading. *Int. Conf. Eur. Energy Mark. EEM*, vol. 2019- September, IEEE Computer Society; 2019. <https://doi.org/10.1109/EEM.2019.8916563>.

Fang X, Misra S, Xue G, Yang D. Smart grid - The new and improved power grid: A survey. *IEEE Commun Surv Tutor* 2012;14:944–80. <https://doi.org/10.1109/SURV.2011.101911.00087>.

Farhangi H. The path of the smart grid. *IEEE Power Energy Mag* 2010;8:18–28. <https://doi.org/10.1109/MPE.2009.934876>.

Faria P, Spínola J, Vale Z. Methods for Aggregation and Remuneration of Distributed Energy Resources. *Appl Sci* 2018, Vol 8, Page 1283 2018;8:1283. <https://doi.org/10.3390/APP8081283>.

Faria P, Vale Z. Demand response in electrical energy supply: An optimal real time pricing approach. *Energy* 2011;36:5374–84. <https://doi.org/10.1016/j.energy.2011.06.049>.

Ferrag MA, Maglaras LA, Janicke H, Jiang J, Shu L. A systematic review of data protection and privacy preservation schemes for smart grid communications. *Sustain Cities Soc* 2018;38:806–35. <https://doi.org/10.1016/J.SCS.2017.12.041>.

Flach P. Machine learning: the art and science of algorithms that make sense of data. Cambridge University Press; 2012.

François-Lavet V, Henderson P, Islam R, Bellemare MG, Pineau J. An Introduction to Deep Reinforcement Learning. *Found Trends® Mach Learn* 2018;11:219–354. <https://doi.org/10.1561/22000000071>.

Frieden D, Roberts J, Gubina AF. Overview of emerging regulatory frameworks on collective self-consumption and energy communities in Europe. *Int Conf Eur Energy Mark EEM* 2019;2019-September. <https://doi.org/10.1109/EEM.2019.8916222>.

Fujimoto S, Van Hoof H, Meger D. Addressing Function Approximation Error in Actor-Critic Methods. *35th Int Conf Mach Learn ICML* 2018 2018;4:2587–601. <https://doi.org/10.48550/arxiv.1802.09477>.

Ghorashi SM, Rastegar M, Senemmar S, Seifi AR. Optimal design of reward-penalty demand response programs in smart power grids. *Sustain Cities Soc* 2020;60:102150.

<https://doi.org/10.1016/J.SCS.2020.102150>.

Gianey HK, Choudhary R. Comprehensive Review On Supervised Machine Learning Algorithms. Proc - 2017 Int Conf Mach Learn Data Sci MLDS 2017 2018;2018-January:38–43. <https://doi.org/10.1109/MLDS.2017.11>.

Giordano A, Ghiani E, Pilo F, Rosetti L. Planning of Energy Production and Management of Energy Resources in Local Energy Communities: The Case of Berchidda Municipality (Italy) † 2019:5–7. <https://doi.org/10.3390/proceedings2019020016>.

Gomes L, Vale Z. Energy Predictions for System on a Chip Solutions. Adv. Intell. Syst. Comput., Springer, Cham; 2022, p. 243–50. https://doi.org/10.1007/978-3-030-87869-6_23.

Gomes L, Vale Z, Corchado JM. Microgrid management system based on a multi-agent approach: An office building pilot. Meas J Int Meas Confed 2020a;154:107427. <https://doi.org/10.1016/j.measurement.2019.107427>.

Gomes L, Vale Z, Corchado JM. Multi-Agent Microgrid Management System for Single-Board Computers: A Case Study on Peer-to-Peer Energy Trading. IEEE Access 2020b;8:64169–83. <https://doi.org/10.1109/ACCESS.2020.2985254>.

Goncalves C, Barreto R, Faria P, Gomes L, Vale Z. Dataset of an Energy Community's Consumption and Generation with Appliance Allocation for One Year 2022. <https://doi.org/10.5281/ZENODO.6778401>.

Gridwise Architecture Council. GridWise Transactive Energy Framework Version 1.0. 2015.

Gruber TR. A translation approach to portable ontology specifications. Knowl Acquis 1993;5:199–220. <https://doi.org/10.1006/KNAC.1993.1008>.

Guerrero J, Gebbran D, Mhanna S, Chapman AC, Verbič G. Towards a transactive energy system for integration of distributed energy resources: Home energy management, distributed optimal power flow, and peer-to-peer energy trading. Renew Sustain Energy Rev 2020;132:110000. <https://doi.org/10.1016/J.RSER.2020.110000>.

Gui EM, MacGill I. Typology of future clean energy communities: An exploratory structure, opportunities, and challenges. Energy Res Soc Sci 2018;35:94–107. <https://doi.org/10.1016/J.ERSS.2017.10.019>.

Gunduz MZ, Das R. Cyber-security on smart grid: Threats and potential solutions. Comput Networks 2020;169:107094. <https://doi.org/10.1016/J.COMNET.2019.107094>.

Han B, Ren Z, Wu Z, Zhou Y, Peng J. Off-Policy Reinforcement Learning with Delayed Rewards 2021.

Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nat 2020 5857825 2020;585:357–62. <https://doi.org/10.1038/s41586-020-2649-2>.

Hasanuzzaman Shawon M, Muyeen SM, Ghosh A, Islam SM, Baptista MS. Multi-agent systems in ICT enabled smart grid: A status update on technology framework and applications. IEEE Access 2019;7:97959–73. <https://doi.org/10.1109/ACCESS.2019.2929577>.

Heaslip E, Costello GJ, Lohan J. Assessing Good-Practice Frameworks for the Development of Sustainable Energy Communities in Europe: Lessons from Denmark and Ireland. J Sustain Dev Energy, Water Environ Syst 2016;[4]:[307]–[319]. <https://doi.org/10.13044/J.SDEWES.2016.04.0024>.

Hebrail G, Barard A. Individual household electric power consumption Data Set. UCI Mach Learn Repos Irvine, CA Univ California, Sch Inf Comput Sci 2012;1:1–3. <https://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption> (accessed October 21, 2022).

Hernandez-Leal P, Kartal B, Taylor ME. A survey and critique of multiagent deep reinforcement learning. Auton Agent Multi Agent Syst 2019;33:750–97. <https://doi.org/10.1007/S10458-019-09421-1/FIGURES/7>.

Hinton G, Sejnowski TJ. Unsupervised Learning: Foundations of Neural Computation. The MIT Press; 1999. <https://doi.org/10.7551/mitpress/7011.001.0001>.

Horling B, Lesser V. A survey of multi-agent organizational paradigms. *Knowl Eng Rev* 2004;19:281–316. <https://doi.org/10.1017/S0269888905000317>.

Hossain E, Khan I, Un-Noor F, Sikander SS, Sunny MSH. Application of Big Data and Machine Learning in Smart Grid, and Associated Security Concerns: A Review. *IEEE Access* 2019;7:13960–88. <https://doi.org/10.1109/ACCESS.2019.2894819>.

Huang Q, Amin W, Umer K, Gooi HB, Eddy FYS, Afzal M, et al. A review of transactive energy systems: Concept and implementation. *Energy Reports* 2021;7:7804–24. <https://doi.org/10.1016/J.EGYR.2021.05.037>.

Huang X, Wu L, Ye Y. A Review on Dimensionality Reduction Techniques. <https://doi.org/10.1142/S0218001419500174> 2019;33. <https://doi.org/10.1142/S0218001419500174>.

Imani MH, Ghadi MJ, Ghavidel S, Li L. Demand Response Modeling in Microgrid Operation: a Review and Application for Incentive-Based and Time-Based Programs. *Renew Sustain Energy Rev* 2018;94:486–99. <https://doi.org/10.1016/J.RSER.2018.06.017>.

Inshakov O V., Bogachkova LY, Popkova EG. The Transformation of the Global Energy Markets and the Problem of Ensuring the Sustainability of Their Development. *Lect Notes Networks Syst* 2019;44:135–48. https://doi.org/10.1007/978-3-319-90966-0_10.

Iria J, Soares F. Real-time provision of multiple electricity market products by an aggregator of prosumers. *Appl Energy* 2019a;255:113792. <https://doi.org/10.1016/J.APENERGY.2019.113792>.

Iria J, Soares F. A cluster-based optimization approach to support the participation of an aggregator of a larger number of prosumers in the day-ahead energy market. *Electr Power Syst Res* 2019b;168:324–35. <https://doi.org/10.1016/j.epsr.2018.11.022>.

Jan SU, Lee YD, Koo IS. A distributed sensor-fault detection and diagnosis framework using machine learning. *Inf Sci (Ny)* 2021;547:777–96. <https://doi.org/10.1016/J.INS.2020.08.068>.

Jordehi AR. Optimisation of demand response in electric power systems, a review. *Renew Sustain Energy Rev* 2019;103:308–19. <https://doi.org/10.1016/J.RSER.2018.12.054>.

Kaelbling LP, Littman ML, Moore AW. Reinforcement Learning: A Survey. *J Artif Intell Res* 1996;4:237–85. <https://doi.org/10.1613/JAIR.301>.

Kakran S, Chanana S. Smart operations of smart grids integrated with distributed generation: A review. *Renew Sustain Energy Rev* 2018;81:524–35. <https://doi.org/10.1016/J.RSER.2017.07.045>.

Karim MR. *Machine Learning with Scala Quick Start Guide*. Packt; 2019.

Kaur R, Schaye C, Thompson K, Yee DC, Zilz R, Sreenivas RS, et al. Machine learning and price-based load scheduling for an optimal IoT control in the smart and frugal home. *Energy AI* 2021;3:100042. <https://doi.org/10.1016/J.EGYAI.2020.100042>.

Kersch S, Arboleya P. The key role of aggregators in the energy transition under the latest European regulatory framework. *Int J Electr Power Energy Syst* 2022;134:107361. <https://doi.org/10.1016/J.IJEPES.2021.107361>.

Kimani K, Oduol V, Langat K. Cyber security challenges for IoT-based smart grid networks. *Int J Crit Infrastruct Prot* 2019;25:36–49. <https://doi.org/10.1016/J.IJCIP.2019.01.001>.

Koirala BP, Araghi Y, Kroesen M, Ghorbani A, Hakvoort RA, Herder PM. Trust, awareness, and independence: Insights from a socio-psychological factor analysis of citizen knowledge and participation in community energy systems. *Energy Res Soc Sci* 2018;38:33–40. <https://doi.org/10.1016/J.ERSS.2018.01.009>.

Kolluri RR, De Hoog J. Adaptive Control Using Machine Learning for Distributed Storage in Microgrids. *E-Energy 2020 - Proc 11th ACM Int Conf Futur Energy Syst* 2020;7:509–15. <https://doi.org/10.1145/3396851.3402122>.

Kotzur L, Markewitz P, Robinius M, Stolten D. Impact of different time series aggregation methods on

optimal energy system design. *Renew Energy* 2018;117:474–87. <https://doi.org/10.1016/j.renene.2017.10.017>.

Kowalska-Pyzalska A. What makes consumers adopt to innovative energy services in the energy market? A review of incentives and barriers. *Renew Sustain Energy Rev* 2018;82:3570–81. <https://doi.org/10.1016/J.RSER.2017.10.103>.

Kulkarni V, Kulkarni M, Pant A. Quantum computing methods for supervised learning. *Quantum Mach Intell* 2021;3:23. <https://doi.org/10.1007/s42484-021-00050-0>.

Kumar P, Lin Y, Bai G, Paverd A, Dong JS, Martin A. Smart Grid Metering Networks: A Survey on Security, Privacy and Open Research Issues. *IEEE Commun Surv Tutor* 2019;21:2886–927. <https://doi.org/10.1109/COMST.2019.2899354>.

Kumar S, Kumar U. Java Agent Development Framework. *Int J Res* 2014;1:1022–5.

Lazaridou A, Peysakhovich A, Baroni M. Multi-agent cooperation and the emergence of (natural) language. 5th Int. Conf. Learn. Represent. ICLR 2017, International Conference on Learning Representations, ICLR; 2017.

Lee JG, Shim H. Design of Heterogeneous Multi-agent System for Distributed Computation. *Lect Notes Control Inf Sci* 2022;488:83–108. https://doi.org/10.1007/978-3-030-74628-5_4/COVER.

Lengyel A, Botta-Dukát Z. Silhouette width using generalized mean—A flexible method for assessing clustering efficiency. *Ecol Evol* 2019;9:13231–43. <https://doi.org/10.1002/ece3.5774>.

Li J, Zhang C, Xu Z, Wang J, Zhao J, Zhang YJA. Distributed transactive energy trading framework in distribution networks. *IEEE Trans Power Syst* 2018;33:7215–27. <https://doi.org/10.1109/TPWRS.2018.2854649>.

Liang E, Liaw R, Moritz P, Nishihara R, Fox R, Goldberg K, et al. RLlib: Abstractions for Distributed Reinforcement Learning. 35th Int Conf Mach Learn ICML 2018 2017;7:4768–80. <https://doi.org/10.48550/arxiv.1712.09381>.

Liang H, Ma J. Data-Driven Resource Planning for Virtual Power Plant Integrating Demand Response Customer Selection and Storage. *IEEE Trans Ind Informatics* 2022;18:1833–44. <https://doi.org/10.1109/TII.2021.3068402>.

Lin S, Li F, Tian E, Fu Y, Li D. Clustering Load Profiles for Demand Response Applications. *IEEE Trans Smart Grid* 2019;10:1599–607. <https://doi.org/10.1109/TSG.2017.2773573>.

Lu X, Li K, Xu H, Wang F, Zhou Z, Zhang Y. Fundamentals and business model for resource aggregator of demand response in electricity markets. *Energy* 2020;204:117885. <https://doi.org/10.1016/J.ENERGY.2020.117885>.

Macieira P, Gomes L, Vale Z, Ascione F. Energy Management Model for HVAC Control Supported by Reinforcement Learning. *Energies* 2021, Vol 14, Page 8210 2021;14:8210. <https://doi.org/10.3390/EN14248210>.

Mahela OP, Khosravy M, Gupta N, Khan B, Alhelou HH, Mahla R, et al. Comprehensive overview of multi-agent systems for controlling smart grids. *CSEE J Power Energy Syst* 2022;8:115–31. <https://doi.org/10.17775/CSEEJPES.2020.03390>.

Marnay C, Chatzivasileiadis S, Abbey C, Iravani R, Joos G, Lombardi P, et al. Microgrid evolution roadmap. Proc - 2015 Int Symp Smart Electr Distrib Syst Technol EDST 2015 2015:139–44. <https://doi.org/10.1109/SEDST.2015.7315197>.

Martinez-Pabon M, Eveleigh T, Tanju B. Smart Meter Data Analytics for Optimal Customer Selection in Demand Response Programs. *Energy Procedia* 2017;107:49–59. <https://doi.org/10.1016/J.EGYPRO.2016.12.128>.

Marutho D, Hendra Handaka S, Wijaya E, Muljono. The Determination of Cluster Number at k-Mean Using Elbow Method and Purity Evaluation on Headline News. Proc - 2018 Int Semin Appl Technol Inf Commun

Creat Technol Hum Life, ISemantic 2018 2018:533–8. <https://doi.org/10.1109/ISEMANTIC.2018.8549751>.

Masad D, Kazil J. Mesa: An Agent-Based Modeling Framework. Proc 14th Python Sci Conf 2015:51–8. <https://doi.org/10.25080/MAJORA-7B98E3ED-009>.

McArthur SDJ, Davidson EM, Catterson VM, Dimeas AL, Hatziargyriou ND, Ponci F, et al. Multi-agent systems for power engineering applications - Part I: Concepts, approaches, and technical challenges. IEEE Trans Power Syst 2007a;22:1743–52. <https://doi.org/10.1109/TPWRS.2007.908471>.

McArthur SDJ, Davidson EM, Catterson VM, Dimeas AL, Hatziargyriou ND, Ponci F, et al. Multi-agent systems for power engineering applications - Part II: Technologies, standards, and tools for building multi-agent systems. IEEE Trans Power Syst 2007b;22:1753–9. <https://doi.org/10.1109/TPWRS.2007.908472>.

Mehrabi N, Morstatter F, Saxena N, Lerman K, Galstyan A. A Survey on Bias and Fairness in Machine Learning. ACM Comput Surv 2021;54. <https://doi.org/10.1145/3457607>.

Meira J, Carneiro J, Bolón-Canedo V, Alonso-Betanzos A, Novais P, Marreiros G. Anomaly Detection on Natural Language Processing to Improve Predictions on Tourist Preferences. Electron 2022, Vol 11, Page 779 2022;11:779. <https://doi.org/10.3390/ELECTRONICS11050779>.

Miller T. Explanation in artificial intelligence: Insights from the social sciences. Artif Intell 2019;267:1–38. <https://doi.org/10.1016/J.ARTINT.2018.07.007>.

Mitrovic D, Ivanovic M, Budimac Z, Vidakovic M. An overview of agent mobility in heterogeneous environments. WASA, 2011, p. 52–8.

Moncecchi M. Renewable Energy Communities. Politecnico di Milano, 2021.

Monteiro N, Rossetti R, Campos P, Kokkinogenis Z. A Framework for a Multimodal Transportation Network: An Agent-based Model Approach. Transp Res Procedia 2014;4:213–27. <https://doi.org/10.1016/j.trpro.2014.11.017>.

Montoya Giraldo OD. On Linear Analysis of the Power Flow Equations for DC and AC Grids with CPLs. IEEE Trans Circuits Syst II Express Briefs 2019;66:2032–6. <https://doi.org/10.1109/TCSII.2019.2894149>.

Moradmand A, Dorostian M, Shafai B. Energy scheduling for residential distributed energy resources with uncertainties using model-based predictive control. Int J Electr Power Energy Syst 2021;132:107074. <https://doi.org/10.1016/J.IJEPES.2021.107074>.

Moretti M, Djomo SN, Azadi H, May K, De Vos K, Van Passel S, et al. A systematic review of environmental and economic impacts of smart grids. Renew Sustain Energy Rev 2017;68:888–98. <https://doi.org/10.1016/J.RSER.2016.03.039>.

Mota B, Albergaria M, Pereira H, Silva J, Gomes L, Vale Z, et al. Climatization and luminosity optimization of buildings using genetic algorithm, random forest, and regression models. Energy Informatics 2021;4. <https://doi.org/10.1186/s42162-021-00151-x>.

Naeem M, Rizvi STH, Coronato A. A Gentle Introduction to Reinforcement Learning and its Application in Different Fields. IEEE Access 2020;8:209320–44. <https://doi.org/10.1109/ACCESS.2020.3038605>.

Ndousse KK, Eck D, Levine S, Jaques N. Emergent social learning via multi-agent reinforcement learning. Int. Conf. Mach. Learn., 2021, p. 7991–8004.

Nepal, Yamaha, Sahashi, Yokoe. Analysis of Building Electricity Use Pattern Using K-Means Clustering Algorithm by Determination of Better Initial Centroids and Number of Clusters. Energies 2019;12:2451. <https://doi.org/10.3390/en12122451>.

Neto G. From single-agent to multi-agent reinforcement learning: Foundational concepts and methods. Learn Theory Course 2005;2. <https://doi.org/10.13140/RG.2.2.16116.78727>.

Nguyen TT, Nguyen ND, Nahavandi S. Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications. IEEE Trans Cybern 2020;50:3826–39. <https://doi.org/10.1109/TCYB.2020.2977374>.

Oh KK, Park MC, Ahn HS. A survey of multi-agent formation control. *Automatica* 2015;53:424–40. <https://doi.org/10.1016/j.automatica.2014.10.022>.

Olivares DE, Mehrizi-Sani A, Etemadi AH, Cañizares CA, Iravani R, Kazerani M, et al. Trends in microgrid control. *IEEE Trans Smart Grid* 2014;5:1905–19. <https://doi.org/10.1109/TSG.2013.2295514>.

Oliveira P, Pinto T, Morais H, Vale Z. MASGriP a multi-agent smart grid simulation platform. *IEEE Power Energy Soc. Gen. Meet.*, 2012. <https://doi.org/10.1109/PESGM.2012.6345649>.

OMIE. MIBEL day-ahead hourly market prices 2022. <https://www.omie.es/en/market-results/daily/daily-market/daily-hourly-price> (accessed October 21, 2022).

Ottesen SØ, Tomasgard A, Fleten S-E. Multi market bidding strategies for demand side flexibility aggregators in electricity markets. *Energy* 2018;149:120–34.

Ourahou M, Ayrir W, EL Hassouni B, Haddi A. Review on smart grid control and reliability in presence of renewable energies: Challenges and prospects. *Math Comput Simul* 2020;167:19–31. <https://doi.org/10.1016/J.MATCOM.2018.11.009>.

Padakandla S. A Survey of Reinforcement Learning Algorithms for Dynamically Varying Environments. *ACM Comput Surv* 2021;54. <https://doi.org/10.1145/3459991>.

Palanca J, Terrasa A, Julian V, Carrascosa C. Spade 3: Supporting the new generation of multi-agent systems. *IEEE Access* 2020;8:182537–49. <https://doi.org/10.1109/ACCESS.2020.3027357>.

Park CK, Kim HJ, Kim YS. A study of factors enhancing smart grid consumer engagement. *Energy Policy* 2014;72:211–8. <https://doi.org/10.1016/J.ENPOL.2014.03.017>.

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine learning in Python. *J Mach Learn Res* 2011;12:2825–30.

Pereira H, Faia R, Gomes L, Faria P, Vale Z. Incentive-based and Price-based Demand Response to Prevent Congestion in Energy Communities. 2022 IEEE Int Conf Environ Electr Eng 2022 IEEE Ind Commer Power Syst Eur (EEEIC / I&CPS Eur 2022a):1–6. <https://doi.org/10.1109/EEEIC/ICPSEUROPE54979.2022.9854648>.

Pereira H, Gomes L, Faria P, Vale Z, Coelho C. Web-based platform for the management of citizen energy communities and their members. *Energy Informatics* 2021;4:43. <https://doi.org/10.1186/s42162-021-00155-7>.

Pereira H, Gomes L, Morais H, Vale Z. Dataset of 30 energy customers with flexibility data, and distributed generation, considering residential, small commerce, large commerce, and industrial customers 2022b. <https://doi.org/10.5281/ZENODO.6783289>.

Pereira H, Ribeiro B, Gomes L, Vale Z. CECOS: a Centralized Management Platform Supported by Distributed Services to Represent and Manage Resources Aggregation Entities and its End-users in a Smart Grid Context. *IFAC-PapersOnLine*, vol. 55, Elsevier; 2022c, p. 309–14. <https://doi.org/10.1016/J.IFACOL.2022.07.054>.

Pinto T, Praça I, Vale Z, Santos C. Collaborative Reinforcement Learning of Energy Contracts Negotiation Strategies. *Commun Comput Inf Sci* 2019;1047:202–10. https://doi.org/10.1007/978-3-030-24299-2_17.

Ponds KT, Arefi A, Sayigh A, Ledwich G. Aggregator of Demand Response for Renewable Integration and Customer Engagement: Strengths, Weaknesses, Opportunities, and Threats. *Energies* 2018, Vol 11, Page 2391 2018;11:2391. <https://doi.org/10.3390/EN11092391>.

Poplavskaya K, de Vries L. Aggregators today and tomorrow: from intermediaries to local orchestrators? Behind Beyond M Digit Aggregation, *Optim Monet* 2020:105–35. <https://doi.org/10.1016/B978-0-12-819951-0.00005-0>.

Poslad S, Charlton P. Standardizing Agent Interoperability: The FIPA Approach. In: Luck M, Mařík V, Štěpánková O, Trapp R, editors. *Multi-Agent Syst. Appl. ACAI 2001*, Springer, Berlin, Heidelberg; 2001, p. 98–117. https://doi.org/10.1007/3-540-47745-4_5.

Pouyanfar Samira, Sadiq S, Yan Yilin, Tian Haiman, Reyes Maria Presa, Chen Shu-Ching, et al. A Survey on

- Deep Learning. *ACM Comput Surv* 2018;51:92. <https://doi.org/10.1145/3234150>.
- Purlu M, Turkyay BE. Estimating the Distributed Generation Unit Sizing and Its Effects on the Distribution System by Using Machine Learning Methods. *Elektron Ir Elektrotehnika* 2021;27:24–32. <https://doi.org/10.5755/J02.EIE.28864>.
- Qiu D, Wang Jianhong, Wang Junkai, Strbac G. Multi-Agent Reinforcement Learning for Automated Peer-to-Peer Energy Trading in Double-Side Auction Market. *IJCAI Int Jt Conf Artif Intell* 2021;3:2913–20. <https://doi.org/10.24963/IJCAI.2021/401>.
- Qureshi KN, Alhudhaif A, Jeon G. Electric-vehicle energy management and charging scheduling system in sustainable cities and society. *Sustain Cities Soc* 2021;71:102990. <https://doi.org/10.1016/J.SCS.2021.102990>.
- Radl J, Fleischhacker A, Revheim FH, Lettner G, Auer H. Comparison of Profitability of PV Electricity Sharing in Renewable Energy Communities in Selected European Countries. *Energies* 2020, Vol 13, Page 5007 2020;13:5007. <https://doi.org/10.3390/EN13195007>.
- Rahimi F, Albuyeh F. Applying lessons learned from transmission open access to distribution and grid-edge Transactive Energy systems. 2016 IEEE Power Energy Soc Innov Smart Grid Technol Conf ISGT 2016 2016. <https://doi.org/10.1109/ISGT.2016.7781236>.
- Rahman MA, Asyhari AT. The Emergence of Internet of Things (IoT): Connecting Anything, Anywhere. *Comput* 2019, Vol 8, Page 40 2019;8:40. <https://doi.org/10.3390/COMPUTERS8020040>.
- Rahman MS, Isherwood N, Oo AMT. Multi-agent based coordinated protection systems for distribution feeder fault diagnosis and reconfiguration. *Int J Electr Power Energy Syst* 2018;97:106–19. <https://doi.org/10.1016/j.ijepes.2017.10.031>.
- Rahmani R, Moser I, Cricenti A, Karimi H. Decentralised Multi-Agent based Demand Response for Smart Grid with Inclusion of Green Data Centre. 2018 26th Int Conf Software, Telecommun Comput Networks, SoftCOM 2018 2018:385–92. <https://doi.org/10.23919/SOFTCOM.2018.8555837>.
- Rajabi A, Eskandari M, Ghadi MJ, Li L, Zhang J, Siano P. A comparative study of clustering techniques for electrical load pattern segmentation. *Renew Sustain Energy Rev* 2020;120:109628. <https://doi.org/10.1016/J.RSER.2019.109628>.
- Ramos D. Reinforcement Learning of a Multi-Agent System for the Forecasting of Electricity Consumption. *Faculdade de Economia da Universidade do Porto*, 2021.
- Ramos D, Faria P, Gomes L, Campos P, Vale Z. A Learning Approach to Improve the Selection of Forecasting Algorithms in an Office Building in Different Contexts, Springer, Cham; 2022, p. 271–81. https://doi.org/10.1007/978-3-031-16474-3_23.
- Ramos D, Khorram M, Faria P, Vale Z. Load Forecasting in an Office Building with Different Data Structure and Learning Parameters. *Forecast* 2021, Vol 3, Pages 242-255 2021;3:242–55. <https://doi.org/10.3390/FORECAST3010015>.
- Rathor SK, Saxena D. Energy management system for smart grid: An overview and key issues. *Int J Energy Res* 2020;44:4067–109. <https://doi.org/10.1002/ER.4883>.
- Le Ray G, Pinson P. The ethical smart grid: Enabling a fruitful and long-lasting relationship between utilities and customers. *Energy Policy* 2020;140:111258. <https://doi.org/10.1016/J.ENPOL.2020.111258>.
- Reback J, jbrockmendel, McKinney W, Bossche J Van den, Roeschke M, Augspurger T, et al. pandas-dev/pandas: Pandas 1.4.3 2022. <https://doi.org/10.5281/ZENODO.6702671>.
- Recht B. A Tour of Reinforcement Learning: The View from Continuous Control. *Annu Rev Control Robot Auton Syst* 2019;2:253–79. <https://doi.org/10.1146/annurev-control-053018-023825>.
- Rehmani MH, Reisslein M, Rachedi A, Erol-Kantarci M, Radenkovic M. Integrating Renewable Energy Resources into the Smart Grid: Recent Developments in Information and Communication Technologies. *IEEE Trans Ind Informatics* 2018;14:2814–25. <https://doi.org/10.1109/TII.2018.2819169>.

Reis IFG, Gonçalves I, Lopes MAR, Antunes CH. A multi-agent system approach to exploit demand-side flexibility in an energy community. *Util Policy* 2020;67:101114. <https://doi.org/10.1016/j.jup.2020.101114>.

Ribó-Pérez D, Larrosa-López L, Pecondón-Tricas D, Alcázar-Ortega M. A Critical Review of Demand Response Products as Resource for Ancillary Services: International Experience and Policy Recommendations. *Energies* 2021, Vol 14, Page 846 2021;14:846. <https://doi.org/10.3390/EN14040846>.

Roberts J, Frieden D, d'Herbemont S. Energy community definitions. *Deliv Dev Under Scope Compil Proj Integr Community Power Energy Islands* 2019.

Robu V, Flynn D, Andoni M, Mokhtar M. Consider ethical and social challenges in smart grid research. *Nat Mach Intell* 2019 112 2019;1:548–50. <https://doi.org/10.1038/s42256-019-0120-6>.

Romero Agüero J, Takayesu E, Novosel D, Masiello R. Grid modernization: challenges and opportunities. *Electr J* 2017;30:1–6. <https://doi.org/10.1016/J.TEJ.2017.03.008>.

Rosaci D, Sarné GML. Cloning mechanisms to improve agent performances. *J Netw Comput Appl* 2013;36:402–8. <https://doi.org/10.1016/J.JNCA.2012.04.018>.

Russell S, Norvig P. *Artificial intelligence: a modern approach* 2002.

Sachs JD, Schmidt-Traub G, Mazzucato M, Messner D, Nakicenovic N, Rockström J. Six Transformations to achieve the Sustainable Development Goals. *Nat Sustain* 2019 29 2019;2:805–14. <https://doi.org/10.1038/s41893-019-0352-9>.

Saha B. Green computing. *Int J Comput Trends Technol* 2014;14:46–50.

Santodomingo R, Rohjans S, Uslar M, Rodríguez-Mondéjar JA, Sanz-Bobi MA. Ontology matching system for future energy smart grids. *Eng Appl Artif Intell* 2014;32:242–57. <https://doi.org/10.1016/J.ENGAPPAI.2014.02.005>.

Saravanan R, Sujatha P. A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification. *Proc 2nd Int Conf Intell Comput Control Syst ICI CCS 2018* 2019:945–9. <https://doi.org/10.1109/ICCONS.2018.8663155>.

Vom Scheidt F, Staudt P, Weinhardt C. Assessing the Economics of Residential Electricity Tariff Selection. *SEST 2019 - 2nd Int Conf Smart Energy Syst Technol* 2019. <https://doi.org/10.1109/SEST.2019.8849143>.

Schmarje L, Santarossa M, Schröder S-M, Koch R. A survey on Semi-, Self- and Unsupervised Learning for Image Classification 2020. <https://doi.org/10.1109/access.2021.3084358>.

Schram W, Louwen A, Lampropoulos I, Van Sark W. Comparison of the Greenhouse Gas Emission Reduction Potential of Energy Communities. *Energies* 2019, Vol 12, Page 4440 2019;12:4440. <https://doi.org/10.3390/EN12234440>.

Seyedzadeh S, Rahimian FP, Glesk I, Roper M. Machine learning for estimation of building energy consumption and performance: a review. *Vis Eng* 2018 61 2018;6:1–20. <https://doi.org/10.1186/S40327-018-0064-7>.

Shahriar S, Al-Ali AR, Osman AH, Dhou S, Nijim M. Machine learning approaches for EV charging behavior: A review. *IEEE Access* 2020;8:168980–93. <https://doi.org/10.1109/ACCESS.2020.3023388>.

Shaik MA. Protecting Agents from Malicious Hosts using Trusted Platform Modules (TPM). *Proc Int Conf Inven Commun Comput Technol ICICCT* 2018 2018:559–64. <https://doi.org/10.1109/ICICCT.2018.8473278>.

Shakeri M, Pasupuleti J, Amin N, Rokonzaman M, Low FW, Yaw CT, et al. An Overview of the Building Energy Management System Considering the Demand Response Programs, Smart Strategies and Smart Grid. *Energies* 2020, Vol 13, Page 3299 2020;13:3299. <https://doi.org/10.3390/EN13133299>.

Shalev-Shwartz S, Ben-David S. *Understanding Machine Learning*. Cambridge: Cambridge University Press; 2014. <https://doi.org/10.1017/CBO9781107298019>.

- Shehory O, Sycara K, Chalasani P, Jha S. Agent cloning: An approach to agent mobility and resource allocation. *IEEE Commun Mag* 1998;36:58–67. <https://doi.org/10.1109/35.689632>.
- Shin J, Badgwell TA, Liu KH, Lee JH. Reinforcement Learning – Overview of recent progress and implications for process control. *Comput Chem Eng* 2019;127:282–94. <https://doi.org/10.1016/J.COMPCHENG.2019.05.029>.
- Shukla OJ, Soni G, Kumar R, Ahluwalia RS. A Review of Multi Agent-based Production Scheduling in Manufacturing System. *Recent Patents Eng* 2021;15. <https://doi.org/10.2174/1872212114999200423112955>.
- Siano P, De Marco G, Rolan A, Loia V. A Survey and Evaluation of the Potentials of Distributed Ledger Technology for Peer-to-Peer Transactive Energy Exchanges in Local Energy Markets. *IEEE Syst J* 2019;13:3454–66. <https://doi.org/10.1109/JSYST.2019.2903172>.
- Silva C, Faria P, Vale Z. Rating the Participation in Demand Response Programs for a More Accurate Aggregated Schedule of Consumers after Enrolment Period. *Electron* 2020, Vol 9, Page 349 2020;9:349. <https://doi.org/10.3390/ELECTRONICS9020349>.
- Silva C, Faria P, Vale Z. Demand Response and Distributed Generation Remuneration Approach Considering Planning and Operation Stages. *Energies* 2019;12:2721. <https://doi.org/10.3390/en12142721>.
- Sinaga KP, Yang MS. Unsupervised K-means clustering algorithm. *IEEE Access* 2020;8:80716–27. <https://doi.org/10.1109/ACCESS.2020.2988796>.
- Sloane M, Moss E. AI's social sciences deficit. *Nat Mach Intell* 2019 18 2019;1:330–1. <https://doi.org/10.1038/s42256-019-0084-6>.
- Son YB, Im JH, Kwon HY, Jeon SY, Lee MK. Privacy-preserving peer-to-peer energy trading in blockchain-enabled smart grids using functional encryption. *Energies* 2020;16. <https://doi.org/10.3390/en13061321>.
- SRI International. Open Agent Architecture Homepage 2022. <http://www.ai.sri.com/~oaa/> (accessed October 21, 2022).
- Steriotis K, Tsaousoglou G, Efthymiopoulos N, Makris P, Varvarigos E (Manos). A novel behavioral real time pricing scheme for the active energy consumers' participation in emerging flexibility markets. *Sustain Energy, Grids Networks* 2018;16:14–27. <https://doi.org/10.1016/j.segan.2018.05.002>.
- Sutton RS, Barto AG. Reinforcement Learning: An Introduction. 2nd ed. London, England: The MIT Press; 2018.
- Tagliapietra S, Zachmann G, Edenhofer O, Glachant JM, Linares P, Loeschel A. The European union energy transition: Key priorities for the next five years. *Energy Policy* 2019;132:950–4. <https://doi.org/10.1016/J.ENPOL.2019.06.060>.
- Teichgraeber H, Brandt AR. Clustering methods to find representative periods for the optimization of energy systems: An initial framework and comparison. *Appl Energy* 2019;239:1283–93. <https://doi.org/10.1016/j.apenergy.2019.02.012>.
- The European Consumer Organisation. Electricity Aggregators: Starting off on the right foot with consumers. Brussels, Belgium: 2018.
- Thimmapuram PR, Jinho Kim, Botterud A, Youngwoo Nam. Modeling and simulation of price elasticity of demand using an agent-based model. 2010 Innov. Smart Grid Technol., IEEE; 2010, p. 1–8. <https://doi.org/10.1109/ISGT.2010.5434739>.
- Turner L, Scheidler A, Schafer F, Menke J-H, Dollichon J, Meier F, et al. Pandapower—An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems. *IEEE Trans Power Syst* 2018;33:6510–21. <https://doi.org/10.1109/TPWRS.2018.2829021>.
- Triantafyllou A, Jimenez JAP, Torres ADR, Lagkas T, Rantos K, Sarigiannidis P. The Challenges of Privacy and Access Control as Key Perspectives for the Future Electric Smart Grid. *IEEE Open J Commun Soc*

2020;1:1934–60. <https://doi.org/10.1109/OJCOMS.2020.3037517>.

Uddin M, Romlie MF, Abdullah MF, Abd Halim S, Abu Bakar AH, Chia Kwang T. A review on peak load shaving strategies. *Renew Sustain Energy Rev* 2018;82:3323–32. <https://doi.org/10.1016/j.rser.2017.10.056>.

Uludag S, Zeadally S, Badra M. Techniques, Taxonomy, and Challenges of Privacy Protection in the Smart Grid 2015:343–90. https://doi.org/10.1007/978-3-319-08470-1_15.

Varghese AC, Padmini V, Kumar G, Khaparde SA. Smart Grid Consumer Behavioral Model using Machine Learning. *Int Conf Innov Smart Grid Technol ISGT Asia 2018* 2018:734–9. <https://doi.org/10.1109/ISGT-ASIA.2018.8467824>.

Vázquez-Canteli JR, Nagy Z. Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Appl Energy* 2019;235:1072–89. <https://doi.org/10.1016/j.apenergy.2018.11.002>.

Voigt P, dem Bussche A. The EU General Data Protection Regulation (GDPR): A Pratical Guide. vol. 10. 1st ed. Springer; 2017.

Wang S, Duan J, Shi D, Xu C, Li H, Diao R, et al. A Data-Driven Multi-Agent Autonomous Voltage Control Framework Using Deep Reinforcement Learning. *IEEE Trans Power Syst* 2020;35:4644–54. <https://doi.org/10.1109/TPWRS.2020.2990179>.

Wang S, Xu J, Zhang N, Liu Y. A Survey on Service Migration in Mobile Edge Computing. *IEEE Access* 2018;6:23511–28. <https://doi.org/10.1109/ACCESS.2018.2828102>.

Wang X, Wang C, Xu T, Guo L, Li P, Yu L, et al. Optimal voltage regulation for distribution networks with multi-microgrids. *Appl Energy* 2018;210:1027–36. <https://doi.org/10.1016/j.apenergy.2017.08.113>.

Wang Z, Hong T. Reinforcement learning for building controls: The opportunities and challenges. *Appl Energy* 2020;269:115036. <https://doi.org/10.1016/j.apenergy.2020.115036>.

Wei Y, Zhang X, Shi Y, Xia L, Pan S, Wu J, et al. A review of data-driven approaches for prediction and classification of building energy consumption. *Renew Sustain Energy Rev* 2018;82:1027–47. <https://doi.org/10.1016/j.rser.2017.09.108>.

Weiss G. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. vol. 3. The MIT Press; 1999.

Woltmann S, Kittel J. Development and implementation of multi-agent systems for demand response aggregators in an industrial context. *Appl Energy* 2022;314:118841. <https://doi.org/10.1016/j.apenergy.2022.118841>.

Wooldridge M. An introduction to multiagent systems. John Wiley & Sons; 2009.

Wooldridge M, Jennings NR. Intelligent agents: theory and practice. *Knowl Eng Rev* 1995;10:115–52. <https://doi.org/10.1017/S0269888900008122>.

Working Group on Intelligent Data Mining and Analysis (IDMA). Open Data Sets 2022. <https://site.ieee.org/pes-iss/data-sets/> (accessed October 21, 2022).

Wu L, Wang Z. Research on Top-k Association Rules Mining Algorithm Based on Clustering. *J. Phys. Conf. Ser.*, vol. 1682, IOP Publishing; 2020, p. 12064. <https://doi.org/10.1088/1742-6596/1682/1/012064>.

Wu YK, Chang SM, Hu YL. Literature Review of Power System Blackouts. *Energy Procedia* 2017;141:428–31. <https://doi.org/10.1016/j.egypro.2017.11.055>.

Xiao J, Lu J, Li X. Davies Bouldin Index based hierarchical initialization K-means. *Intell Data Anal* 2017;21:1327–38. <https://doi.org/10.3233/IDA-163129>.

Xie J, Liu C-C. Multi-agent systems and their applications. *J Int Counc Electr Eng* 2017;7:188–97. <https://doi.org/10.1080/22348972.2017.1348890>.

Xu D, Tian Y. A Comprehensive Survey of Clustering Algorithms. *Ann Data Sci* 2015 22 2015;2:165–93.

<https://doi.org/10.1007/S40745-015-0040-1>.

Yan X, Ozturk Y, Hu Z, Song Y. A review on price-driven residential demand response. *Renew Sustain Energy Rev* 2018;96:411–9. <https://doi.org/10.1016/J.RSER.2018.08.003>.

Ye D, Zhang M, Vasilakos A V. A survey of self-organization mechanisms in multiagent systems. *IEEE Trans Syst Man, Cybern Syst* 2017;47:441–61. <https://doi.org/10.1109/TSMC.2015.2504350>.

Ye Y, Qiu D, Sun M, Papadaskalopoulos D, Strbac G. Deep Reinforcement Learning for Strategic Bidding in Electricity Markets. *IEEE Trans Smart Grid* 2020;11:1343–55. <https://doi.org/10.1109/TSG.2019.2936142>.

Ye Y, Tang Y, Wang H, Zhang XP, Strbac G. A Scalable Privacy-Preserving Multi-Agent Deep Reinforcement Learning Approach for Large-Scale Peer-to-Peer Transactive Energy Trading. *IEEE Trans Smart Grid* 2021;12:5185–200. <https://doi.org/10.1109/TSG.2021.3103917>.

Ying X. An Overview of Overfitting and its Solutions. *J Phys Conf Ser* 2019;1168:022022. <https://doi.org/10.1088/1742-6596/1168/2/022022>.

Yu L, Qin S, Zhang M, Shen C, Jiang T, Guan X. A Review of Deep Reinforcement Learning for Smart Building Energy Management. *IEEE Internet Things J* 2021;8:12046–63. <https://doi.org/10.1109/JIOT.2021.3078462>.

Zame KK, Brehm CA, Nitica AT, Richard CL, Schweitzer GD. Smart grid and energy storage: Policy recommendations. *Renew Sustain Energy Rev* 2018;82:1646–54. <https://doi.org/10.1016/J.RSER.2017.07.011>.

Zebari R, Abdulazeez A, Zeebaree D, Zebari D, Saeed J. A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *J Appl Sci Technol Trends* 2020;1:56–70.

Zhang Y, Huang T, Bompard EF. Big data analytics in smart grids: a review. *Energy Informatics* 2018;1:8. <https://doi.org/10.1186/s42162-018-0007-5>.

Zhou L, Pan S, Wang J, Vasilakos A V. Machine learning on big data: Opportunities and challenges. *Neurocomputing* 2017;237:350–61. <https://doi.org/10.1016/j.neucom.2017.01.026>.

Zhu XJ. Semi-supervised learning literature survey. University of Wisconsin-Madison Department of Computer Sciences; 2005.

Zia MF, Benbouzid M, Elbouchikhi E, Muyeen SM, Techato K, Guerrero JM. Microgrid transactive energy: Review, architectures, distributed ledger technologies, and market analysis. *IEEE Access* 2020;8:19410–32. <https://doi.org/10.1109/ACCESS.2020.2968402>.