UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dept. of Computer Systems and Computation

Multi-agent Generic Market Simulator for Virtual Markets using SPADE

Master's Thesis

Master's Degree in Artificial Intelligence, Pattern Recognition and Digital Imaging

AUTHOR: Barbas Rebollo, David

Tutor: Botti Navarro, Vicente Juan

Cotutor: Giret Boggino, Adriana Susana

ACADEMIC YEAR: 2021/2022

Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València

Multi-Agent Generic Market Simulation for Virtual Markets using SPADE

MASTER'S THESIS

Master's Degree in Artificial Intelligence, Pattern Recognition and Digital Imaging

Author: David Barbas Rebollo

Tutor: Vicente Juan Botti Navarro
Adriana Giret Boggino

Course 2021-2022

Acknowledgements

First of all, I would like to thank the VRAIN for giving the opportunity to develop this work with them. But I would especially like to express my gratitude towards my tutors, Vicente J. Botti Navarro and Adriana Giret Boggino, for their guidance during the development of this project, offering their support and experience whenever necessary.

I would also like to thank the Institute of Energy Engineering for allowing this collaboration. I am especially grateful to have had the chance to work with Carlos Álvarez Bel and Carlos Sanchís Gómez to further develop this project.

Finally, I would like to thank my family for all the unconditional support offered throughout my university studies and for teaching me that with hard work and dedication, anything is possible.

Abstract

Nowadays, negotiations between entities are undergoing a substantial change, as it is rapidly transitioning into a virtual and automated environment. On top of that, there is an increasing demand for market simulation tools. Therefore, multi-agent technologies are becoming essential in different types of market simulations, specially in electricity, financial and e-commerce markets. These technologies are used to assist the design of negotiating agents and how different strategies affect the results obtained. They allow faithful simulations of real-life negotiation scenarios. This, which enables to carry out experimentation with different behaviors and strategies in order to obtain deeper insights and a better understanding of the results.

State-of-the-art market simulation models have evolved over the last few years. These have progressed from periodic behavioral methods to neural-based models using artificial neural networks and reinforcement learning. To date, the vast majority of the research in this area has focused on the design of negotiation protocols. However, lately, studies have shifted their interest to the creation of generic simulation frameworks. Despite this, there is no published research related to generic virtual market simulations.

Generic simulation frameworks are one of the best tools to study virtual markets. As stated previously, there is also an increasing demand for market simulation tools. However, until recently, tailored simulators have been used for almost every research study. This ultimately is an inefficient solution. Alternatively, using a parameterizable generic tool will reduce financial costs and development time. Moreover, sandbox like tools have shown that eliminating restrictions allows a better experimentation to reach deeper insights, which cannot be obtained in regular restrained market re-creations. Therefore, exploring the development of an accessible application for generic virtual market simulations would be a remarkable contribution to this area. On top of that, simulators have poor visualization tools to analyze agent interaction. Proof of this is that nearly all frameworks show agent communication via log registers. That is why it would be interesting to explore friendlier tools for this matter.

Therefore, this work provides a novel solution in this application area, as its main aim is to develop a generic virtual market simulator using multi-agent systems. In addition, a visualization tool is created for the system. For the reasons stated above, this study does not aim to design optimal interaction protocols but to create the first milestone that allows researchers to parametrize an environment to their needs regardless of the type of market studied. To this end, our tool is based on SPADE, an open-source platform for multi-agent simulation. Hence, the simulator's main features consist of: (1) a 2-step negotiation process (an initial bilateral negotiation followed up by an auction), (2) a customizable negotiation behavior, (3) optional restrictions over the maximum number of exchanged items and (4) a web visualizer showing the interactions and results of the negotiations.

Furthermore, this work is evaluated with a real-life case study provided by the Institute of Energy Engineering of the *Universitat Politècnica de València*. This group defined several case scenarios in an electricity market in order to examine the emergent behavior in the different simulations. Finally, results show that this model can be considered the starting point for the successful development of a more complex system to study the behavior in virtual markets.

Key words: multi-agent system, agent-based simulation, generic market simulation, virtual market, automatic negotiation, auction, bilateral negotiation

Resumen

Actualmente, en el ámbito de las negociaciones entre entidades es importante reseñar que se está produciendo un cambio sustancial, puesto que se está evolucionando hacia un entorno virtual y automatizado. No sólo eso, sino que, además, existe una creciente demanda de herramientas para la simulación de mercado. Es, por tanto, que los sistemas multi-agente están convirtiéndose en un instrumento imprescindible dentro de este sector, sobre todo en el mercado eléctrico, financiero y de comercio electrónico. La función de esta tecnología es ayudar en el diseño de los diferentes agentes de negociación, así como estudiar de qué manera diferentes estrategias pueden alterar el resultado final. Asimismo, simulan verazmente escenarios de negociación de la vida real. Esto permite probar diferentes comportamiento y estrategias con el fin de obtener un mayor conocimiento y una mejor compresión de los resultados alcanzados.

A lo largo de los últimos años, el estado del arte de los modelos de simulación de mercado ha evolucionado de métodos conductuales periódicos a modelos basados en redes neuronales artificiales y aprendizaje por refuerzo. Hasta la fecha, el grueso de la investigación en esta área se ha centrado en el diseño de protocolos de negociación. Sin embargo, en los últimos años, los estudios han mostrado un interés creciente en la creación de plataformas de simulación genéricas. Aun así, a día de hoy, no se han encontrado publicaciones relacionadas con *frameworks* para la simulación genérica de mercados virtuales.

Una de las mejores herramientas para estudiar los mercados virtuales son las plataformas de simulación genéricas. Tal y como se expuso previamente, la demanda de herramientas de simulación de mercado está en aumento. Sin embargo, hasta hace poco, en la totalidad de las investigaciones realizadas se han utilizado simuladores hechos a medida. Cabe destacar que se trata de solución ineficiente y, por este motivo, es importante reseñar que utilizar una herramienta genérica parametrizable sería una alternativa a tener en cuenta, ya que reduciría tanto el tiempo de desarrollo como el coste. Además, las herramientas de tipo *sandbox* han demostrado que eliminar restricciones permite realizar una experimentación más original, los cuales no podrían obtenerse de recreaciones de mercado restringidas. Es por ello, que desarrollar una aplicación accesible para simulaciones genéricas de mercados virtuales sería una contribución notable dentro de este ámbito. Aparte de ello, resulta importante tener en cuenta que los simuladores poseen herramientas poco eficientes para visualizar interacciones entre agentes. Una prueba de ello es que la mayoría de los *frameworks* muestran la comunicación entre agentes a través de registros *log*. Por este motivo, sería interesante explorar opciones más accesibles.

Por tanto, este trabajo propone una solución novedosa en esta área de aplicación, ya que su principal objetivo es desarrollar un simulador de mercado virtual genérico utilizando sistemas multi-agente. Además, se pretende crear una herramienta de visualización del sistema. Por las razones expuestas en el párrafo anterior, la finalidad de este estudio no es diseñar protocolos de actuación óptimos, sino permitir a los investigadores parametrizar un entorno según sus necesidades, independientemente del tipo de mercado estudiado. Con este fin, se ha diseñado una herramienta basada en SPADE, que es una plataforma de código abierto para la simulación multi-agente. Las características principales de este simulador se exponen a continuación: (1) un proceso de negociación dividido en dos etapas (una negociación bilateral seguida de una subasta), (2) un comportamiento de negociación personalizable, (3) restricciones opcionales sobre el número máximo de artículos intercambiados y (4) un visualizador web que muestra las interacciones y los resultados de las negociaciones.

En este sentido, se pone en práctica este modelo a través de un estudio real, proporcionado por el Instituto de Ingeniería Energética de la *Universitat Politècnica de València*.

Se definieron varios escenarios dentro un mercado eléctrico con la finalidad de examinar el comportamiento derivado de las diferentes simulaciones. En definitiva, los resultados obtenidos muestran que este modelo puede ser considerado el punto de partida para el desarrollo de un sistema de mayor complejidad que permita estudiar el comportamiento de los mercados virtuales.

Palabras clave: sistema multi-agente, simulación basada en agentes, simulación de mercados genéricos, mercado virtual negociación automática, subasta, negociación bilateral

Resum

Actualment, a l'àmbit de les negociacions entre entitats és important ressenyar que s'està produint un canvi substancial, ja que s'està evolucionant cap a un entorn virtual i automatitzat. No només això, sinó que, a més, hi ha una demanda creixent d'eines per a la simulació de mercat. És, per tant, que els sistemes multiagent s'estan convertint en un instrument imprescindible dins aquest sector, sobretot en el mercat elèctric, financer i de comerç electrònic. La funció daquesta tecnologia és ajudar en el disseny dels diferents agents de negociació, així com estudiar de quina manera diferents estratègies poden alterar el resultat final. Així mateix, simulen veraçment escenaris de negociació de la vida real. Això permet provar diferents comportament i estratègies per tal d'obtenir un coneixement més gran i una millor compressió dels resultats assolits.

Al llarg dels darrers anys, l'estat de l'art dels models de simulació de mercat ha evolucionat de mètodes conductuals periòdics a models basats en xarxes neuronals artificials i aprenentatge per reforç. Fins ara, el gruix de la investigació en aquesta àrea s'ha centrat en el disseny de protocols de negociació. En els darrers anys, però, els estudis han mostrat un interès creixent en la creació de plataformes de simulació genèriques. Tot i així, avui dia, no s'han trobat publicacions relacionades amb *frameworks* per a la simulació genèrica de mercats virtuals.

Una de les millors eines per estudiar els mercats virtuals són les plataformes de simulació genèriques. Tal com es va exposar prèviament, la demanda de ferramentes de simulació de mercat està en augment. Tot i això, fins fa poc, en la totalitat de les investigacions realitzades s'han utilitzat simuladors fets a mida. Cal destacar que es tracta de solució ineficient i, per això, és important ressenyar que utilitzar una eina genèrica parametritzable seria una alternativa a tenir en compte, ja que reduiria tant el temps de desenvolupament com el cost. A més, les eines de tipus sandbox han demostrat que eliminar restriccions permet fer una experimentació més original, els quals no podrien obtenir-se de recreacions de mercat restringides. És per això que desenvolupar una aplicació accessible per a simulacions genèriques de mercats virtuals seria una contribució notable dins aquest àmbit. A banda d'això, és important tenir en compte que els simuladors tenen eines poc eficients per visualitzar interaccions entre agents. Una prova és que la majoria dels *frameworks* mostren la comunicació entre agents a través de registres *log*. Per això, seria interessant explorar opcions més accessibles.

Per tant, aquest treball proposa una solució nova en aquesta àrea d'aplicació, ja que el seu objectiu principal és desenvolupar un simulador de mercat virtual genèric utilitzant sistemes multiagent. A més, es pretén crear una eina de visualització del sistema. Per les raons exposades al paràgraf anterior, la finalitat d'aquest estudi no és dissenyar òptims protocols d'actuació, sinó permetre als investigadors parametritzar un entorn segons les seves necessitats, independentment del tipus de mercat estudiat. Amb aquesta finalitat, s'ha dissenyat una eina basada en SPADE, que és una plataforma de codi obert per a la simulació multiagent. Les principals característiques d'aquest simulador s'exposen a continuació: (1) un procés de negociació dividit en dues etapes (una negociació bilateral seguida d'una subhasta), (2) un comportament de negociació personalitzable, (3) restriccions opcionals sobre el nombre màxim d'articles intercanviats i (4) un visualitzador web que mostra les interaccions i els resultats de les negociacions.

En aquest sentit, es posa en pràctica aquest model a través d'un estudi real, proporcionat per l'Institut d'Enginyeria Energètica de la Universitat Politècnica de València. Es van definir diversos escenaris dins d'un mercat elèctric amb la finalitat d'examinar el comportament derivat de les diferents simulacions. En definitiva, els resultats obtinguts mostren que aquest model es pot considerar el punt de partida per al desenvolupament

d'un sistema de més complexitat que permeti estudiar el comportament dels mercats virtuals.

Paraules clau: sistema multiagent, simulació basada en agents, simulació de mercats genèrics, mercat virtual, negociació automàtica, subhasta, negociació bilateral

Contents

List of Figures

List of Tables

CHAPTER 1
Introduction

This work studies the viability of developing a generic multi-agent market simulation with the SPADE platform. It also studies the viability of validating the simulator in a real-life case scenario provided by the Institute of Energy Engineering of the Polytechnic University of Valencia.

This chapter explores the motivation which led to carry out the project and the problems that were tackled during the process. Afterwards, the scope and the objectives of this study are described. To conclude this chapter, a guideline of the thesis is outlined.

1.1 Motivation

Agent-based simulation has been a popular technique for modeling and analyzing different scenarios in recent years. Simulators provide an environment to faithfully re-enact real world scenarios. This is extremely helpful in order to test several hypotheses without taking any risk, since they are tested in a controlled and, sometimes, a virtual environment. Therefore, simulators are a fundamental step before applying gained knowledge outside of the experimentation environment. This is due to the fact that researchers can analyze the interactions that have taken place in the system and gain further insights about their hypotheses. Some examples of the current most popular modeled scenarios include self-driving cars, epidemic spreading and virtual markets.

Multi-agent systems are perhaps the best fit to simulate dynamic and adaptive environments as well as actors with complex behaviors who can interact with other actors and/or the environment. As a result, they are one the most used tools to model negotiations between agents. Nowadays, negotiations between entities are rapidly becoming virtual and automated. Therefore, multi-agent system technologies becomes a fundamental tool to create virtual market simulators. Chapter 3 will introduce and display the fundamentals about multi-agent systems to understand an carry out this thesis.

Furthermore, before the implementation of any multi-agent system important to study the structure of organizations. An organization is a collection of entities working together to achieve a goal. This concept is an effective tool for activity coordination. That is why, organization theory would be useful in order to implemented a multi-agent system. In specific, virtual markets, a subset of virtual organizations, contains most key concepts in order to design the structure of this work's developed system. Therefore, chapter 2 dives into these concepts which will be valuable later in chapter 4 when designing the proposed system.

Lately, there has been an increase in demand for market simulation tools. This is evidenced by the increase of papers published related to virtual markets, specially electricity, financial and e-commerce markets. This is because, simulation tools allow to researchers to test different strategies and negotiating protocols in specified environments. Therefore, is a faster and safer way to gain information about the interactions and results in a market.

Research over the last years has mainly centered on the design of agents and negotiation protocols. However, lately, some researchers have shifted their focus to the creation of generic simulation frameworks. Generic simulations have various advantages. Having a generic environment for research offer a more efficient solution than having to develop a tailored system for every project. In addition, it has been proven that sandbox like tools could result in a more creative use of a platform. Ultimately, reaching deeper insights they could have not obtained in regular restrained market re-creations. Despite this benefits, there is no published research related to generic virtual market simulations.

This paper aims to provide the first milestone for generic virtual market simulations which could be a key investigation in this area. Throughout this project we propose an accessible application with parameterizable attributes for a generic market. On top of that, simulators have poor visualization tools to analyze agent interaction. Proof of this is that most frameworks show agent communication via log registers. That is why the tool developed will also explore a friendlier approach to this matter.

1.2 Scope and objectives

The scope of this thesis is to develop a generic multi-agent simulator that can provide information about the different interactions regardless the market's type. To this end, this simulator provides an initial platform as a sandbox like tool to offer a flexible and expressive experimentation software. This is the reason why this work's contribution is an initial platform for generic virtual markets. This platform will showcase its customization potential and offer visualization tools to study the interactions that have taken place during the simulation negotiation phases.

Moreover, the purpose of this work is not to carry out an exhaustive experimentation of negotiation protocols. To date, the vast majority of research in this area has focused on the design of negotiation protocols, so studying in depth this field would not be a remarkable contribution to the literature available. Instead, this simulator aims to create the first milestone that allows researchers to parametrize an environment to their needs independent of the type of market studied.

The specific goals of this work can be summarized as:

- Study and analyze the requirements of virtual markets for a generic market simulation.
- Study and use the SPADE platform to create the simulation environment.
- Define a multi-agent infrastructure in order to simulate generic virtual markets.
- Implement an initial prototype that allows the correct interaction between seller and buyer agents.
- Validate the presented solution via tests and using a real-life case study.
- Develop a web visualizer to analyze the interactions that took place in a simulation.

1.3 Thesis outline

This document is structured in the following way:

- **Chapter 1**: establishes the motivations that have led to the development of this study and introduces the goals to be fulfilled throughout the work.
- **Chapter 2**: introduces the basic information every reader should have in order to fully understand this work.
- **Chapter 3**: describes the essential knowledge about the technologies applied through this study as well as introducing relevant work regarding multi-agent simulation.
- **Chapter 4**: explains the analysis and design of the generic market. In this chapter, all the elements of the developed prototype and the testing performed to ensure that the system works properly will be explained.
- **Chapter 5**: assesses the usefulness of the developed platform in the wild, using a real-life scenario as case study.
- **Chapter 6**: reports the conclusions gathered during the study. In other words, summarizes the facts brought up by this work and proposes future lines of work that may be carried out regarding this topic.

CHAPTER 2
Fundamentals of organizations

Before developing a multi-agent system it is important to study the structure of organizations. An organization is a collection of entities that work together to achieve a specific goal. Therefore, it is considered an effective tool for activity coordination. As a result, organization theory would be useful in the implementation of a multi-agent system.

This chapter delves into the main characteristics of human and agent organizations, as well as, how they function. Additionally, the similarities and differences between both organizations will be described. Also, the parallelism between both of them and the principal factors that affect the modeling of human organization will be explained. Furthermore, the concept of an organization as an open system will be detailed. To conclude this chapter, the advantages organizations are able to offer to design multi-agent systems will be emphasized.

2.1 Concept of organization

Organizations have been an effective tool for activity coordination, not only for humans but also for agents. To be more specific, over the years the concept of an organization has increased its relevance in the area of multi-agent systems given the ease it offers to analyze and design coordination and collaboration mechanisms. Moreover, having this structural resource makes more straightforward the analysis and design of a multi-agent system. Therefore, developers focus on the concepts of structure, rules, roles, etc. in order to guide the development process and simplify it.

This concept is commonly used as a way to model agent coordination and ensure social order in an application [20]. However, this has only taken place because over decades humans have developed the fundamentals and mechanisms to organize themselves towards coordination and task distribution between themselves.

2.1.1. Human organizations

In order to better comprehend how to model a multi-agent system as an organization, the focus should turn to the concepts of how human organizations are formed. The Oxford English Dictionary defines an organization as "a group of people who form a business, club, etc. together in order to achieve a particular aim".

Other examples of the definition of human organizations in the literature are: "the formation of a social entity with a specific number of members which can be differentiated internally by the tasks carried out" [53]; "the coordination of all activities performed by the individuals which make up a company. Their objective is to maximize their profit given all their resources, material, technical and human resources available" [70].

Given these definitions, it is possible to describe an organization as an ensemble of individuals which handle different and specific activities or functions. Additionally, individuals are structured by given rules. By these means, they are able to achieve the organization's objectives. These are commonly known by all individuals, who work together to accomplish them [53]. Moreover, organizations are required to formally determine the possible actionable tasks, the standard at which they have to be completed and in which context they can be carried out.

Therefore, it is possible to outline the features that constitue a human organization as the following [27]:

- Organizations are made up of people.
- The aims of an organization must be clear to perform. Its members carry out activities which lead to these objectives by coordination and control mechanisms.
- Work is clearly divided between different individuals. This is achieved via specialization and task allocation.
- It requires a formally defined structure, with established roles (independent of organization members), responsibilities associated to each role and pre-established relations between other organization members.
- All activities must be related to the global objectives of the organization. An activity is related to a role, hence a role may only exist if it contributes to achieve an activity.
- It has defined limits, which establish who the members of the organization are. These can be specified previously or defined by the activities they perform.

2.1.2. Agent organizations

Also known and referred to as agencies, an organization in a multi-agent system is used to describe a collection of agents. Broadly speaking, these agents coordinate themselves to achieve the global system's goals through predefined roles and interaction protocols.

The existing organization theory [16] studies and analyses human organizations. Given this theory, it determines the basic concepts, relations and features of each existing type of organization. Human organizations are remarkably efficient to manage the coordination between its members and the accomplishing their aims. For that reason, this concept is adopted to form agent organizations.

Experts describe agent organizations as structured systems. This means they are able to divide themselves into different units and/or groups. It is important to note that each of these subdivisions exist independently.

Agent organizations are labeled structures based upon how the activity between agents is performed. Agents have an assigned role, an expected behavior and defined authority relations with other agents. As a result, the structure of an organization relies on how agents naturally exist instead of how it is explicitly defined.

Additionally, the position of an agent in an organization is determined by his or her role. Human organizations can be composed of partners, directors, managers, supervisors and workers. Every member of an organization must have the gist of each role, their aims and the hierarchical relations between them. So, likewise agent entities also have their responsibilities to carry out and achieve given specific rules.

Another way of describing agent organizations is to interprete them as a collection of behavioral restrictions a group of agents accepts in order to control their autonomous behavior. The purpose of this is to fulfill the main goals in a more efficient way [34].

In summary, the properties of agent organizations can be described as:

- It is composed by several agents (software, physical and/or human). Agents can have independent characteristics and objectives between themselves.
- Common and global aims are pursued. These goals do not directly depend on an individual agent's targets and desires.
- The workload of the organization is divided between the agents. Task allocation is performed by roles. These roles describe the activities carried out and their functionality inside the agency.
- The agency can be partitioned into groups or individuals. These gatherings of agents are then able to interact with each other.
- Well defined limits are described. These are determined by the environment, internal and external agents, as well as the functionality offered and the service provided.

  These are determined by the organization's environment, internal and external agents

2.1.3. Comparison between organizations

By comparing the characteristics of both organizations, it is important to outline that they both share similar qualities. This is no surprise given the fact that, agent developments have been implemented from the simulation and adaptation of human organizations. Therefore, it is logical to think that extending the understanding of these organizations will enable a better analysis, design and implementation of agent organizations.

As a result, in future sections it will be possible to see a parallelism between human and agent organizations. Furthermore, the new methods, systems, procedures and strategies that will be described for human organizations will be translated into agent organizations in order to test new courses of action to coordinate a group of agents.

2.2 Organization factors

When studying and analyzing an organization is important to take into account the entities which take part in it. In other words, it is necessary to consider the pursued objectives, the functionality of the system, the environment where relationships are nurtured and the rules which limit the behavior of all the elements that belong to the organization. Consequently, the mentioned ideas will be explained in further detail.

2.2.1. Structure

The structure is defined as the abstraction of individuals into roles, groups of entities, the dependencies and the patterns used to interact between themselves. In essence, it encompasses the static elements of the organization. It is independent to the fact that individuals may leave the organization and new ones may join it. Moreover, a key element to organizations are groups. These encapsulate a limited number of entities with common interactions, values (to some degree) and shared rules to follow [53].

The fundamental elements that ensemble an organization's structure are [27]:

- Specialization: in order to increase efficiency, an agent is assigned a reduced set of tasks. These tasks depend other their role. It is important to note that specialization is directly proportional to how repetitive a job it is.
- Departmentalization: consists of grouping jobs so common tasks can be easily coordinated. It can be accomplished by product or service, geographical location, processes performed or depending on the client or clients they are working with.
- Line of command: represents the authority relationships between hierarchies and their correct way of interaction.
- Level of centralization: quantifies how the decisions should be performed. They can be carried out by a reduced number of partners, between all the individuals conforming the organization or across all management levels.
- Formalization: indicates to what degree roles are standardized and explicitly specified.

Specifically, in agent organizations, their structure is achieved by the means of roles and groups. Roles represent the activities they are able to perform, alongside with their limitations, objectives and how they are going to try and achieve them. Meanwhile, groups define the context where the different activities take place. It is important to note that communication is performed inside those groups, therefore it also specifies the social reach of an agent [20]. Furthermore, with subgroups it is possible to specify the authority relationships between them.

2.2.2. Functionality

An organization must be able to state their global goals, and that is the reason why specifying the services and functionality offered is so important. As a result, the concept of subgoals arise in order to clarify and specify the tasks that must be performed in order to reach the global goals. Moreover, this clarifies who the possible clients or users of this system are.

Nowadays, it is fairly common to see that organizations have a mission statement. This is to help understand the organization's members and external entities what the final goal is and how they are working towards its achievement. Additionally, as time passes it also makes it less likely to side track with irrelevant activities. As management employees change, it may result in losing sight of the organization's initial goal. Furthermore, having a mission statement makes it easy to asses whether an organization is completing or not their objectives. This allows to measure their performance and counteract appropriately.

Given an organization has clearly stated their mission statement, now it is essential to clarify how their goals are going to be achieved. Therefore, the basic tasks for its completion should be created. The complexity of this task depends on dividing an objective into categories, subcategories and finally into atomic actionable actions. By gathering these actions as collections of very simple tasks it is possible to maximize productivity and efficiency, while minimizing costs [53].

Similarly, agent organizations also need clear global objectives which will determine the general function of the system. Roles and groups will have their own aims which establish the actions to complete the global objectives. Therefore, the organization's aims are split into intro sub-objectives that will characterize the agent's behaviors. In this way, roles will adapt to specific targets that will need to be achieved.

Taking this into account, models based on UML diagrams [35] offer the perfect system to model the cohesion between different roles and groups of agents in an organization while including specific interactions between themselves.

2.2.3. Normalization

Normalization is the formal declaration of the rules and predefined actions to control the behaviors of the members inside an organization. This includes the obligations, permitted and prohibited actions as well as the corresponding sanctions and rewards.

Human organizations propose standards and procedures to help its members determine how to carry out their tasks. There are four types of normalization [73]:

- Tasks: exactly clarify what someone is expected to do.
- Results: specify what is the expected outcome of a well completed task.
- Skills: determine the competences required to successfully complete a task
- Behaviors: indicate the consensus for the accepted and expected behavior while completed a task.

When analyzing agent organizations, one can see that coordination is achieved through social norms. These guidelines establish what is authorized to do, obligations and prohibitions. They depend on the context and current time an agent is in. The context is given by the agent's role, its group (if it has one) and its specified objectives.

Therefore, in order to ensure agents comply with the dictated social norms, it is required to apply a system that guarantees it. One option would be to have an agent, *issuer*, whose role is the task descrbied above [29]. Another option would be to make agents accept a contract which given certain conditions enables different interactions with associated rules [19]. An alternative option would be to pre-establish sanctions and rewards in order to adjust the agent's different behaviors [43].

2.2.4. Dynamism

Dynamism embraces the fact that organizations cannot be describes as static entities. This means that members are allowed to enter and exit an organization repeatedly during their lifespan. Furthermore, as organizations age their aims may diverge from the initial ones meaning members may leave the organizations as their interest may no longer agree with the organization ones and new members may be recruited.

Within agent organizations, the organization must control who can take part and the roles new agents will undertake. They also are able to expel agents due to an inadequate behavior. Additionally, dynamic organizations also need to consider the fact that they should allow to create and dissolve internal groups of agents.

2.2.5. Environment

As one may think, in human organizations the term environment entails more than its location. Moreover, it encompasses everything outside an organization, such as providers, clients, the competition, financial institutions, investors, governmental organizations, which can regulate their operations, and the labor market, which provides an organization with its employees [73].

Therefore, organizations may not exist unless they are in a suitable environment. They need resources from the outside such as physical materials and members to develop their products or services as well as clients willing to buy them. That is the reason why all opportunities and threats exist outside an organization. For this reason, the environment directly determines if an organization will thrive or fail [27].

On the other hand, it is important to take into account that the term environment entails subtle differences when speaking about multi-agent systems. As a matter of fact, it is associated with the resources, the applications used by agents and how to access them. Agents are able to interact with their environment via perceptive sensors and effectors in order to take actions upon it. This concept will be analyzed in more detail later on.

2.3 Open systems

As discussed previously, organizations are dynamic structures. An open system allows the flow of unknown entities in and out during the organization's lifespan. This requirement means models need to allow exiting the system without disturbances and be ready for external inputs. Furthermore, an organization has to consider that new incorporations, not taken into account in the designing phase, may be formalized [32]. From a multi-agent system perspective, these entities may be human or software agents who join their system and that have not been designed or developed by the same research group.

Most implemented multi-agent systems are considered a closed system. This means all the entities acting in the planned ecosystem are taken into account in the design phase. Additionally, it is assumed that agents are benevolent and trust each other's interactions [78].

On the other hand, regarding open systems, it is important to consider that unknown agents may want to enter the organization. These agents may use different protocols, communication systems and even have a different architecture. Therefore, it is essential to incorporate a middleware to allow the interaction between all agents. Moreover, the intentions of these agents are also unknown. Hence, the requirement of further identification, identity verification or authorization of new entities needs to be taken into consideration [18].

A common auction is an example of an open system. Auctioneers must have previous identification to participate and non-auctioneers may attend the event as bidders. Agents who undertake these roles can be developed by anyone. Therefore an auction needs methods to ensure all agents are going to be able to interact with others properly and that all entities comply with the event's obligations, restrictions and regulations.

To sum up, open systems need to able to homogenize their participants' agent structure in order to be safely included in the system. It is not acceptable to allow the execution of unknown software in an environment without an type of control. This encourages trust between participants, as well as allowing them to coordinate correctly.

2.4 Organization taxonomies

Agent organizations share some structural patterns with human organizations. This section dives into the different types of both kinds of organizations. These patterns will be applied throughout the fourth chapter to model the multi-agent system.

2.4.1. Human organizations

Before describing in depth the different types of human organizations, it is important to note that these are subdivided into two different types: mechanical and organic organizations.

The main characteristics of mechanical organizations are that they have a strict authority and a defined hierarchy. This is achieved via a supervisor-subordinate relationship and a predominant vertical type of communication. Morever, mechanical organizations excel at stable environments. These include [5]:

- Simple structure: these organizations are made of a reduced number of departments, where the decision making is only done by the head of department. Usually, managers need to coordinate a substantial number of employees, leading to a flat pyramidal structure. Additionally, there is little formalization and standardization of the workforce's positions. This type of organization is most suitable for small companies or for crisis situations [58].

  These organizations are easy to maintain, have clear responsibilities for their employees, in addition to excelling at a fast communication, decision making and task execution.However, this model can hinder a business's growth due to the responsibility overload the manager may have. Moreover, the consequences of wrong decision making may be decisive for the organization.

- Bureaucracy: this type of structure was very popular in the 1950's and 1960's. Organizations with a bureaucratic structure are characterized by having repetitive and very specific tasks as well as many formalized rules. Another quality of these type of organizations that that are made up of several departments with specific aims. These departments are small in size, meaning that the manager only needs to coordinate a reduced number of employees. However, this means the amount of hierarchical levels increases. Additionally, authority in these organizations is centralized and decision making follows the chain of command [58].

  Bureaucratic organizations are able to standardized activities efficiently via specialized departments. As groups are formed by specialists, improving communication between employees. Moreover, thanks to the high regulatory policies applied, decision making can be centralized. In spite of these advantages, it is important to state that highly specialized groups may lead to task conflicts between departments. Furthermore, department managers may only focus on their own sub-goals instead of thinking of other department's goals. Due to their high amount of bureaucracy, these organizations struggle in dynamic environments where unpredicted situations arise and there are no established rules no tackle the situation.

- Matrix structure: this type of structure adds a further level of complexity to bureaucratic organizations. For each product or client there is a functional level of departments (sales, marketing, countability, production, R&D, etc.). These means all employees have two managers, a product department manager and a functional department manager. This structure is usually adopted by engineering companies [58].

This type of structure helps small and medium sized businesses to grow safer. It also provides a better flexibility, intra-department communication and coordination, given the fact that conflicts normally arise between common product departments. The major drawback is that with two managers employees have a fuzzier line of command.

Figure 2.1 shows a clear example of this type of organization. In the diagram one can see that project A has one manager for the entire project. This manager has not only to supervise the functioning of the project and but to make sure the project's aims are achieved. Meanwhile, a specific departmental manager, such as the marketing manager, takes care of all the marketing facets of every project in the organization. Thus, each group has several managers.

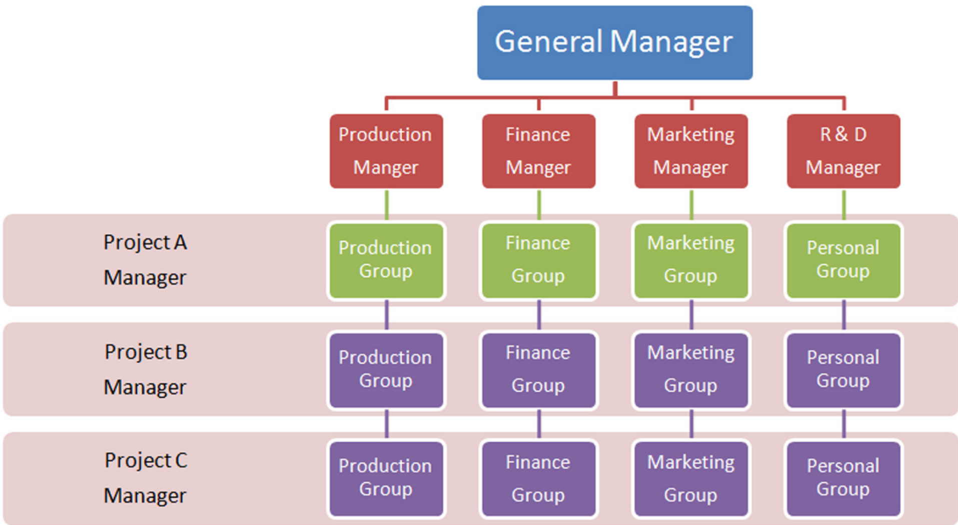

Figure 2.1: Diagram displaying a matrix structured organization.

On the other hand, organic organizations have a more relaxed hierarchy. The control responsibility of the organization is distributed between more members. Also, tasks are distributed among all the department members. Communication is horizontal between members of a department or between different departments leading to faster and more flexible responses. Therefore, this structure excels at dynamic environments which require more adaptability. These include [5]:

- Team structure: this type of organizations remove departmental divisions and decentralize decision making. Therefore, decisions are made inside each team. The organization's global objectives are broken down into and assigned to different groups according to their expertise to solve them. Additionally, team members share all the information regarding a task between themselves. Therefore, coordination is done by the consensual decision making done by the group [58].

  All teams are able to work independently, as they have access to all the necessary resources related to their project. This highlights how departmentalization is removed. This structure also allows employees to switch teams in order to work in the one that suits them the most.

  Usually, these organizations are composed by highly motivated employees which try to exceed the goals given by the organization's administration. Also, teams help to reduce excessive administrative tasks. But before this can be accomplished, this structure requires high administrative preparation and adaptation on behave of the organization. Employees also need to adjust to this collaborative way of working, which requires them to invest more time in meetings in order to facilitate task coordination.

- Adhocracy: the term adhocracy means "absence of hierarchy". Therefore, by definition, this type of organization is the opposite of bureaucracies. These organizations lack a formal structure constituted of specialized multidisciplinary teams grouped by functions. Additionally, all members have equal rights to make decisions on the organization.

  To sum up, one could describe this type of organization as flexible and adaptable, where creativity and innovation is promoted due to its informal structure. Although these organizations are suitable for rapid growth on the early stages of a business, it may not be sustainable for further development. Moreover, it would entail a huge risk to allow any employee to make organization-wide decisions.

- Virtual organizations: these companies are characterized for distributing some of its departments across different external companies. Thus, this organizations adopts a centralized configuration with few internal departments and has the flexibility of choosing who to associate with for the remaining required functionalities [58].

  This system offers great flexibility as it is possible to hire a more proficient staff at a lower cost. It also reduces the core duties of the company as the rest are delegated. However, this implies that it is more difficult to control the organization's external members as they have their own management systems.

  Moreover, a sub-type of virtual organizations are **markets**. This structure is made from several separate organizations which negotiate, through contracts, goods and services. With this model it is not necessary to create new functional branches, instead it is possible to reach an agreement to acquire them. This inherently establishes high competitiveness between organizations.

  Depending on the freeness of a market, the price of the products offered is self-regulated or intervened by an external control authority. Therefore, in a free market, supply and demand will dictate the price of goods and services. However, an external organization, for example a government, can interfere with this model dictating a price floor or ceiling price.

  Specifically, **virtual markets** can be defined as the simulation of a real marketplace where buyers and sellers meet to negotiate transactions via a virtual platform. For instance, this platform can be a web where negotiators may interact with each other. In other words, a virtual marketplace is an environment that allows on-line individuals to showcase their goods and/or services to others, as well as provides a communication channel [40].

  Furthermore, their main difference with traditional markets is their capability of on-line communication. This allows transactions to be faster, eliminates on-site interaction, reduces human intervention, enables the possibility of automation using software agents and reduces paperwork. So, overall it can streamline the communication between sellers and buyers while reducing the cost of each operation [72].

  There are three types of virtual markets ans each one has its own negotiation technique. These are [72]:

  - Catalogs: the seller displays what he offers. In this type of market the price can sometimes be negotiable.

- Auctions: This market consists of 1..n auctioneers and 1..n bidders, who compete for a product. Depending of the type of auction the price of a good changes significantly.
- Price systems: good's price fluctuate according to its offer and demand. An example of a market that follows this type of behavior would be the stock market.

2.4.2. Agent organizations

Although there are no two equal agent organizations it is possible to identify common characteristics and group them into organizational models. They share several similarities with human organizations as these where the base from which they were developed. These include [5]:

- Hierarchies: hierarchical organizations have a pyramidal structure where agents at the top have a global vision of the organization. Agents have limited interaction and communication, only with members in the same level or an adjacent one. Commands travel from top to bottom, while information travels in the inverse direction. Therefore, the higher up an agent is the in a hierarchy, the more information available he has.
- Holarchy: a holarchy is defined as the connection between holons. A holon is close to the idea of a fractal. It is described as something that is simultaneously a whole in and of itself, as well as a part of a larger whole. The holon represents a way to overcome the dichotomy between parts and wholes. In other words, this entity is something that has integrity and identity on itself while being a part of a subsystem and part of a larger system. This model is commonly referred to as a form of hierarchy. As a hierarchy has both a top and bottom level this statement is contradictory with a holarchy's definition. The "hierarchical" relation between holons at different levels can be described as: holons at one level make up holons as a whole or parts of another level.

  This type of organization has been used in the past to model businesses and manufacturing systems. It is easily implemented in systems where objectives can be recursively decomposed into sub-tasks. These are then assigned to individual holons.
- Coalitions: these groups emerge as a result of agents wanting to cooperate with each other in order to achieve a goal that otherwise is unachievable o less likely to be accomplished. This are temporary, and usually short, associations. This is due the fact that, after reaching their aims the coalition dissolves, as there is no longer a meaningful reason to continue with it. Depending on the conditions to set up or join a coalition, they may be disjointed or may allow overlapping with other groups.
- Teams: teams are groups of agents who cooperate between themselves to achieve a common objective. Some systems have an explicit representation of their team's common aims, beliefs and their plan. Unlike coalitions, this group is not dissolved after completing its goal. Teams are created in order to undertake a challenging goal, otherwise unobtainable by a single agent.
- Congregations: this model can be described like a social event. Agents group together to find other agents with whom they can collaborate to increase their benefits. The meeting's objective is to help agents find other agents with some specific abilities or requirements. Congregations usually take place over a long period of

time, where entities can dynamically join or exit, depending on their current desires.

- Federations: in contrast with the previously mentioned organizations, federations have a sole agent, who is responsible for communicating with the external world. Inside a federation agents delegate the task of communication to a single agent who now acts as an intermediary. Additionally, this agent coordinates the group by notifying information and assigning tasks. It also standardizes incoming messages so the whole federation can understand them.
- Matrix structure: similarly, to human organizations, agents in this group have more than one authority figure. In this model, agents have different ways to perform an action. They will choose the one that satisfies the most their manager's preferences. Moreover, problems may arise if their decisions contradict their managers'. When an agent is required to perform two tasks which require the same ability, it will try to perform both if its computational power allows it.

All the agent organizations described above display many similarities with human organizations. In specific, hierarchies, holarchies and matrix structured organizations as they are based on are based on mechanical human organizations (simple structure, bureaucratic and matrix structured organizations).

On the other hand, teams, coalitions, congregations and federations match closely with the examples of organic organizations. Therefore, they can be related to team structured, adhocratic and virtual organization models. Hence, this emphasizes the fact that human organizations are a valuable resource as a baseline and inspiration to design agent organizations for multi-agent systems.

Finally, table 2.1 shows a comparison between the different human organizations previously explained.

Table 2.1: Comparison of the characteristics of the different human organizations.

Organization's characteristics	Human organizations				
	Simple	Bureaucracy	Matrix	Teams	Virtumaeral
Centralized decision making	x	x	x	-	-
Specialized activities	-	x	x	x	-
Generic members	-	-	-	x	x
Departmentalization	-	x	x	-	-
Control division	x	-	x	-	-
Task formalization	-	x	x	-	-
Coordination between experts	-	x	x	x	-
Authority	x	x	x	-	-
Line of command	-	x	x	-	-
Several mangers	-	-	x	-	-
Department objectives	-	x	-	-	-
Organization objectives	x	-	x	x	x
Shared information	-	-	-	x	x
Flexible	-	-	-	x	x
Hire external services	-	-	-	-	x

CHAPTER 3

Multi-agent, Negotiation and Agreement Technologies

Multi-agent systems are the best fit to simulate dynamic and adaptive environments with complex behaviors and interactions among its participants. As a result, they are one the most useful tools to model negotiations between agents. Therefore, understanding multi-agent system technologies is fundamental to create market simulators.

This chapter will start by discussing the basis of multi-agent oriented programming. Then, the concepts of negotiation and agreement technologies will be described. These elements are essential in the development of this project.

Afterwards, the different available frameworks for implementing multi-agent based solutions will be reviewed. Also, the reasons why SPADE platform was chosen to perform this study will be explained. Finally, this chapter will conclude by outlining relevant studies from the same field of work.

3.1 Fundamentals of multi-agent systems

3.1.1. Intelligent agents

Originally, an intelligent agent was a metaphor used to refer to artificial intelligence (AI). However, the definition has evolved since with the introduction of multi-agent systems. Before diving into what a multi-agent system is, it is important to define its key element.

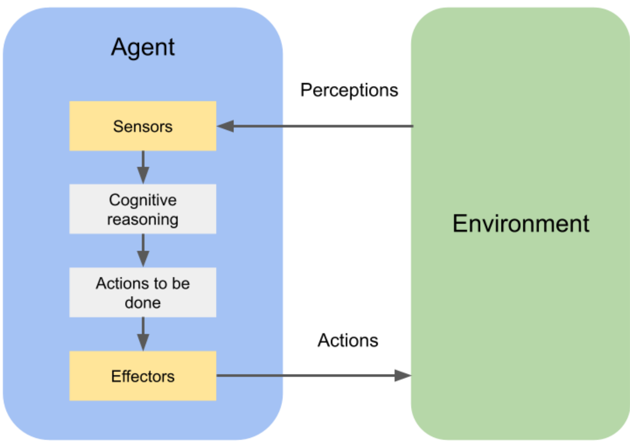

Figure 3.1: Basic diagram of an intelligent agent.

An individual agent is the core element of these systems. It is defined as a software entity which acts autonomously in a given environment [14]. Broadly speaking, it can be summarized as a software program which is able to perform intelligent reasoning and has the abilities required to carry out specific actions. In figure 3.1, one can see the layout of an intelligent agent.

The intelligent behavior of agents arises from the fact they have four essential properties [77]. These are:

- Autonomous: agents perform actions without the need for external command by a sense-plan-act deliberative cycle. This enables them to control their internal state and behavior [28].
- Reactive: agents are capable of reacting to different changes in their environment or other agent's actions.
- Proactive: these entities must be able to have the initiative to achieve or create goals for themselves. Therefore, agents are not only driven by events, but also by the desires they want to accomplish.
- Social: this feature eases the cooperation, coordination and negotiation with other agents to accomplish individual, common or disjoint aims.

Other properties intelligent agents may have are:

- Mobility: the ability to change a communication network.
- Veracity: make a conscious effort to not communicate false information on purpose; benevolence, not having contradictory goals and be trying to fulfill them.
- Rationality: have clear objectives and always try to achieve them.
- Learning: being able to learn from interactions and information obtained from the environment.

Therefore, intelligent agents should not be categorised as being either smart or dumb, but rather as having intellectual capabilities. Taking this into account, the more intelligent a software agent is the more capabilities it will have. Despite this, in certain applications, intelligent agents with limited capabilities will be all that is required [3].

3.1.2. Multi-agent system

There is no single definition for a multi-agent system. Despite this, the most accepted definitions share some common features. As it name states, multi-agent systems are composed by several agent entities. These intelligent agents are able to interact between them in a physical or virtual environment. Interactions can be done indirectly through the environment or directly via some type of message [28].

In this ecosystem we define a *platform* as what allows agents in a system to interact without taking into consideration its features (centralized or not, embedded into the agents or not, etc.). A platform typically assists agents with a set of tasks depending on the system's needs. It is also considered a tool for the agents, thus it does not exhibit an autonomous or proactive behavior [55].

Communication may vary from simpler to more complex ones. However, agents are require to have a common communication language, so they are able to understand each other. For instance, a simple type of communicating information can be signaling. On the contrary, a more complex one would be template messages, where different elements are involved, such as the sender, the receiver, the body message and meta-data. That is why there is a remarkable interest in standardizing their medium of communication. Standards which stand out are:

- Knowledge Query and Manipulation Language (KQML): this was the first standard developed. Messages between agents include performatives (why the message was sent) and several parameters (such as a message, the receptor, the ontology used, etc.) [25]. This simple scheme is enough to implement any communication protocol.

- Foundation for Intelligent Physical Agents - Agent Communication Language (FIPA-ACL): this standard can be defined as an extension of KQML. These messages share the same structure but include more performatives to extend the communication between agents. This removed the need of generating or deleting agent's beliefs and/or desires. For example, this new language increased the possibilites of KQML with performatives like: ACCEPT-PROPOSAL, REJECT-PROPOSAL or NOT-UNDERSTOOD [26].

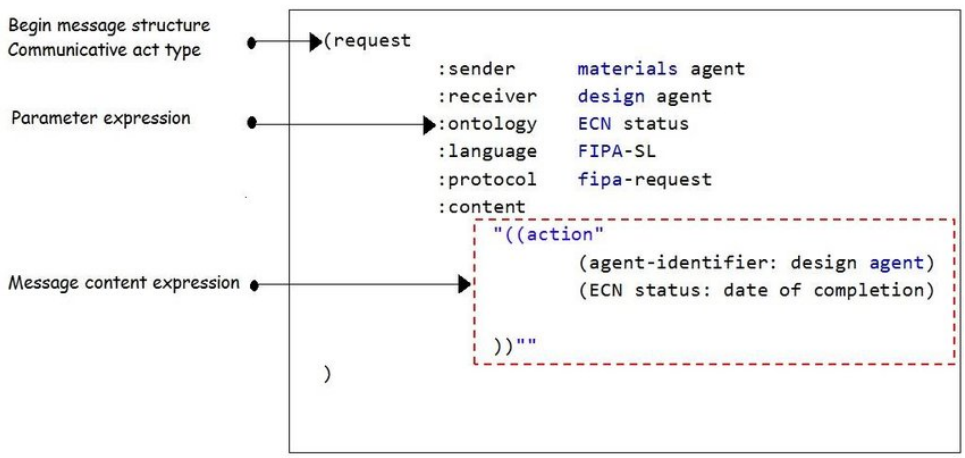

Figure 3.2: Diagram displaying a FIPA-ACL message structure.

As mentioned previously, the most meaningful feature of intelligent agents is that they are autonomous. The relevance of this attribute extends to multi-agent systems. It enables these systems to self-organize and to find the best solutions to their problems without any kind of intervention. It is extremely important to distinguish correctly between automatic and autonomous systems [28].

Automatic systems are designed and fully programmed to act repeatedly and independently of external interactions. One could describe them as self-regulated system as they are able to follow an externally given path. For this reason, these systems lack the ability to define their own path according to a given goal.

On the other hand, autonomous systems, as a multi-agent system, are determined to achieve their objectives without the need of an external interaction. Instead they follow the laws and strategies that clearly make a difference between traditional and multi-agent systems. On top of that, if machine learning is included in the equation, autonomous systems can can adapt their behavior depending on the feedback received and therefore change their actions dynamically.

Agents, due to their socializing capabilities, are able to develop relationships. They exist in a society where they perform actions to achieve their goals. Therefore, coordination is a distinguishing element of multi-agent systems, as it allows agents to achieve a coherent behavior in a system. However, this does not imply that cooperation between agents always exists. This presents the case that agents may identify others as allies and therefore be willing to form coalitions to achieve common objectives. On the contrary, agents may compete with each other to fulfill their desires, as they may have antagonistic or exclusive.

Furthermore, interactions lead to solutions that exceed by far the capabilities and knowledge of a single agent. Another characteristic of multi-agent systems is their emergent behaviour. This concept does not refer to the outcome of one or a few agents but the global outcome achieved by the coordination of the whole system [42]. In other words, emergent behavior can be described as a collective behavior.

Other definitions, emphasize that emergent behavior cannot be predicted unless the system is analyzed as a whole. Emergent behavior, by definition, is what's left after everything else has been explained [22]. This highlights that the predictability and explainability of emerging behaviors is a difficult task. Therefore, by definition, predictable behavior is not an emerging one. Emergence is also defined as the action of simple rules combining to produce complex results [59]. In other words, tasks of individual agents may be simple, but combined, they can generate a complex and unpredictable collective behavior.

Essentially, any behavior of a system that is not produced by a component of itself is considered emergent behavior. This is created from the interactions among the system's components. Focusing on biological model colonies, societies or organizations exhibit emergent behavior. This is due to the fact that the production of complex behaviors can be achieved through the interaction of multiple simple entities. Therefore, it can also be described as an unconscious behavior performed by a group as a result of their interactions.

The previous statements may help one to understand how computing can be carried out through communication between computational entities. Consequently, in this scenario, computing can be a social activity rather than an individualistic action. This led to a new computational paradigm, which can be refereed as *computation by interaction*. Which leads to new ways of conceiving, designing, developing and managing computational systems which require both interaction and negotiation.

As it has previously been stated, agents exist in a social environment. As they live in a society subject to regulations, these can impact on the agents' deliberative cycle. For this reason, sometimes they will need to take into account the social consequences of their actions [11].

3.2 Automatic negotiation

Negotiations can be defined as the collection of mechanisms by which self-interested agents are able to interact in order to coordinate themselves. Negotiations are carried out specifically for each task and/or resource allocation, the recognition and resolution of conflicts and the determination of organizational structures [49].

Taking this into account, it is possible to affirm that negotiations emerge from the need to resolve conflict and achieve an agreement from several parties. The first papers related to negotiations between self-interested were published in the 90's with the system PERSUADER from Sycara [68] and the work from Rosenchein and Zlotkin [60].

Nowadays, it is possible to differentiate between manual and automatic negotiation. Manual negotiations are performed by a human user. This means that it will take a longer period of time to accomplish and, therefore, it will be more costly. On top of that, human emotions, such as embarrassment or frustration have not been taken into account during the execution of this activity.

In contrast, automatic negotiations are executed by software agents. As a result, no human action is involved during the negotiation period. This automation leads to yielding higher quality results since human emotion are kept out of the equation [45] [17] [44]. Additionally, automatic negotiation can be interesting not only from an application perspective but also from a mechanism design point of view. This allows to experiment and simulate more scenarios [4] [21]. Inherently, this gives researchers the tools to facilitate the study of negotiation technologies and the simulation of agreements.

In the area of multi-agent systems, negotiation technologies were identified as a key component by Wooldridge [76]. On the first place, as mentioned previously, it is the cornerstone for the interaction paradigm as a way of computing. Secondly, they are very relevant for the development of techniques that enable software components to reach deals to exchange goods via agreement technologies [66].

Indisputably, negotiations are part of agreements [36]. Obviously, there are different types of negotiations depending on terms like the type of deal, the participants taking part, the protocols being used and the set of rules applied [56] [51]. That is why, automated negotiation is emphasized as a core technology to reach agreements in this new computing paradigm.

Depending on the nature and the cardinality of an interaction, it is possible to classify automatic negotiation models into [36]:

- One-to-one: this model refers to the direct interaction between two individual agents [8] [23].
- One-to-many: in this model one agent deals with several different others. This category can be easily identified in auctions [54] [2] [65].
- Many-to-many: this model involves the cross-interaction of N to N agents. Bilateral negotiations are the most representative example of this model [74] [75].
- Argumentation based: agents from this model have the resources to generate arguments, threats, rewards and appeals to persuade other agents to agree to the offered terms, which might have not been accepted before this action. [39] [38].

Negotiations have three distinct elements [36]:

- Negotiated objects: this refers to *what* is being negotiated. When a discussion emerges it is necessary to clarify the stakes at hand. This defines the objectives of the negotiation, which can be as simple as only taking into account the negotiation price or more complex if different elements such as quality, time, regulations, terms, conditions etc. are taken into consideration.
- Negotiation protocols: these are the set of rules which define *how* the interaction between involved agents is done. It also includes negotiation states (eg. negotiation offering, counter offering, offer acceptation, negotiation closure), the events that guide the whole interaction (eg. no negotiable items left, negotiation time exceeded, etc.) and the valid actions agents can perform at a specific state (eg. who can communicate, to who they can send messages, when can they interact, etc.).

- Negotiation strategies: this is *how* the internal deliberation of negotiating agents is performed. It is their reasoning system to take decisions in a negotiation in order to achieve their objectives. The complexity of is this model as well as the amount of decision making required is directly influenced by the object negotiated.

Negotiation protocols

Although there is a vast amount of negotiations protocols, the three most popular are [31]:

- Market mechanisms: this model compares a multi-agent system with a virtual market. The aim of this model is to simulate a real market properties in a virtual environment. It is based on the known *general equilibrium theory* [71]. This theory allows agents a distributed method to efficiently assign tasks and resources to the negotiating entities. The model developed in [10] studies distributed auction mechanisms and bilateral negotiations to assign the negotiated object between the actors in the system (suppliers and demanders).
- Haggling theory: during a haggle, entities push towards achieving a mutually beneficial agreement. However, the conflict relies on which side will make the bigger concession. This theory has two main approaches. The strategic approach focuses on the actions each player is capable of doing to achieve some kind of equilibrium. On the other hand, the axiomatic approach studies the results instead of the means to achieve them. Therefore, it focuses on establishing the desirable properties of the solution with axioms [69].
- Coalition formation: this protocol is one of many cooperation mechanisms. Cooperative game theory studies mathematical models for multi-entity cooperation and conflict resolution. Agents' objective is to maximize their own utility, assuming that the rest of the participants will act in the same way [6].
- Voting: this mechanism has the objective of simulating the rules of social elections. This establishes a result based on the individual choices of each participant [62].

3.3 Background and related works

When analyzing the scientific literature, one can see that there are numerous works that use automatic negotiation to create virtual markets based on intelligent agents, but each one from a different perspective.

When the first multi-agent systems were conceptualized, developers focused on the agents' design: how they were internally structured and their behavior. Organizations were merely thought as a consequence produced by agent interaction, and, therefore, the agencies' analysis and design were not taken into consideration. Additionally, agents were only seen as autonomous entities which dynamically evolved as their desires changed, without considering external restrictions that influenced their communications or behavior [12].

Nowadays, developers have changed their focus to the organizational aspects of agent societies [24] [15]. This implies that the design process has introduced the concepts of organization, groups, norms, roles, etc. to describe an environment at organization and agent level [12]. Agents are now understood as dynamic entities that are able to change inside an agency. This limits their actions, but at the same time is the result of their activities.

However, all of these distinctive features must be implemented explicitly or implicitly by developers. Depending on an organization's topology, different limitations in communication and interaction would need to be imposed [5].

Moreover, the design of a market's environment has been receiving increasing attention. In chapter 2 it was made clear that the environment is a crucial factor that needs to take into account in their decision making process. For instance, governmental policies may be needed to considered before allowing any interactions. Therefore, valid actions need to comply with these policies [30] [79], which means that external inputs can also undertake an important role when modeling a virtual market in order to react to economic, financial, and regulatory environments.

Different intelligent computing methods have been used in the recent years. Initially, classical methods such as knowledge based systems and case based reasoning were used the most. With this approach studies explored different models manually. Later on, the researchers' perspective shifted towards using optimization algorithms to automate the exploration of different agent behaviors.

Nowadays, models have transitioned to use reinforcement learning algorithms as agents are able to find the optimal strategy on their own [33]. Moreover, with the increasing amount of available data it is also common to include other machine learning techniques to assist agent reasoning [46].

Within the set of virtual markets, it is possible to find three main markets were researchers focus their efforts on. These are e-commerce, finance and electricity markets. Throughout the years these have been the sectors with most research related to multi-agent systems and virtual markets.

Although there is limited research towards generic multi-agent systems, one can note a trend of increasing interest in this field. Some examples are:

- The creation of a generic software platform for the geo-simulations in virtual geographic environments involving several thousand agents [48].
- The development of a framework for the generic simulation for physical multi-agent simulations in three-dimensional environments [50].
- The implementation of a generic software framework for car-sharing modeling based on traffic simulation [41]. This work was mainly motivated by the numerous car-sharing systems created in the last years.
- The design of a generic spatial Susceptible-Infected-Recovered (SIR) model for infection spread management simulation [7]. This study was driven to improve the management of future epidemics.

Regarding generic market studies, it is important to state that their development is scarcer and it focus only in one domain. These studies are related to e-commerce markets [37] and electric markets [63] [64]. Consequently, no research evaluating the capabilities a flexible generic virtual market can offer was found.

Even though, one can find research on generic virtual markets, they mainly focus on one domain. This project aims to be able to model both, e-commerce markets and electricity markets. Financial markets are not taken into account as the way actors buy and sell financial assets in the stock market or forex market works differently to how an item or a package of items are negotiated in other markets.

Additionally, the published works provide a poor visualization about the negotiator's interaction. Normally, the events occurred or the messages in a negotiation are displayed as log registers. Therefore, another goal of this study is to tackle this problem by providing a friendlier and more comprehensive visual design.

3.4 Multi-agent system frameworks

Before starting the development of any application is important to review the available tools for the task at hand. Therefore, to develop this generic market multi-agent system some of the most popular frameworks have been studied in order to choose the most suitable instrument. The examined frameworks are shown below:

- Java Agent DEvelopment Framework (JADE): JADE is a software framework fully implemented in Java language. It uses the FIPA specifications and graphical tools to simplify implementation of multi-agent systems and support the debugging and deployment phases. It also enables the possibility to change the configuration at run-time. This framework is a middleware which has been in use for more than a decade and has been widely referenced in multi-agent system literature. The documentation allows users, given a moderate level of experience in Java programming, to develop agents in a very simple way. Additionally, there is a wide support in the community and many examples of it being used [9].

- Jason: Jason is an interpreter for an extended version of AgentSpeak [57]. AgentSpeak has been one of the most influential abstract languages based on the Beliefs-Desires-Intentions (BDI) architecture. Jason implements the operational semantics of that language. It has agent's basic functionality. As a result, developers can focus on the agent's logical reasoning. In the vast majority of the cases some knowledge of Java programming will be requiered (e.g., if you want to create an environment where the agents are located). It is considered one of the best-known approaches for the development of BDI agents. Additionally, it provides the possibility to run a multi-agent system distributed over a network [13].

- Magnetix2 framework: Magnetix2 framework is an agent platform for open multi-agent systems. Its main objective is to bring agent technology to real domains: business, industry, logistics, e-commerce, health-care, etc. Another of its key characteristics is its focus to develop technologies to cope with the high dynamism of the system topology and with flexible interactions, which are both natural consequences of the distributed and autonomous nature of the components. Finally, Magentix2 incorporates the THOMAS framework, allowing users to manage with organizational and service aspects easily [67].

- Python Agent DEvelopment Framework (PADE): PADE is a framework which allows users the development, execution and management of multi-agent system applications in environments of distributed computation. The language in which this framework is developed is in Python. It is object oriented and uses FIPA-ACL message and protocol standarization. It also includes a graphical user interface (GUI) for easy management of the MAS application developed [47].

- Smart Python Agent Development Environment (SPADE): SPADE is a multi-agent system platform written in Python and based on instant messaging (XMPP). Instant messaging helps to develop agents that can chat with other agents and/or humans. The agent model is basically composed of a connection mechanism to the platform, a message dispatcher, and a set of different behaviours that the dispatcher gives the messages to. Every agent needs a Jabber Id and a valid password to establish a connection with the XMPP server. It also includes a native web visualization tool to examine the actions and communication of agents [52].

- Netlogo framework: Netlogo is a programming language and integrated development environment for agent-based modeling. It was designed for modeling complex systems which evolve over time while supporting thousands of agents operating independently and listening for the modeler's instructions. The purpose of this framework was to explore the connection between the micro-level behavior of individuals and the macro-level partners that emerge from their interaction. This has been very popular among different groups as it is used by many hundreds of thousands of students, teachers, and researchers. Also, includes a GUI to easily explore the possibilities of simulations [1].

3.4.1. Selected framework for development

After analyzing the advantages and disadvantages of each framework, the framework chosen to develop this work's prototype was SPADE. The main reasons that leads us choose SPADE were the ones explained below:

- Programming language: SPADE is developed in Python. Nowadays, Python and Java have a very similar number of users. However, Python's code is more readable and faster to develop than Java, since it has simpler syntax. Additionally, Pyhton is considered more accessible for non-IT developers which would want to add custom functionality.

- JSON vs XML: an important factor is that SPADE uses JSON structured data. Traditionally, multi-agent frameworks primarily used XML format. However, JSON is more readable, has the ability to store arrays and, on top of that, it is easier to parse. For the reasons explained above, JSON seems a better format.

- SPADE vs. PADE:
  - SPADE has a better way to filter messages through behavior queued inboxes. This can be seen in figure 3.3, were each behavior can receive a different template. This is was a key factor to identify important messages from the normal receiving messaging queue.
  - SPADE has more available behaviors. This enables to have a more customizable development. If SPADE and PADE are compared, the first one has four and the second one has only two available.
  - PADE's documentation is far less detailed and it has a less active community than SPADE.
  - SPADE customization is greater than the one in PADE. SPADE only includes the bare bones of a multi-agent system platform. It was design to be as a very simple version and provide only the core features of multi-agent systems. Regardless of this, it does not lack any elements for development. SPADE is being extended through plugins. For example, it has extensions for BDI agents, PubSub protocols, Artifacts, etc.

Thus, as described above, SPADE has been selected as the main tool to develop this work.

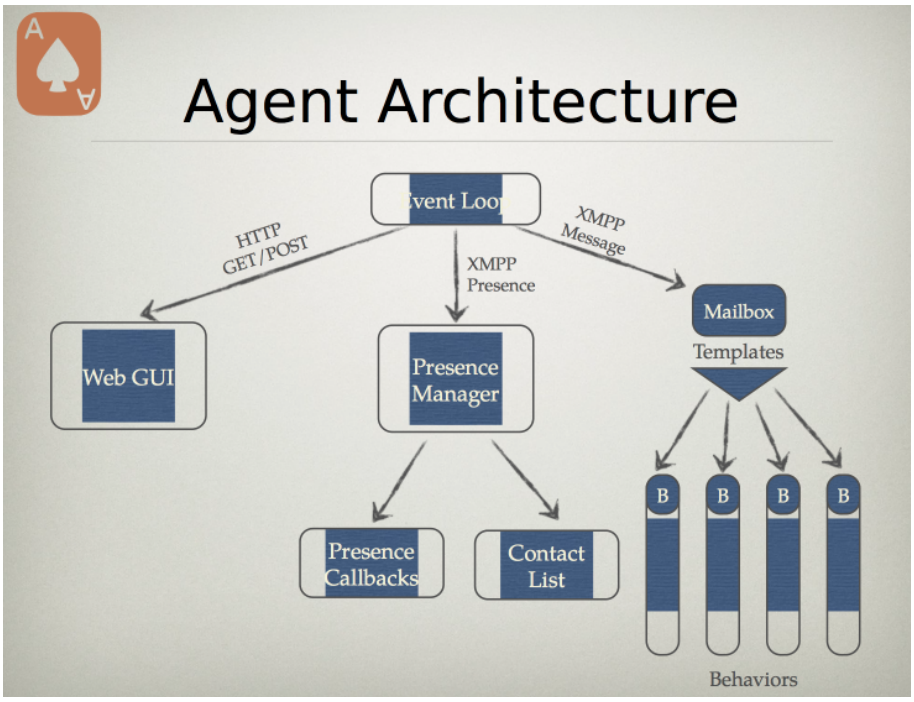

Figure 3.3: Diagram showing a SPADE's agent architecture.

CHAPTER 4
Market development

In this chapter our proposed solution to the generic virtual market problem is offered. In order to achieve it, organizational concepts and multi-agent system technologies will be used. While these have been explained throughout chapter 2 and 3, respectively, the current chapter will illustrate each software development stage.

Before developing the market, the first step one should do is to outline a formal proposal. Secondly, the generic virtual market's needs must be analyzed. Subsequently, a design that meets all the requirements set out in the analysis phase should be made. Finally, after its implementation, various validation test should be carried out in order to verify that the application works as planned.

4.1 Proposal

The layout proposed for the generic virtual market simulator is a three-layer structure. Likewise, one can find the system layer, the market layer and the negotiation table. The first layer is the system layer as it is the one where a market is opened and the negotiations may take place in. Secondly, the next layer is the market layer. This layer can be seen as a negotiation hall, where all the agents willing to take part in this market may be invited to the negotiation process. Lastly, the third layer represents the negotiation table. Here, the agents interested in taking part in the negotiation process register for the event and begin to interact with one another.

For a better understanding, the three-layer structure will be explained through an example. The example chosen is a casino. From a casino hall one can see different areas of gambling games, such as poker tables, blackjack tables, slot machines etc. One may ask or be asked to join a table as well as decide not to play or to change game.

In addition to the explained above, a market has n restrictions to be "solved". A restriction is defines as the item's maximum allowed quantity that can be exchange during a negotiating period. This value can range between $\{-\infty, \infty\}$. A negative restriction value shows that the market is interested on selling items, while a positive restriction value indicates the market is interested in buying them. A zero value means that there is no interest in any transaction being done. Positive and negative $infinity$ restrictions allow to trade an unlimited quantity of goods until the end of a negotiation period.

Therefore, given a list of restrictions for each non-zero value it is mandatory to arrange a negotiation process. Agents will be invited to participate in this event, and on their interests, they can accept or decline the invitation.

Negotiations involve a 2-step process between n buyers and n sellers. Firstly, the process begins with a bilateral negotiation and ends with an auction. On one hand, a bilateral negotiation enables agents to bargain the price of items while, on the other hand, the auction offers the agents a final opportunity to buy a desired good. It is important to state that both phases can be interrupted if agents do not reach an agreement.

The purpose of a 2-step negotiation is to promote as much transactions as possible, enable negotiators to maximize their profit, and allow both parties to select a negotiation strategy for each phase. Furthermore, this does not interfere with the desire of agents to negotiate with a fixed price or skip to the auction process if they think they can obtain better prices.

Agents negotiate the price at which they exchange n units of an item during the negotiation process. When agents are invited to a negotiation table it is because it is assumed that they could be interesting in the negotiation process. The element negotiated is implicitly known by the agents involved.

4.2 Analysis

Before analyzing the simulator requirements, it is necessary to state that the software developed must follow the thesis's proposal. Also, the application must be functional in different market scenarios and provide an easy characterization of agents. In addition, this work also develops a web visualizer to simplify the understanding of the interactions occurred.

In this section, the requirements needed to be met by the simulator will be described below.

4.2.1. Market

The features the market must have are the following:

- A negotiating entity to represent the role of a buyer, a seller or both during a negotiating period.
- An entity to represent the control authorities in the whole system.
- An entity to represent the control authorities in a specific market.
- An entity to represent the control authorities at the negotiation table.
- Negotiation periods for each of the market's restrictions must be created.
- Each negotiator agent must be parameterizable in order to customize its behavior. Negotiators must have a role, an initial negotiation price, a final negotiation price and the negotiation strategy they want to adopt during the process.
- Information about the final state of the restrictions and the ended satisfactory negotiations must be available for users at the end of the simulation.

4.2.2. Web visualizer

The characteristics the web visualizer must have are the following:

- The communication performed between agents will be presented in a clear and simple way.
- The messages must show when an agent is invited to the process and if it is willing to enter a negotiation.
- The messages must clearly display how the negotiations progress in time.
- The results of each negotiation period must show the result of the period's restriction.
- Each period will show the agreements reached by agents, including its properties: agents involved, agreed quantity and agreed price.

4.3 Design

The design phase of an application is an advisable and preliminary step before the implementation phase. This section describes and argues the design decisions done for this project for, both, the market and visualizer.

4.3.1. Market

As previously explained in this work, it has been discussed why multi-agent system technologies have been selected to develop the proposed solution. This section displays a complete view of the market's features. It begins by introducing the system as a whole and continues by analyzing in depth its different components.

It should be noted that the design presented in this section is not the only possible solution. Hence, this is one of many designs available that can offer an effective solution to the requirements stated previously. Despite this, this design has been planned out to be flexible enough so that it can include future changes.

Furthermore, the design of the market focuses on having a flexible baseline from which it is possible to build-up different applications.

System

This section outlines not only the system's structure but also introduces the roles involved in a system. Moreover, the organizational relationships between agents are explained.

In figure 4.1 it is possible to see how the three-layer market is structured and how it inherently offers a hierarchical structure. This figure shows how the system is designed as a pyramidal organizational structure. The agent found in outer layers will have more authority that ones found in inner layers. From highest to lowest control authority, the agents are ranked as such: system manager, market facilitator, table manager and negotiators.

A classical business hierarchy is a clear example of this type of system structure. The company's president would be the equivalent of the system manager, the department manager of the market facilitator, the team lead of the table manager and the regular employees of the negotiator. Finally, departmental tasks would represent the restrictions. As mentioned previously a restriction is the maximum allowed quantity that can be exchange during a negotiating period.

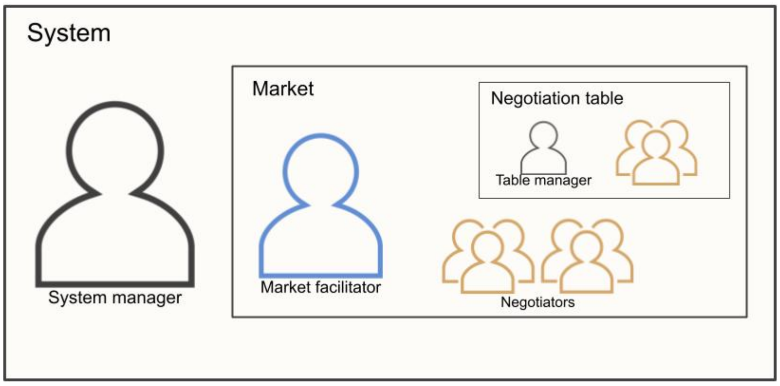

Figure 4.1: Sketch of the market proposed.

In this proposal, the system layer is where the system manager opens a market. The market layer, which can be seen as a negotiation hall, is where negotiation opportunities are coordinated. The final layer, the negotiation table, is where the proposed 2-step negotiation process takes place.

As stated previously, negotiation process is made up of a bilateral negotiation between the buyers and sellers involved in the negotiation process. Afterwards, it continues with an auction, where sellers will take the role of auctioneers and buyers will become bidders.

In this initial design of the market, it is considered that bidders are not concerned about who provides the good offered. Instead, they want to buy as much as possible before the restriction limit is reached. Therefore, the bidding places bidders in a descending order from most willing to buy to least. This process is achieved via a first-price sealed bid auction. After all bids are placed, the table manager is in charge on matching the bidders in descending order with the auctioneers.

Furthermore, figure 4.2 defines the roles involved in the market in order to fulfill the analyzed requirements. Four main roles have been identified. Additionally, a negotiator agent will be able to undertake the sub-roles of a seller, buyer or both. The following section will discuss in further detail this idea/concept.

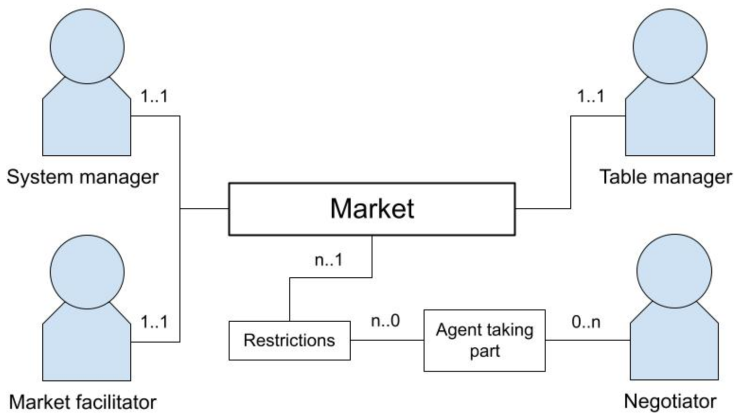

Figure 4.2: Diagram of the roles involved in the market proposed.

Agents

This section will explain the market's identified roles, their beliefs, objectives and tasks each one of the roles has to accomplish. It is worth to mention that tasks are identified specifically and individually for each role. This is accomplished by carrying out a modular design of the agents. Therefore, if an agent's behavior needs to be modified, it is not necessary to modify everything concerning the agent's implementation.

The roles identified are the following:

- **System manager**: this agent is in charge of setting up the system and opening the market. Another of its responsibilities is to set the market's restrictions and identify the agents involved in it. After all negotiation periods have finished, it gathers the results of the market.

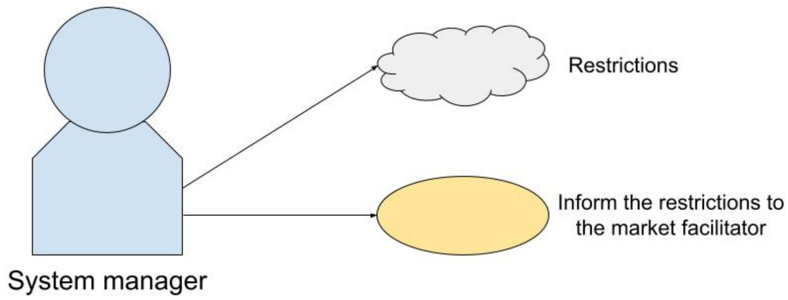

Figure 4.3: Diagram of the system manager's role.

- **Market facilitator**: this agent's main goal is to set up all the negotiation periods correctly. Each negotiation period is related with the restriction established by the system manager.

  The role of the market facilitator is essential during the development of the market. It must coordinate the start of a negotiation phase, specifically the registration process. This process ensures that only valid agents enter a given negotiation table. Therefore, this agent must be aware of all the organizational convections and market rules in order to manage the sent invitations. Despite of this, it is not directly involved in the negotiation process. It focuses on overseeing that everything runs coherently in the negotiation hall.

  The invitation process is carried out following the steps explained below:

  1. Send an invitation, which includes the current restrictions, to the negotiators.
  2. Authenticate the agent's participation by verifying that the negotiator's capability of buying or selling items corresponds to the role they want to undertake.
  3. After all negotiators have replied in due time, the table manager is informed about the current restrictions, as well as the negotiator agents taking part. The information about the agent's role and their maximum negotiable quantity is also provided to the table manager.
  4. Send negotiators the contact list of the agents with their counterpart role. Give sellers the buyer's contact list and vice-versa.
  5. Initiate the negotiation process.

  Finally, after all negotiation sessions end, the results of these are communicated to the system manager.

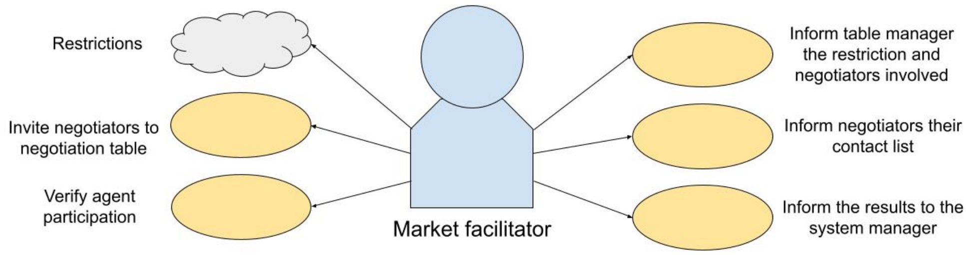

Figure 4.4: Diagram of the market facilitator role.

- **Table manager**: this agent is a mediator in a negotiation table. It must lead the negotiations with minimal intervention, letting these run smoothly.

  Its main task is to ensure all reached agreements are done correctly. In other words, it must verify agents are undertaking the role they are supposed to and check that agents are able to sell or buy the agreed number of items. As a result when an agreement is reached the table manager needs to update its beliefs about the negotiator's maximum negotiable items. This needs to be done to ensure that the amount of negotiated items in all agreements is correct.

  Agreements must be verified for bilateral negotiations as wells as winning bids. The validation of n to n bilateral negotiations is quite straightforward. As soon as a verification petition from both agents is received, the agreement will be confirmed if all the terms mentioned above are satisfied.

  In order to verify bids, although the conditions to be met are the same, the process is different. In this case, the table manager needs to match bidders and auctioneers. At first, buyers are arranged regarding their bid, from the highest amount to the lowest and, afterwards, they are matched with the auctioneers in descending order.

  Additionally, the table manager must take action in some specific scenarios:

  - **Transition to the auction phase**: the restriction has not been solved in the bilateral negotiation phase. This occurs because not enough agreements are reached during the bilateral negotiation phase. Settlements are not reached since they are not economically viable. If negotiations do not naturally cease, the table manager is responsible of stopping all negotiations and making negotiators transition to the auction phase.
  - **End the negotiation period**: this can happen in two case scenarios:
    * The restriction has been solved - this scenario can take place in the first or second phase of a negotiation process. If agents reach enough agreements to solve a restriction, no more agreements are allowed to take place and the period must end.
      On one hand, if this happens in the bilateral negotiation phase, the auction phase will not be reached and the negotiation period ends. On the other hand, if the auction phase is reached, the negotiation process must end either if the restriction is solved or not.
    * The negotiations have ended - this means no agent is willing or able to continue further negotiations and the negotiation process must end.

  When this process is over, the table manager must communicate the events occurred in the negotiation table.

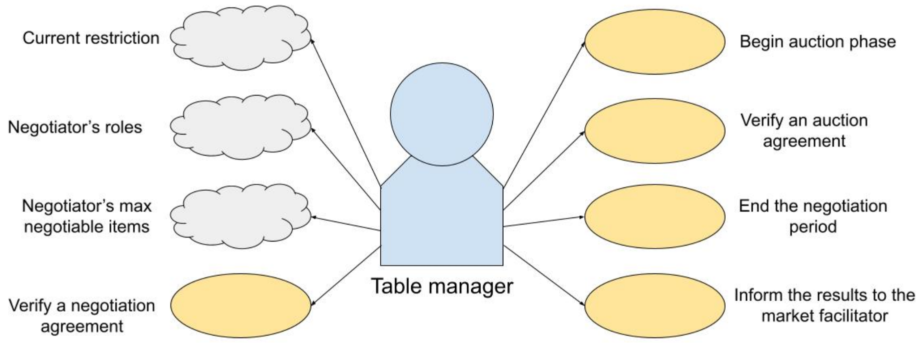

Figure 4.5: Diagram of the table manager role.

- **Negotiator agent**: this agent is the one in charge of carrying out all the negotiations. It is important to specify that negotiator agents have full capability of assuming the sub-role of a seller and of a buyer. This is done to allow a negotiator to take both roles in a negotiation period. Furthermore, the same negotiator agent could take on both roles but the items bought cannot be sold during the same negotiation process.

  Negotiator's objective is to exchange its maximum number of items at the best price as sellers will try to maximize their selling price, while buyers will try to minimize their buying price.

  In order to take part in a negotiation period, it must accept an invitation from the market facilitator, where the role it will undertake and its maximum negotiation items for that period will be stated. After its registration in the negotiation table, it receives the list of agents with who it can negotiate. Therefore, the seller agents will receive the buyer's list and vice versa. The agents assuming both roles will receive both lists. Following this message, agents will set their initial price.

  At the beginning of the bilateral negotiation phase, sellers will start the negotiations. When an agent receives an offer from all active negotiators, it will check if they are admissible and, then, it will choose the most beneficial one. If they accept the offer, it must halt other negotiations, notify the table manager and the other involved entity in order to formalize the agreement before resuming other negotiations. Otherwise, they must calculate the next proposal can counter-offer and negotiations may continue.

  Moreover, negotiators are able to distinguish the role they must take in a negotiation by checking the role of who sent the offer. For example, given a seller-buyer agent John, he can receive offers from both sub-roles. When John receives an offer, he checks the role of the sender. Therefore, if John receives an offer from buyer agent Karen, stating she is a buyer, John knows he needs to take the role of a seller in that negotiation.

  Furthermore, these agents may decide to stop negotiations if they are not able to negotiate any more items or if they have the time limit for that negotiation. The negotiation time limit is determined by the maximum amount of interactions they can perform. One offer and counter-offer counts as one interaction.

  At the end of the negotiation process, in the auction phase, buyer agents will send their bids to the table manager so that the match between both parties can take place.

Regarding the price strategies applied, it is important to state that sellers and buyers will use descending and ascending price strategies respectively. These strategies will be explained in further detail along this section.

The following diagrams show the knowledge and available actions agents will have with the negotiator role, the seller sub-role and buyer sub-role will have:

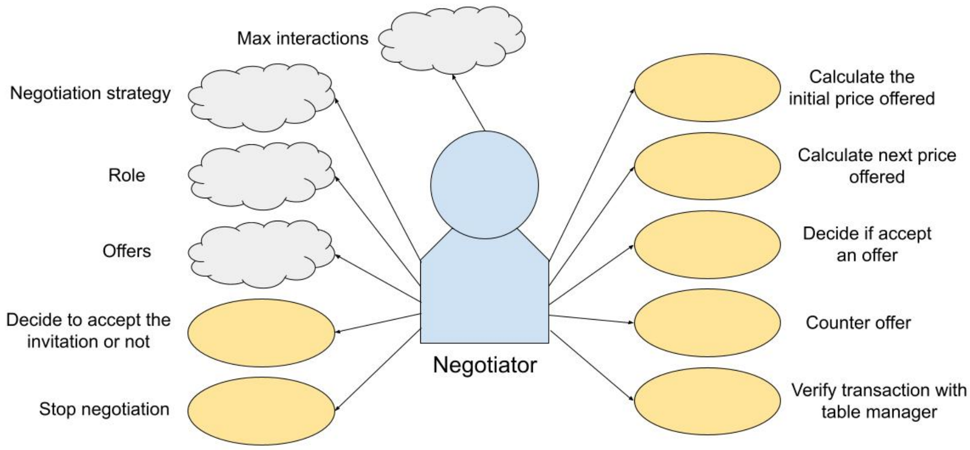

Figure 4.6: Diagram of the table negotiator role.

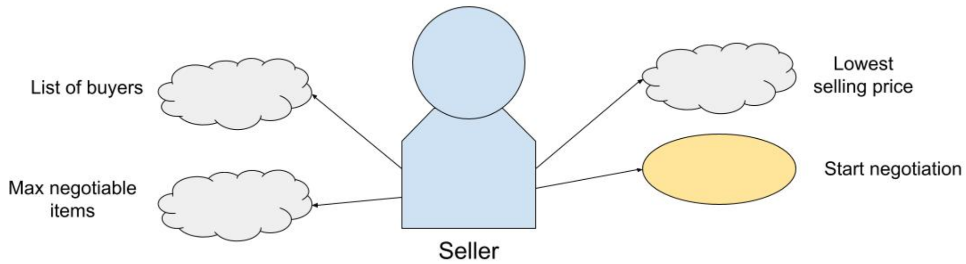

Figure 4.7: Diagram of the table seller sub-role.

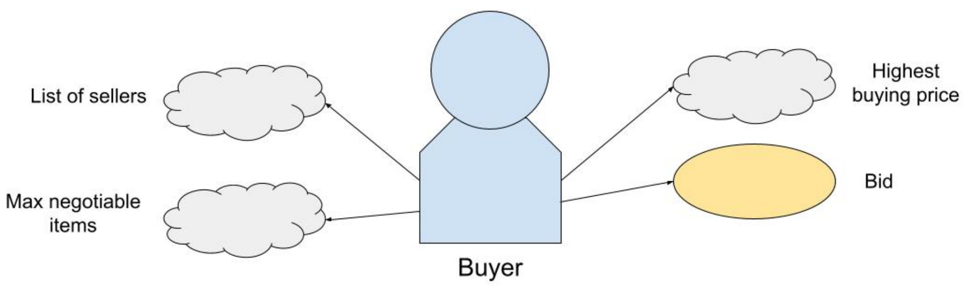

Figure 4.8: Diagram of the table buyer sub-role.

Class diagram

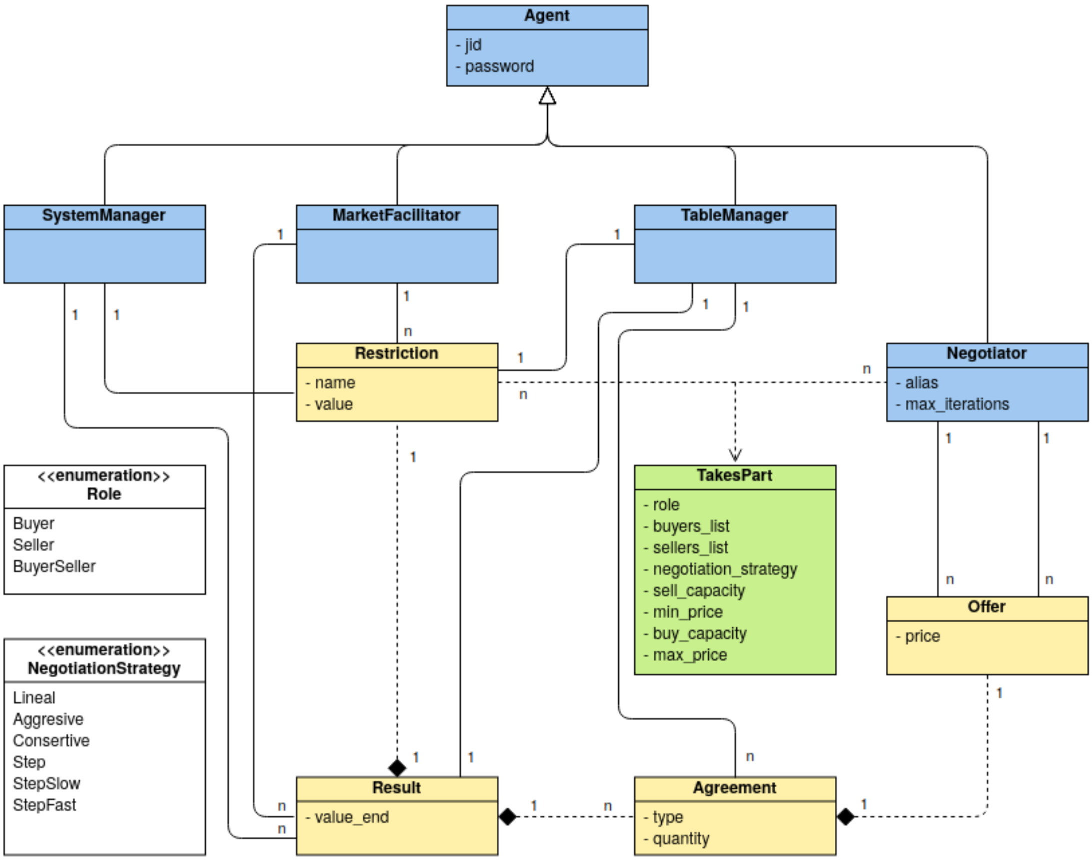

Figure 4.9: Diagram of the classes used in the market.

Figure 4.9 represents to the market's class diagram. The blue classes symbolize the agents involved in the market. The yellow classes are related to the results of each negotiation period. The green class illustrates to the characterization of negotiator agents. The white enumerators are constants to parametrize negotiator agents. Next, the classes will be explained further detail:

- **Agent**: is SPADE's agent parent class from which all agents are developed. It contains all the necessary methods and attributes to instantiate an agent.
- **SystemManager**: this agent in charge of opening each market with its restrictions. It contains two different lists, the market restriction's list and the restrictions' results list.
- **MarketFacilitator**: this agent opens every negotiation period for each restriction. Therefore, it must have the list of restrictions of the different negotiation periods. In addition, when this period ends it receives the results of the negotiations.
- **TableManager**: a mediator agent is needed during every negotiation period. This is the table manager's task and in order to accomplish it, this agent needs to know the restriction of the period it is in. Also, it is in charge of the list of verified agreements reached by negotiators. Each period these are put together as the result of a negotiation process.

- **Negotiator**: this is one of the agents that takes part in the negotiation periods. For each restriction, it has parameterizable features which define its behavior as well as the role it takes and the agents who it can negotiate with. Also, in order to perform negotiations, it has a list of offers of the other negotiator agents.
- **Restriction**: this class contains the maximum negotiable limit in a negotiation period.
- **TakesPart**: this represents all the parameters a negotiator has to take part in a restriction. Moreover, they define a negotiator's interests and conduct.
- **Role**: is defined by the group of constants negotiators may take for a role.
- **NegotiationStrategy**: is the group of constants negotiators use to specify their negotiation strategy. As will be explained later, they may change the negotiation strategy of the agents.
- **Offer**: represents the offer made from an agent to another, where the price at which the sender wants to settle the agreement is detailed.
- **Agreement**: once an offer wants to be settled, it is verified by the table manager and an agreement is created. An agreement class describes the type of negotiation, the quantity agreed as well as the settled price.
- **Result**: this represents all the reached agreements associated to a restriction.

Price reasoning

As mentioned in previous sections, negotiators must be able to know what price to offer during a negotiation. This section displays how these agents are able to reason and calculate the offered price in each step on the negotiation.

Moreover, negotiations are time to avoid sending infinite useless offers. This time limit is represented as the maximum number of iterations in a negotiation. Apart from defining a maximum number of interactions one must establish the initial price offered, the final or limit price offered and the function that connects both prices.

Firstly, the initial price needs to be set. Buyers will calculate a lower price than the maximum they are willing to pay. Meanwhile, sellers will calculate a higher price than the minimum they are willing to sell. Negotiators use, these functions to calculate their initial prices:

$$Seller\_IP = FP \cdot IP\_factor$$

$$Buyer\_IP = \frac{FP}{IP\_factor}$$

Where:

- Seller_IP: initial priced offered by sellers.
- Buyer_IP: initial priced offered by buyers.
- FP: final or limit price offered.
- IP_factor: value that defines how much the initial price is increased or decreased. The default value is 4.

Two different functions have been designed to calculate the price change offered during negotiations. These are a concession function and a linear step function. Both of them can be used by seller and buyer agents. In addition, the concession function can be parametrized to have a linear, a more aggressive or a more conservative change in price.

On the other hand, the step function is always linear but the number of steps can be changed to three, five or eight, until the limit price is reached. Negotiators use one of the six possible functions depending on its negotiation strategy variable.

The concession function is defined as:

$$Price\,(t) = IP + (FP - IP) \cdot \left(\frac{min\,(t, T-1)}{T-1} \right)^{\frac{1}{\beta}}$$

The step function is defined as:

$$Price\,(t) = IP + \frac{FP - IP}{step} \cdot \left\lfloor \frac{min\,(t, T-1)}{\left(T/step + 1\right)} \right\rfloor$$

Where:

- IP: initial price offered.
- FP: final or limit price offered.
- T: maximum number of iterations of a negotiation.
- t: current iteration of the negotiation.
- β: value which defines the functions curvature.
  - $\beta = 1$: it will have a linear behavior and all iterations will have the same change in value.
  - $0 < \beta < 1$: it will have a conservative behavior. The first iterations the price will change less its value. However, in the last iterations the price will vary faster as it approaches the limit value.
  - $1 < \beta$: it will have an aggressive behavior. In the initial iterations the price will change in big increments and in the final iterations the price will vary less as it approaches its limit value.
- step: the number of increments taken from the initial value to the final value.

For a better understanding of these functions, an example is prepared for all the negotiation strategies the negotiator agent may take over a 20 iterations period. In this example, 10 and 2 are the IP and FP for sellers and 2 and 10 are the IP and FP for buyers, respectively. The concession function is used for the conservative, linear and aggressive strategies, where β equals 0.5, 1 or 1.5, respectively. This examples are shown in figures 4.10 and 4.11

The step function is used for the slow step, the step and the fast step strategies. The steps taken to reach the final price are 3, 5 and 8 respectively. This examples are shown in figures 4.12, 4.13 and 4.14.

Figure 4.18 and 4.19 show how these functions behave against a linear strategy. As expected from the designed functions, the more aggressive strategies will reach agreements faster.

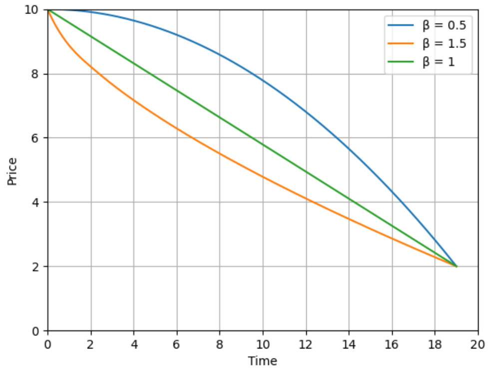

Figure 4.10: Graph of the strategies used by a negotiator agent with the seller role.

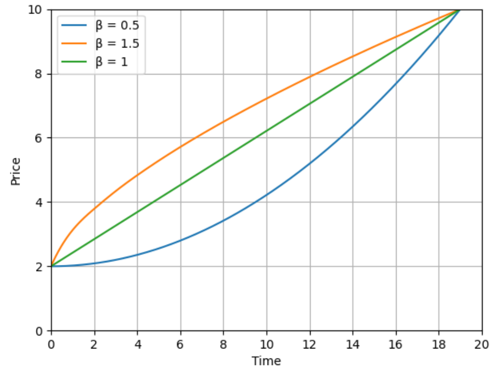

Figure 4.11: Graph of the strategies used by a negotiator agent with the buyer role.

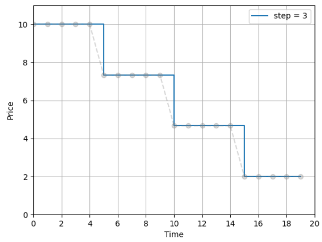

Figure 4.12: Graph of the slow step strategy used by a negotiator agent with the seller role.

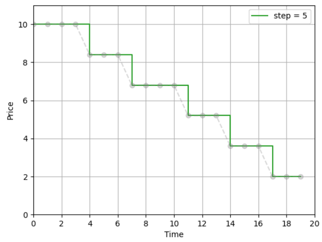

Figure 4.13: Graph of the step strategy used by a negotiator agent with the seller role.

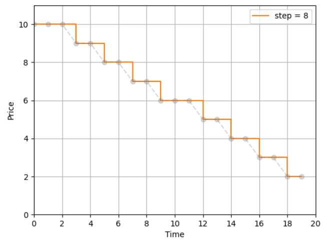

Figure 4.14: Graph of the fast step strategy used by a negotiator agent with the seller role.

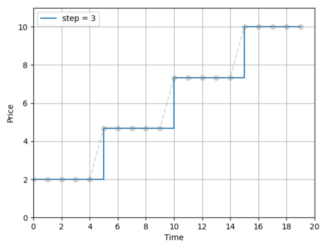

Figure 4.15: Graph of the slow step strategy used by a negotiator agent with the buyer role.

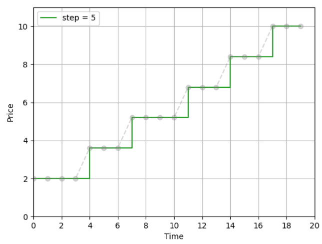

Figure 4.16: Graph of the step strategy used by a negotiator agent with the buyer role.

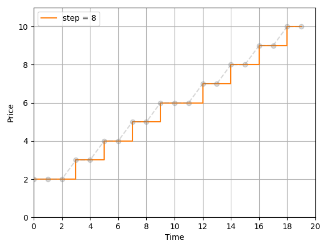

Figure 4.17: Graph of the fast step strategy used by a negotiator agent with the buyer role.

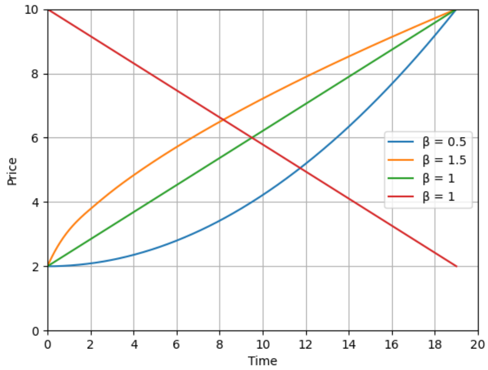

Figure 4.18: Graph of an example of 3 different concession strategies against a linear strategy.

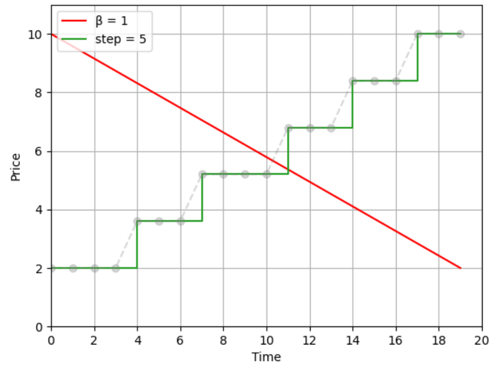

Figure 4.19: Graph of a fast step strategy against a linear strategy.

In the final step of the negotiation process, the bid for the first-price sealed bid auction is calculated using the following function:

$$Buyer\_BID = FP \cdot BID\_factor$$

Where:

- FP: final or limit price offered.
- Buyer_BID: is the value of the bid offered.
- BID_factor: value that defines how much the initial price is increased or decreased to set the bid. The default value is 0.75.

Interaction diagrams

This section explains the different interactions proposed for the market to work:

- **Market initialization**

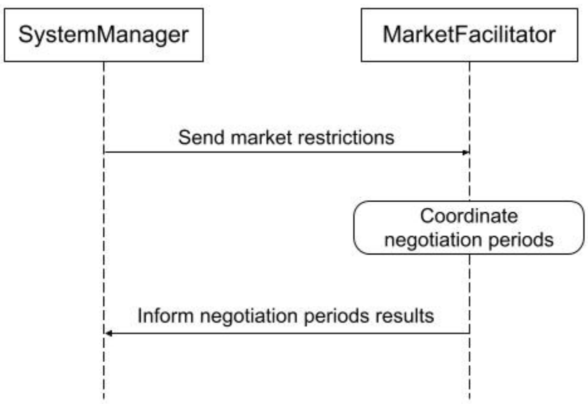

Figure 4.20: Interaction diagram displaying the market initialization.

When a system is executed the opening interaction that occurs is showed in diagram 4.20. In the firstplace, the system manager will send the market's restrictions. After receiving the restrictions, the market facilitator will be able to open the market and start coordinating the negotiation periods. After all negotiation periods are over, the results are sent to the system manager.

- **Negotiation period arrangement**

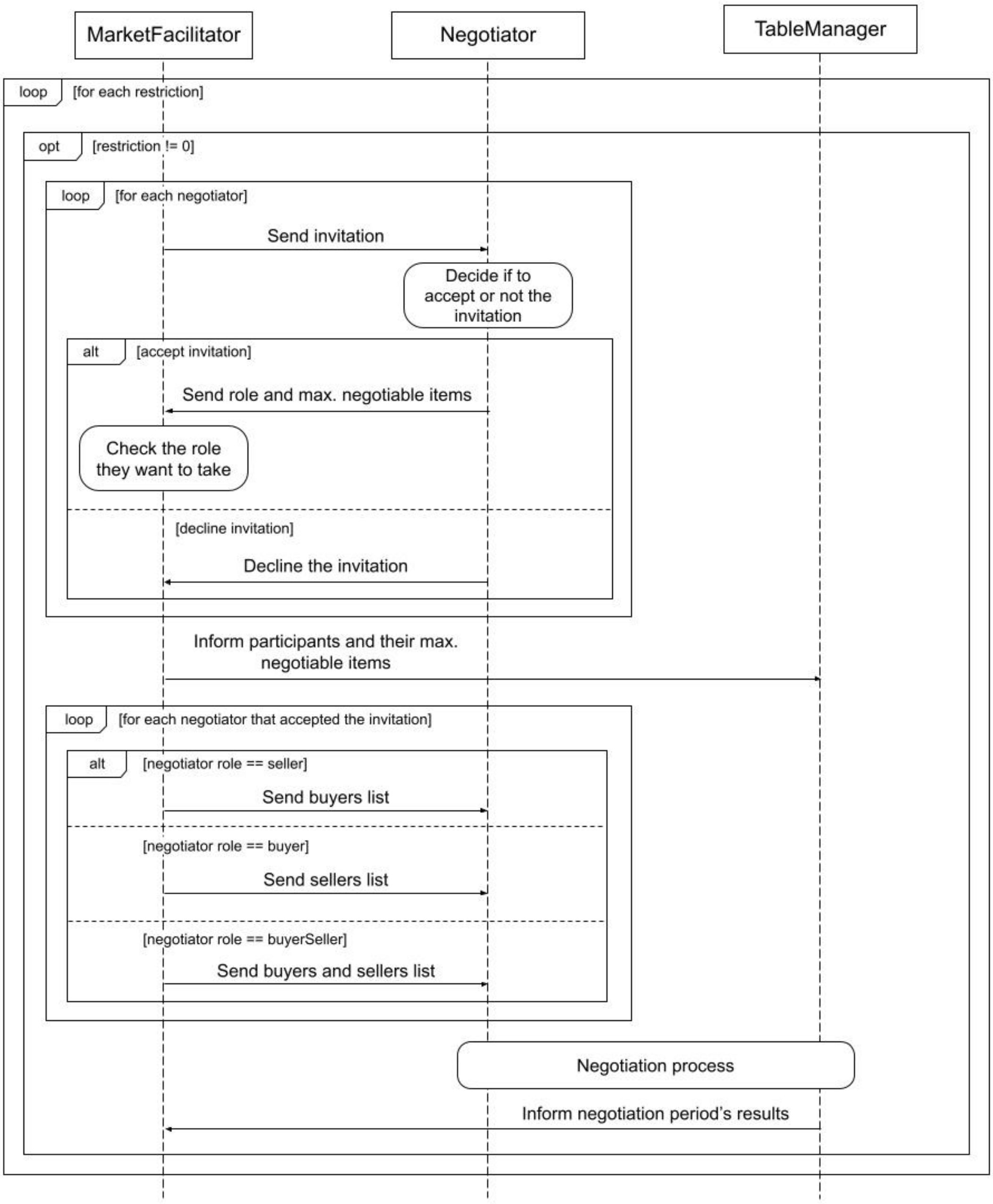

Figure 4.21: Interaction diagram displaying the market initialization.

Once the market facilitator is in the possession of the negotiation period's restrictions, it is possible to move onto the arrangement of the negotiation's periods. The market facilitator will send to all negotiators an invitation for each negotiation period with a non-zero restriction.

After a negotiator receives an invitation, it must decide if it is interested or not in taking part in the negotiation process. If a negotiator wishes to join the negotiation table, it will need to inform the market facilitator the conditions in which they want to enter the negotiation process. In other words, it should report the role it wants to undertake and its maximum negotiable items. his is required to verify negotiated transactions are possible. Additionally, the market facilitator verifies that negotiators are able to join the negotiation table with their stated characteristics.

On the contrary, if a negotiator is not interested in taking part in this negotiation period, it declines the invitation.

Once all the negotiators have answered their invitation or the time limit to join has come to an end, the market facilitator informs the table manager who will be joining the negotiation table, their roles and the maximum negotiable items. Immediately after, it notifies the negotiators who they will be able to negotiate with.

At the end of the negotiation process, the table manager lets the market facilitator know the events that have taken place during the negotiation period.

- **Bilateral negotiation period**

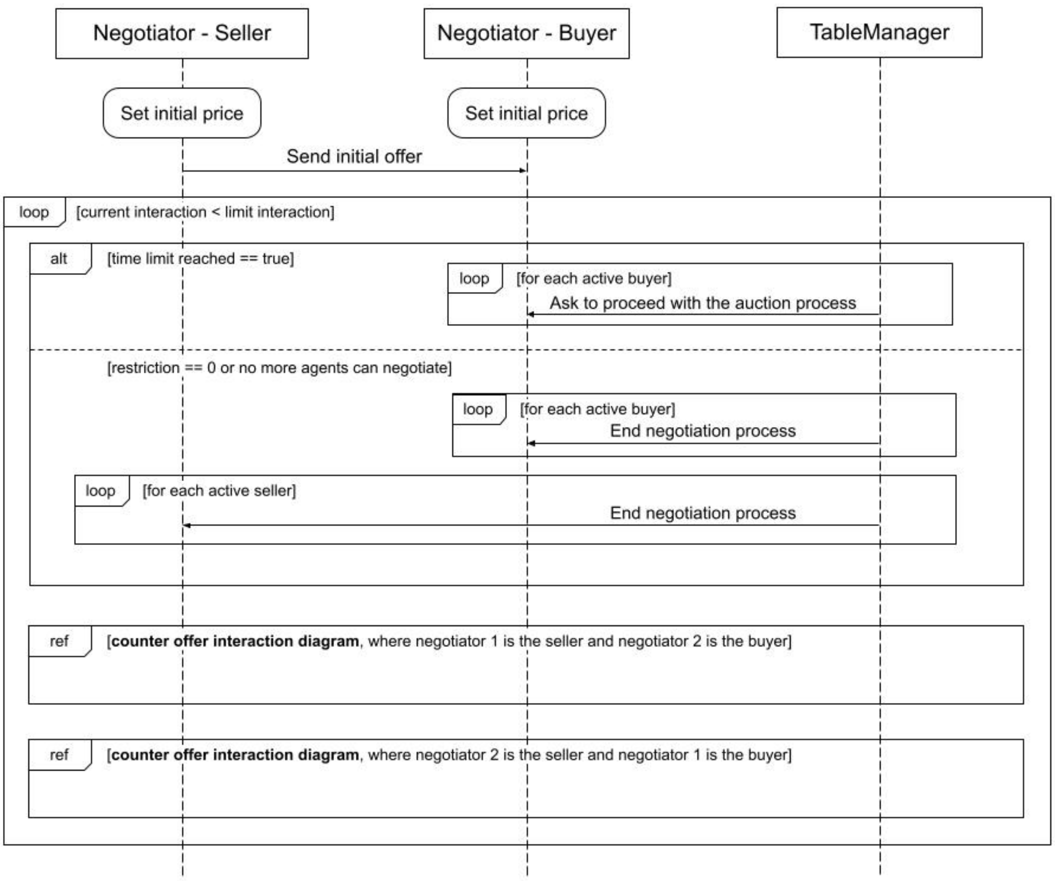

Figure 4.22: Interaction diagram displaying the bilateral negotiation process.

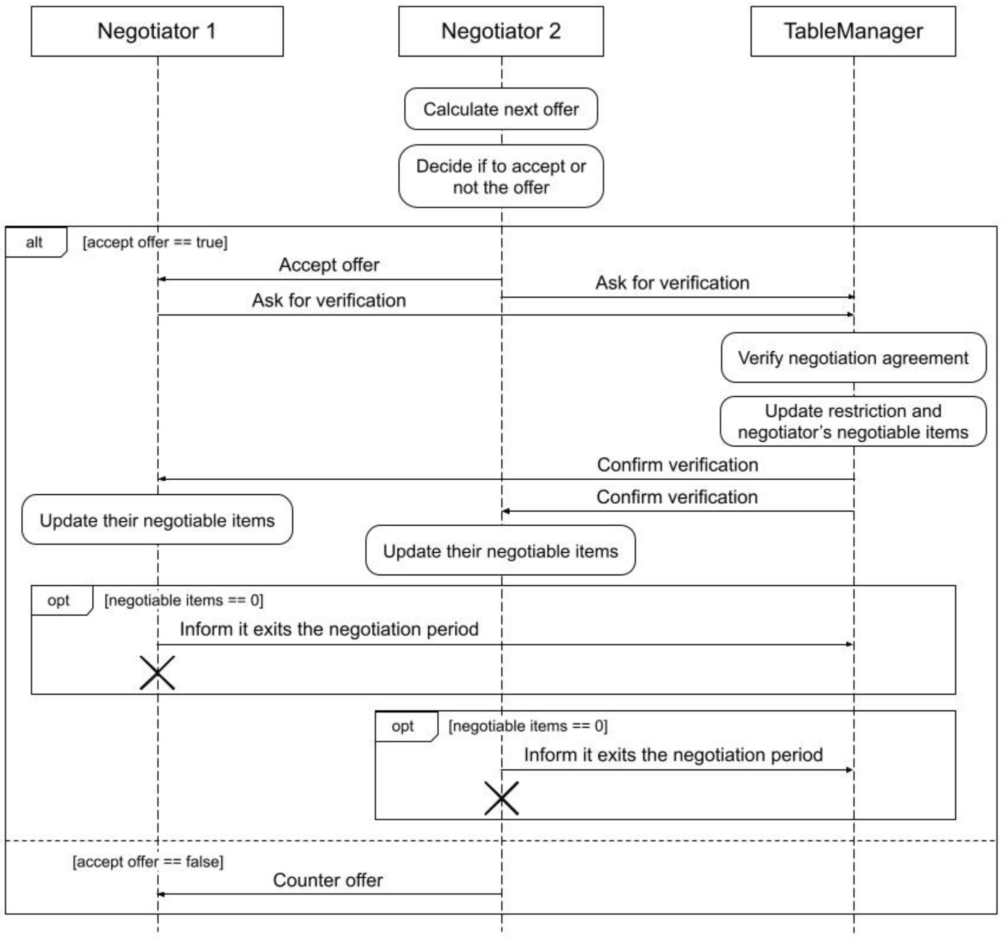

Figure 4.23: Interaction diagram displaying how offers are managed in the bilateral negotiation process. In this example, negotiator 2 is the one receiving an offer.

In order to simplify and offer a better understanding of the interaction diagram about the bilateral negotiation is has been divided into two figures. While figure 4.22 represents the bilateral negotiation process as a whole, figure 4.24 represents the interaction after a negotiator has received an offer. It is important to note that, in this figure, negotiator 1 and negotiator 2 can either be seller or buyer, but they cannot have the same sub-role. Additionally, figure 4.22 only shows the interaction between a seller, a buyer and the table manager. However, as mentioned previously, this interaction's cardinality would be n buyers, n sellers and one table manager.

At the beginning of this phase of the negotiation period, negotiators set their initial prices and sellers send the initial offers to the buyers. After this moment, negotiations and counter offering will begin.

Meanwhile, the table manager's duties are to ensure reached agreements are feasible, to end the negotiation period if the restriction limit is reached or there are no more agents able to continue negotiations and to move negotiations to the next phase when the time limit is over.

During the negotiations between agents, different scenarios ma take place. First, when an agent accepts an offer it must notify the counterpart and both of them must request the table manager to verify the agreement. After verification, the three agents update their information about their remaining negotiable items. If an agent has no negotiable agents left, it notifies the table manager and stops all other conversations. It would return to the negotiation and wait for the next invitation.

On the other hand, if the interaction limit between agents is not reached and agents do not agree with the offers received, they will continue to counter offer until they agree on a price or the period ends. Agents wait to receive an offer from all other sub-role agents before deciding to accept an offer or continue counter offering.

- **Auction period**

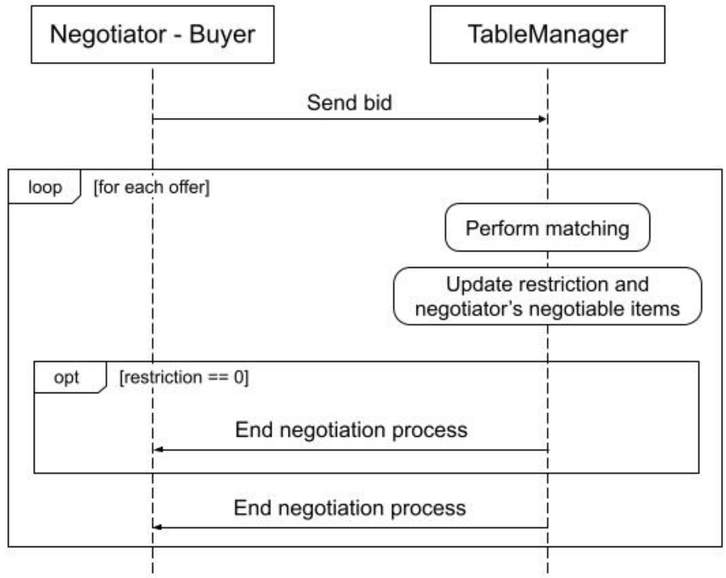

Figure 4.24: Interaction diagram displaying the auction phase.

At the end of the negotiation process, in the auction phase, all buyers send their bids to the table manager. Then, these bids are matched with the seller agents until the restriction reaches zero or all bids have been matched.

Input and output files

In order to carry out a market simulation it is indispensable to provide the system with a properties file. This file contains all the necessary information to run a virtual market. When the simulation is ended, the output will be a log file and a file used for the web visualization.

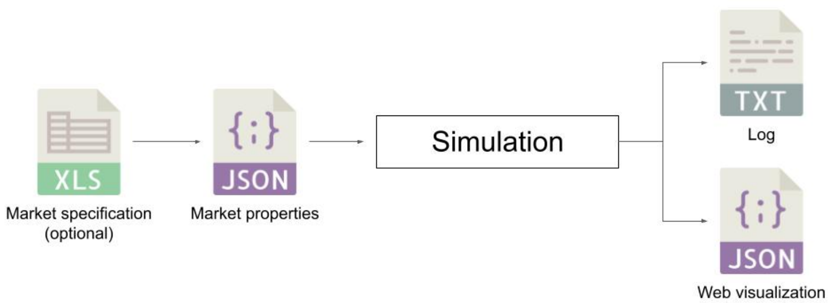

Figure 4.25: Image showing the input and output files of the simulation.

The different files used are explained in detail below:

- Input files:
  - **Market properties**: this JSON file specifies the input data of the market. It provides information for each restriction associated with a negotiation period. Also, regarding each agent it identifies its alias and their characterization for all the negotiation periods, the type of auction and the maximum number of interactions per negotiation. Additionally, it includes parameters required by SPADE to run the simulation.
  - **Market specification** (optional): this file is an optional and provisional way of specifying a virtual market. This solves the temporary problem of not having an interface to create agents for the simulation. Therefore, instead of writing by hand the previous JSON file, users can specify the market in a XLS file. It would be composed only ofs market related information. A XLS file would have one tab to state the market restrictions and a second tab to characterize all agents. This file can afterwards be parsed to create the market properties file.
- Output files
  - **Log**: this is a plain text file used for debugging as it saves all events of the simulation in a raw format.
  - **Web visualization**: this JSON file contains all the information needed to launch the web visualization. This file has all conversations between agents and the results of each negotiation period. The following section describes how this information will be displayed.

Using JSON format allows to have a standard format for the market specification and the web visualization file. This also allows other applications create the market properties file. Moreover, these files could also be read by other simulations or interpreted by other visualization tools.

4.3.2. Web visualizer

The web visualizer's design for the market proposed in the previous section is made up of 3 elements: the home page, the agent communications page and the results page. This section will show for each window their scheme, their functionality and how they fulfill the previously identified requirements.

A simple example is used to display the design of the web visualization tool.

Home page

The home page is the first page shown when the website is first loaded and the "home" button is pressed.

The home page is the first page shown when the website is loaded and the "home" button in pressed. This page is also referred to as the index page, front page or main web page of a website.

Figure 4.26: Image showing the home page's design.

Figure 4.26 displays the home page, where one can see the presentation of the application and the sidebar. The side bar highlights the current page and is appears in all other pages. The elements present in the sidebar are:

- **Logo button**: loads the home page.
- **Home tab**: loads the home page.
- **Agent communications tab**: loads the agent communications page.
- **Results tab**: loads to the results page.

Agent communications page

The agent communications page exhibits all the messages sent between agents. The layout of this page is made up of 3 columns:

- **Agent list**: contains all the agents taking part in the system.
- **Contact list**: is the set of agent who the selected agents has been interacting with.
- **Chat**: Displays the whole conversation between two agents.
  - Messages are color coded and indicate the sender.
  - Messages are separated by the period's name in order to distinguish different conversations.

Examples of the page's design are shown below:

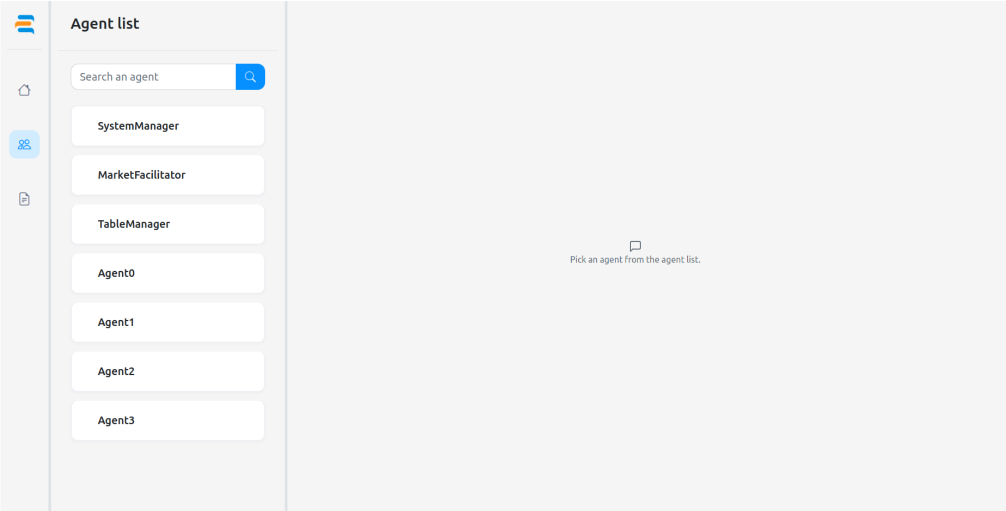

Figure 4.27: Image showing the agent communications page's design.

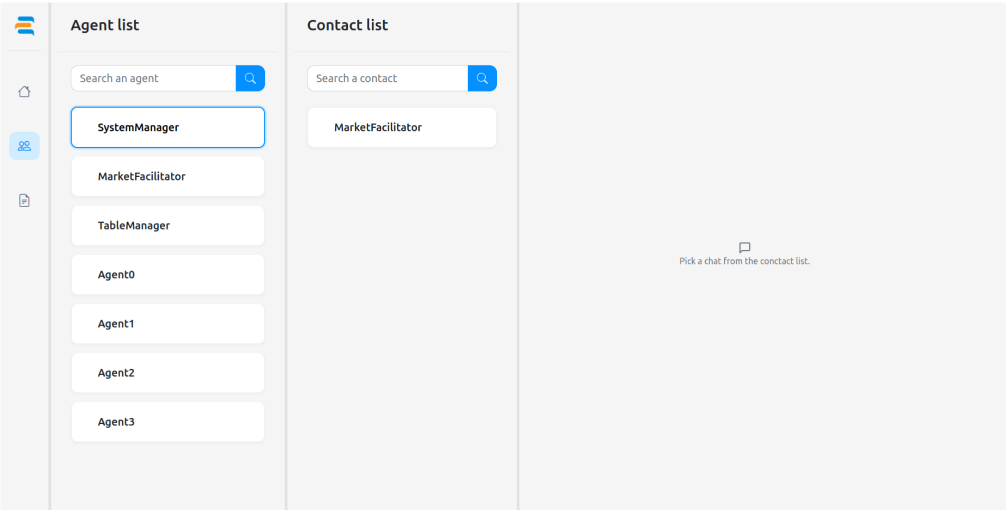

Figure 4.28: Image showing the system manager's contacts list.

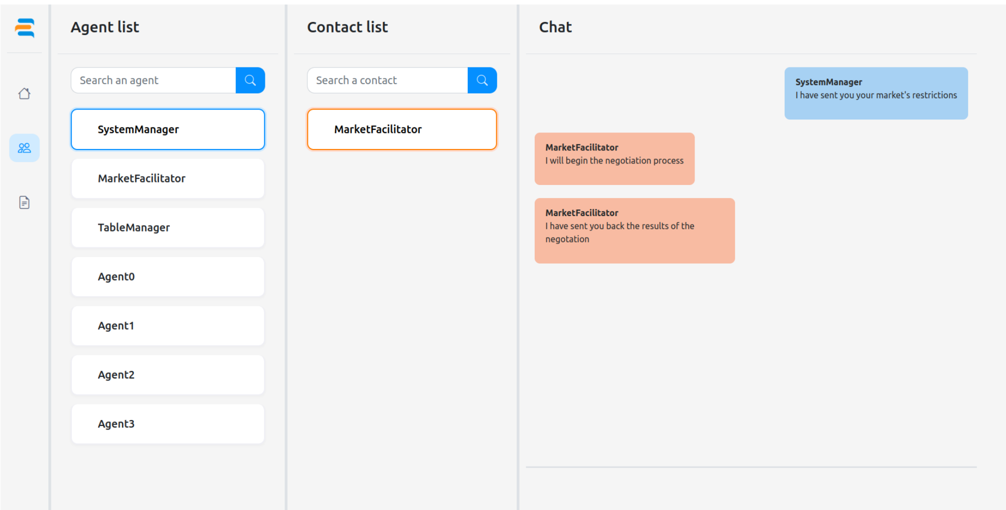

Figure 4.29: Image showing the conversation between the system manager and the market facilitator.

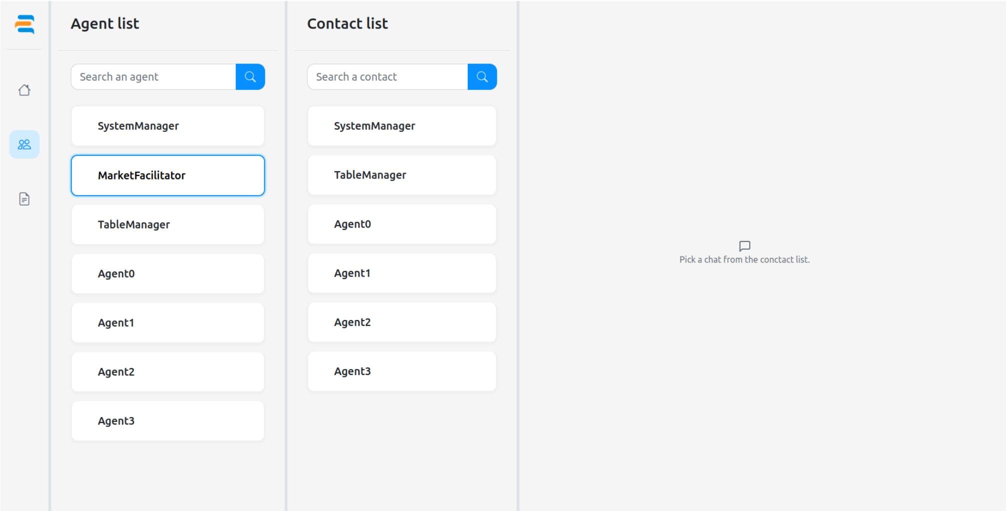

Figure 4.30: Image showing the market facilitator's contacts list.

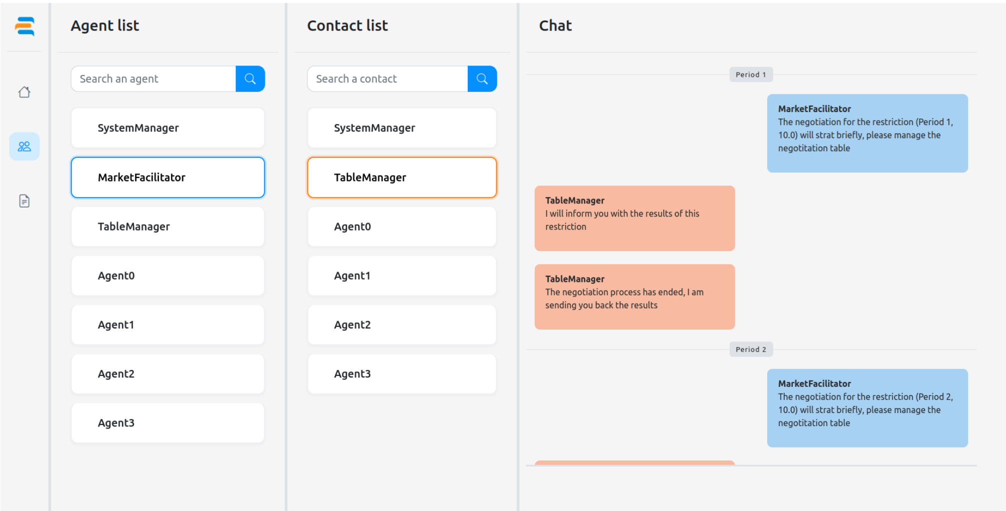

Figure 4.31: Image showing the communication between the market facilitator and the table manager.

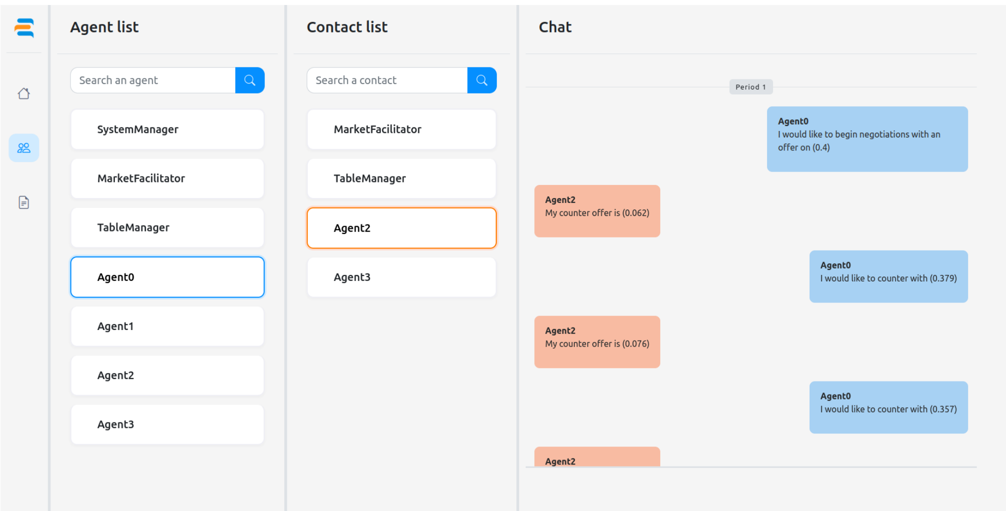

Figure 4.32: Image showing the conversation between the Agent0 and Agent2.

Results page

The results page shows the outcome of each negotiation process.

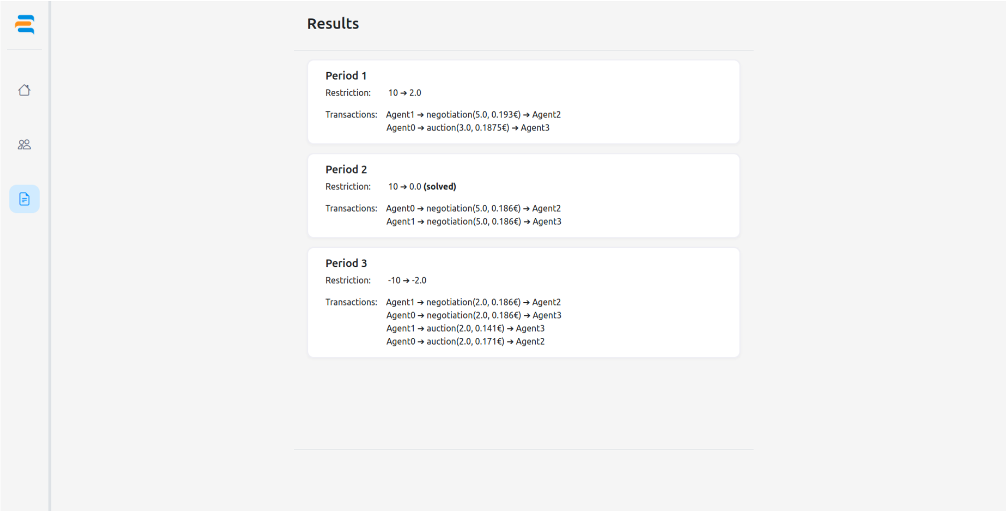

Figure 4.33: Image showing the results page's design.

Every element in the result list contains:

- **Period name**
- **Restriction**: displays the progress of the period's restriction at the end of the negotiation process. Additionally, solved restrictions are tagged as solved.
- **Transactions**: displays, schematically, what has occurred during the negotiation process. Each transaction has these elements from left to right: seller, agreement, quantity agreed, price agreed, buyer. Arrows show the flow of goods from the seller to the buyer.

After completing the design phase in this section, it is possible to confirm all the considered requisites displayed in the analysis section have been taken into account and have been fulfilled. In the validation section, experiments will be carried out to confirm the design's correct behavior and the implementation of the generic virtual market simulation software.

4.4 Implementation

This section describes the most important aspects of the implementation carried out for this project for the market and visualizer. As these applications have been developed independently, the market and the web visualizer will be explained separately.

4.4.1. Market

The proposed market, a multi-agent system, has been implemented, as explained in section 3.4, using the SPADE framework. This section will explain how different aspects have been implemented, particularly its structure, the communication between agents and additional functionality.

Structure

In order to comply with the design detailed in the point 4.3.1, different agents have been created to satisfy each identified role. These have been defined independently in a python *.py* file: *systemManager.py*, *marketFacilitator.py*, *tableManager.py* and *negotiator.py*. Each one of the developed agents has the behavior and reasoning explained in the design phase.

To launch a simulation, the *system.py* file needs to be executed. It reads the JSON configuration file and starts the simulation.

At the start of a simulation, the agents from the class *negotiator.py* are just template which are characterized for each negotiation period. This characterization comes from the input file of the simulator displayed in figure 4.25. These agents may have different roles in a negotiation depending of what they can offer. Their behavior is distinguished by their negotiation strategy and their price offerings.

Moreover, the market properties' file includes all the details related to the simulation. They do not only include the agents' specification, but also the restrictions in the market, the type of auction and the maximum number of iterations in the bilateral negotiation.

This implementation allows to have an only file containing the whole specification of a system. This may be considered a convenient solution to unify all the input of a simulation.

Communication between agents

In multi-agent systems, communication between agents are a crucial aspect. The messages sent between agents is internally manage by SPADE with the methods *.send()*. This function allows to send the performative of the message, indicates its sender and allows to send any desired content as JSON data. This is really useful as the content to be sent is completely customizable.

Moreover, agents can have different behaviors implemented and being executed concurrently. Each behavior has its own inbox queue of messages. This can be clearly seen in diagram 3.3. It is convenient to have different behaviors as it is the only way to prioritize which messages are read first. Concretely, negotiators have different behaviors depending on who they are communicating with. This is required so when a negotiation period ends, they can need to immediately cease negotiations when notified.

Additional functionality

Most of the additional functionality has been implemented in a file called *utils.py*. The methods to reason prices and the negotiation strategy functions are written in this file. This includes how initial prices are calculated, the bilateral negotiation strategy functions and how the bids are placed.

SPADE does not have sniffer agents natively implemented like JADE. However, agents have an object called *traces* that can hold up to 1,000 sent and received messages. This attribute may be considered a substitute of a sniffer agent. In order to save the agents' conversations, they implement a behavior that empties the agent's *traces* and saves them in web visualization file periodically and at the end of the simulation.

On the other hand, the log file is created in a different way. It uses the *logging* python library to save the interactions that have taken place, so that the events can be later debugged. This information is saved in a plain text file. In comparison to the web visualization file, the log file includes references to the moment were agents are instantiated and other cues that indicate the state of the agents, such as if its waiting to receive messages.

4.4.2. Web visualizer

In order to implement the web visualizer, Flask and Bootstrap have been chosen as the web framework and CSS framework, respectively.

Flask is considered a *micro* Framework which allows the development of web applications in Python. It is considered *micro* as it has the bare bones to create a web application. Additional functionality can be added by the use of plugins.

On the other hand, bootstrap is an HTML, CSS and JavaScript Library that focuses on simplifying the development of web pages. This library provides the basic style definition for the whole application. In addition, its components are responsive and mobile-first, making easier the implementation of any project.

Both frameworks focus on fast and efficient development. Therefore, it is possible to deliver an elegant solution in short time.

Furthermore, this section will cover the web visualizer's most relevant features:

Visualization input

The web application is a hollow container which can display a simulation's results. Therefore, it is essential to have the means to obtain this information. As stated previously, in the market's design section, the system outputs the file needed for the visualization.

This file is in JSON format. This allows to have a standard input format, so that other software can read this file and interpret it.

The input is made up of two elements: the agents' conversations and the results. These contain all the information about the messages every agent has sent and received as well as the details of all the negotiation periods, displaying the agents involved and the reached agreement.

Templates

One of Flask's key components is Jinja, a web template engine for Python. It is a text-based template language and, thus, can be used to generate any markup as well as source code. A template is a HTML file with a specific format. It allows one to prepare dynamic web pages that are rendered with specific information.

The application's templates are created in Jinja2 format, which allows the rendering process. This made possible to have variables that come from the web's input file as well as the development of modular pieces of reusable code. Therefore, it was possible to build a maintainable, clear and efficient application.

4.5 Validation

After the implementation of any software, it is essential to confirm that the developed solution works as expected. In consequence, this section will display the validation tests performed to check that the simulations behave accordingly to its design.

4.5.1. Validation tests

In order to carry out the validation of the generic market simulator several experiments have been conducted. These experiments will cover all the simulator's implementation to verify the software works as expected.

The configuration of the validation tests has some static parameters. The simulated market is composed of 7 negotiation periods with a different restriction for each one. This is shown in table 5.2. Additionally, the maximum number of interactions during the bilateral negotiation is set to 20.

Table 4.1: Table of the restrictions in each period.

Period	Restriction
1	-inf
2	-15
3	-5
4	0
5	5
6	15
7	inf

Altogether, there will be 2 buyers, 2 sellers and 2 seller-buyers taking part in the market. Their individual negotiable items and price limits are specified in table 5.3.

Table 4.2: Table of the constant parameters of the negotiator agents.

	Agent 0	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Sell capacity	10	5	0	0	10	5
Min. sell price	15	23	-	-	17	22
Buy capacity	0	0	10	5	10	5
Max. buy price	-	-	20	27	30	22

Furthermore, several tests have been designed to check the functionality of all the strategies. To begin with the software verification, a base case has been established. Next, 6 test have been carried out to confirm the strategies provide the expected behavior. All the tests are specified in table 4.3.

Table 4.3: Table of the price reasoning strategies used in each experiment.

Test	Strategy
Base case	Linear
1	Aggressive
2	Conservative
3	Mixed
4	Linear-Slowstep
5	Linear-Step
6	Linear-Faststep

Where:

- **Linear**: All negotiators have a linear strategy.
- **Aggressive**: All negotiators have an aggressive strategy.

- **Conservative**: All negotiators have a conservative strategy.
- **Mixed**: Agents 0 and 2 have a conservative strategy. Agents 1 and 3 have an aggressive strategy. Meanwhile, the seller-buyers agents, 4 and 5, have a linear strategy.
- **Linear-Slowstep**: Buyers have a linear strategy and the remaining have a step strategy.
- **Linear-Step**: Buyers have a linear strategy and the remaining have a slow step strategy.
- **Linear-Faststep**: Buyers have a linear strategy and the remaining have a fast step strategy.

4.5.2. Results

Base case - Linear strategy

It is important to establish a base case so that tests can be compared to a simple and common scenario, where the negotiation strategy has not been altered by the results of the negotiation. Table 4.4 shows the agreements reached during this test.

Table 4.4: Table that shows the results from the base case test.

Period	Restriction	Agreement				
		Type	Seller	Buyer	Quantity	Price
1	-inf	negotiation	Agent 0	Agent 3	5	22.661
		negotiation	Agent 5	Agent 4	5	28.393
		negotiation	Agent 4	Agent 5	5	20.821
		negotiation	Agent 0	Agent 4	5	24.643
		negotiation	Agent 4	Agent 2	5	18.929
		auction	Agent 1	Agent 2	5	15.0
2	-15	negotiation	Agent 0	Agent 3	5	22.661
		negotiation	Agent 5	Agent 4	5	28.393
		negotiation	Agent 4	Agent 5	5	20.821
3	-5	negotiation	Agent 0	Agent 3	5	22.661
4	0	-	-	-	-	-
5	5	negotiation	Agent 0	Agent 3	5	22.661
6	15	negotiation	Agent 0	Agent 3	5	22.661
		negotiation	Agent 5	Agent 4	5	28.393
		negotiation	Agent 4	Agent 5	5	20.821
7	inf	negotiation	Agent 0	Agent 3	5	22.661
		negotiation	Agent 5	Agent 4	5	28.393
		negotiation	Agent 4	Agent 5	5	20.821
		negotiation	Agent 0	Agent 4	5	24.643
		negotiation	Agent 4	Agent 2	5	18.929
		auction	Agent 1	Agent 2	5	15.0

As expected, period 3 has no agreements. This is due to the fact that a negotiation process is not arranged for zero restriction periods. This behavior occurs in all other tests.

Moreover, if the absolute value of 2 restrictions is the same, they will have the same interaction. The sign of a restriction only characterizes which transactions are favored, keeping the results the same. Figure 4.4 verifies this behavior as it is possible to see how the agreements reached for negative and positive restrictions remain the same.

All tests have been run through all periods. The base will show the complete results. However, due to periods 1-4 not adding any new information, the results for tests one to six will only show periods 5, 6 and 7.

Test 1 - Aggressive strategy

As explained in section 4.3.1, the goal of an aggressive behavior strategy is to reach agreements faster at the expense of paying a higher prices. Table 4.5 shows how the agreements have been the exact same as in the base case for the exception of the agreed price, which nearly in all cases is higher.

Table 4.5: Summarized table of the results from test 1.

Period	Restriction	Agreement				
		Type	Seller	Buyer	Quantity	Price
5	5	negotiation	Agent 0	Agent 3	5	22.931
6	15	negotiation	Agent 0	Agent 3	5	22.931
		negotiation	Agent 5	Agent 4	5	28.446
		negotiation	Agent 4	Agent 5	5	21.205
7	inf	negotiation	Agent 0	Agent 3	5	22.931
		negotiation	Agent 5	Agent 4	5	28.446
		negotiation	Agent 4	Agent 5	5	21.205
		negotiation	Agent 0	Agent 4	5	24.259
		negotiation	Agent 4	Agent 2	5	19.277
		auction	Agent 1	Agent 2	5	15.0

Test 2 - Conservative strategy

On the contrary, the aim of a conservative strategy is to minimize changes in the initial price offered instead of reaching an agreement faster. This often leads to deals being settled at lower prices. Table 4.6 displays this behavior. As discussed in the previous case, the agreements reached are the same, but the agreed prices are lower than in the base case.

Table 4.6: Summarized table of the results from test 2.

Period	Restriction	Agreement				
		Type	Seller	Buyer	Quantity	Price
5	5	negotiation	Agent 0	Agent 3	5	21.628
6	15	negotiation	Agent 0	Agent 3	5	21.628
		negotiation	Agent 5	Agent 4	5	26.901
		negotiation	Agent 4	Agent 5	5	19.727
7	inf	negotiation	Agent 0	Agent 3	5	21.628
		negotiation	Agent 5	Agent 4	5	26.901
		negotiation	Agent 4	Agent 5	5	19.727
		negotiation	Agent 0	Agent 4	5	24.031
		negotiation	Agent 4	Agent 2	5	17.934
		auction	Agent 1	Agent 2	5	15.0

Test 3 - Mixed strategy

The purpose of this test was to prove that the outcome obtained from the interaction between agents that have different price reasoning strategies is completely different to the one where all agents share the same strategy. This test should have clear distinct agreements from previous tests.

Table 4.7: Summarized table of the results from test 3.

Period	Restriction	Agreement				
		Type	Seller	Buyer	Quantity	Price
5	5	negotiation	Agent 5	Agent 4	5	28.393
6	15	negotiation	Agent 0	Agent 5	4	28.393
		negotiation	Agent 4	Agent 5	5	20.821
		negotiation	Agent 0	Agent 3	5	21.315
7	inf	negotiation	Agent 0	Agent 5	4	28.393
		negotiation	Agent 4	Agent 5	5	20.821
		negotiation	Agent 0	Agent 3	5	21.315
		negotiation	Agent 0	Agent 2	5	17.934
		negotiation	Agent 1	Agent 4	5	26.786
		negotiation	Agent 4	Agent 2	5	17.934

Test 4 - Linear-Slowstep

Table 4.8: Summarized table of the results from test 4.

Period	Restriction	Agreement				
		Type	Seller	Buyer	Quantity	Price
5	5	negotiation	Agent 4	Agent 5	5	17.0
6	15	negotiation	Agent 4	Agent 5	5	17.0
		negotiation	Agent 5	Agent 4	5	22.5
		negotiation	Agent 1	Agent 4	5	30.0
7	inf	negotiation	Agent 4	Agent 5	5	17.0
		negotiation	Agent 5	Agent 4	5	22.5
		negotiation	Agent 1	Agent 4	5	30.0
		negotiation	Agent 0	Agent 3	5	21.429
		negotiation	Agent 0	Agent 2	5	20.0
		negotiation	Agent 4	Agent 2	5	17.0

Test 5 - Linear-Step

Table 4.9: Summarized table of the results from test 5.

Period	Restriction	Agreement				
		Type	Seller	Buyer	Quantity	Price
5	5	negotiation	Agent 4	Agent 5	5	22.0
6	15	negotiation	Agent 4	Agent 5	5	22.0
		negotiation	Agent 5	Agent 4	5	30.0
		negotiation	Agent 1	Agent 3	5	27.0
7	inf	negotiation	Agent 4	Agent 5	5	22.0
		negotiation	Agent 5	Agent 4	5	30.0
		negotiation	Agent 1	Agent 3	5	27.0
		negotiation	Agent 4	Agent 2	5	17.0
		negotiation	Agent 0	Agent 4	5	26.714
		auction	Agent 0	Agent 2	5	15.0

Test 6 - Linear-Faststep

Table 4.10: Summarized table of the results from test 6.

Period	Restriction	Agreement				
		Type	Seller	Buyer	Quantity	Price
5	5	negotiation	Agent 4	Agent 5	5	19.938
6	15	negotiation	Agent 4	Agent 5	5	19.938
		negotiation	Agent 5	Agent 4	5	27.188
		negotiation	Agent 1	Agent 4	5	27.188
7	inf	negotiation	Agent 4	Agent 5	5	19.938
		negotiation	Agent 5	Agent 4	5	27.188
		negotiation	Agent 1	Agent 4	5	27.188
		negotiation	Agent 0	Agent 3	5	21.938
		negotiation	Agent 0	Agent 2	5	18.125
		negotiation	Agent 4	Agent 2	5	18.125

Step strategies were expected to have similar results to the ones obtained from the base case, test 1 and test 2. However, these differed not only between themselves but also from the base case. Test 4 and 6 have a similar behavior to test 3, while results from test 5 are more alike to the base case.

This shows how the outcome of a market is influenced by the way agents reason their princing and negotiation strategy. This may lead to unexpected results.

Finally, this validation has shown how the market is correctly opened and that the interactions performed comply with the design carried out. The negotiation periods are arranged correctly and negotiators are able to correctly undertake the buyer, the seller and the seller-buyer sub-role.

It is interesting to note that due to the parametrization of agents, agent 2 was the one to reach an agreement in the auction period. Additionally, in all periods of all tests, either the result were solved or agents were able to negotiate an agreement. In the end, this resulted in every agent using all their selling and buying capacities. Moreover, no agreement had an agreed quantity above 5. This is due to the fact that agents 0, 2 and 4 reached a settlement earlier with other agents than between themselves.

Furthermore, all results show that the smaller the restriction is in absolute value less agreements may be reached. On the other hand, as a restriction increases in absolute value the same interactions take place but additional agreements may occur.

After the completion of all tests, it is possible to conclude that the prototype works as expected and that negotiations strategies greatly affect the results obtained.

CHAPTER 5
Case study: Local electricity market

One of the best ways to validate any kind of software is by exposing it to real-life examples. In this case, the proposed generic market simulator is tested with a case study provided by the Institute of Energy Engineering of the *Universitat Politècnica de València* [61].

Throughout this chapter we will explain how the developed simulator has been used to gain further understanding about electricity markets. Moreover, different aspects of the simulator will be detailed, such as the requirements it had to be met and the features that helped to solve the conditions stated in the case study. Furthermore, the need to design additional features to satisfy unsolved requirements will be emphasized and explained. To end this chapter, the scenarios provided for the simulations and their outcome will be described.

5.1 Introduction

In economic terms, electricity is a consumable energy resource. In other words, it is a good that can be bought, sold and/or traded. An electricity market, also known as a power exchange, is a system that enables to purchase power packages through bidding or a pricing agreement. The price is set by supply and demand principles.

The commodities offered in an electricity market can be power or energy. Power is the metered net electrical transfer rate at any given moment and is measured in megawatts (MW). Energy is electricity that flows through a metered point for a given period and is measured in megawatt hours (MWh).

During the past years, there has been an increase in the number of renewable electricity generation sources. Nevertheless, there is a limited amount of electricity generating resources, which leads to high financial costs due to the way the system's energy is distributed. Therefore, with a correct optimization of the energy redistribution, it could be possible to reduce loses of unused electricity packages, electricity costs and redistribute the financial benefits across the system's participants. These electricity generation sources, also called flexible energy resources, can modify electricity production or consumption in response to variability in the system's electricity demands.

The Institute of Energy Engineering's work aims to develop a new local flexibility electricity market. For that matter, this tool is designed to improve the electrical system in favor of the final consumers. This is achieved by optimally exploiting flexible energy resources while maximizing the financial benefits for the agents involved in the local electricity market. Specifically, this would involve the minimization of costs incurred in supplying electricity to consumers and the consequent benefit for agents with flexible capacity in their energy resources.

These flexible energy resources are limited to local areas and have smaller capabilities in comparison to traditional energy resources. Therefore, the urgent design of new market structures and platforms is necessary to negotiate local energy resources.

To this end, this work's simulator will be used to obtain further insights about the flexible electricity local market proposed by the Institute of Energy Engineering. This group has specified several case scenarios in a local electricity market in order to examine the emergent behavior in the different simulations. Moreover, these scenarios have a different parametrization to analyze its effects on the results obtained. The agents and the local market involved will be presented anonymously.

5.2 Local electricity market's simulation

In this section the proposal's simulation requirements will be stipulated and analyzed. This will be achieved by identifying the features already included in the current version of the simulator and the ones that need to be included for this specific case study. Afterwards, the design and implementation of these new elements will be presented. Finally, the data used and the simulation results will be displayed.

5.2.1. Proposal's requirements

The flexible electricity market proposed by the Institute of Energy Engineering is a tool to address technical restrictions in the electric system. Restrictions arise from the result of previous electricity markets and lead to the daily market. In this market, there can be three types of restrictions:

- No restriction: there is no restriction so the local market does not need to open.
- Energy surplus restriction: the local market's objective is to reduce the energy surplus in the system.
- Energy deficit restriction: the local market's objective is to alleviate the system's energy shortage.

This proposal is aimed at an intra-daily market. This market appears in consequence of the results obtained in daily electricity market from the day before. The proposed intra-daily market is composed of two consecutive phases. The first phase consists of a bilateral negotiation where buyers and sellers can make offers in both directions for energy packages. It is aimed to be a competitive n to n negotiation.

The second, and last, phase is an auction. It is important to note that it is not mandatory to reach this phase, as the objective of the market is to solve the restriction at hand. Therefore, once enough agreements have been reached to solve the restriction the period stops. Additionally, in an energy deficit restriction besides the normal exchange of energy packages, some agents can be paid to reduce their energy consumption. As, in the end, this would also be a viable way to solve the energy shortage during a period.

The different roles that take part in the market are explained below:

- System operator: this role's responsibility entails the supervision of the electric network and the electricity markets. Therefore, it must validate the transactions that occur in any given market. Additionally, it is the one who identifies the mentioned market restrictions and who may take part in the market. Furthermore, it needs to ensure all transactions performed are technically viable.

  Moreover, this proposal wants to liberalize, to some extent, the opened market. The system operator will only take part if it is essential. Therefore, it will not intervene during sessions except to validate agreements and during the negotiation of the energy consumption reduction in energy deficit restriction periods. This action is unavoidable as it is the market's only participant that can offer a reward for it. In addition, in order to reduce the influence of this type of negotiation it is only performed during the auction.

- Market operator: this person is whom the system operator assigns the responsibility of arranging the market's sessions. Every one of this sessions corresponds to a non-zero restriction period. Furthermore, it manages the participants for a negotiation period. This agent only supervises the sessions, it does not take part in the negotiations.

  In addition, it verifies reached agreements as it makes sure they help to reduce the current restrictions. Afterwards, these agreements are sent to the system operator so that their technical feasibility can be verified. As agreements are completely verified, the market operator will update the current restriction until the end of the period. If it is solved before the period ends, the market session must be closed.

- Consumer: these entities can modify their energy consumption in order to maximize their profit.

- Generator: these entities have a flexible electricity generation source. They will be able to sell energy packages while trying to maximize their profit.

- Prosumer: this entity has both the consumer's and the generator's properties.

The five roles presented have a clear hierarchical relation between them. At the top of the hierarchical pyramid stands the system operator, who is responsible of the system as a whole. Below him, one can find the market operator, who ensures everything inside the market runs smoothly . At the bottom of the pyramid, the market's participants, consumers, generators and prosumers, are found. Ultimately, consumers can be considered buyers, generators can be referred to as sellers and prosumers undertake both roles at the same time.

Moreover, in order to enter a negotiation session, negotiators may comply with the entry requirements. The energy offered to buy or sell must not worsen the current session's restriction. Agents have two types of generation and consumption: the one expected by the system, the one calculated by the system operator and the additional generation or consumption not taken into account by the system operator. Normal generation and consumption capabilities will always be accepted. However, in excess periods only extra consumption will be admitted. On the contrary, in deficit periods only extra generation will be allowed. The type of energy generated or consumed will always be exclusive. An entity will have normal or extra capabilities to generate or consume energy. In other words, no participant will be able to negotiate with normal and extra generation or consumption of energy. This will, ultimately, condition the roles taken in the market. Therefore, these conditions must be taken into account in the session's registration process.

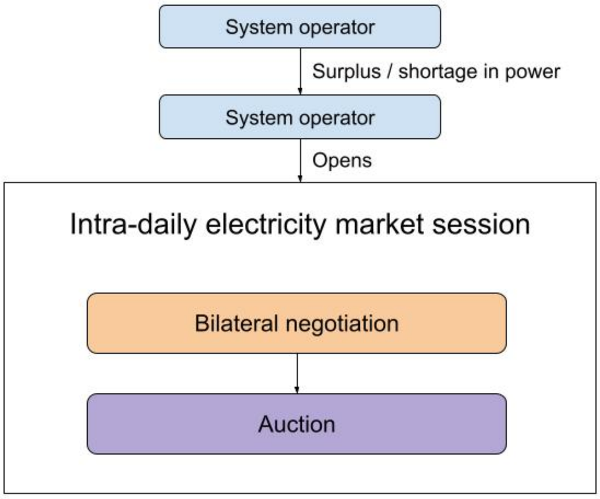

Figure 5.1: Diagram of the market's design.

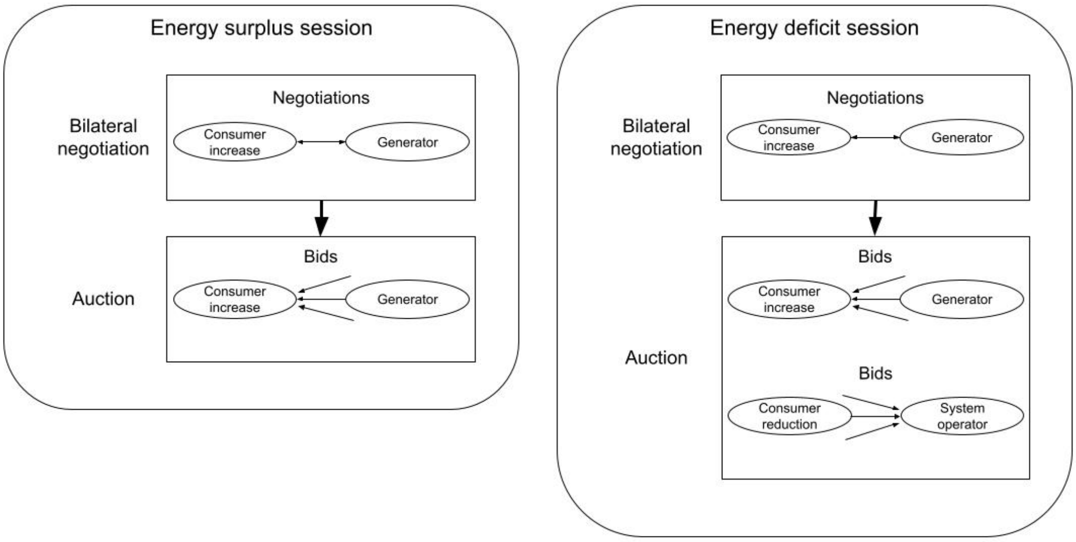

Figure 5.2: Image showing the surplus and shortage sessions' outline.

5.2.2. Analysis

In this section a comparison between the current simulator and the local market proposal will be analyzed. This comparison will not only include the existing features but also emphasize the additional functionality that needed to be implemented.

In first place, the market's negotiation process is exactly the same as the generic market simulator, where a bilateral negotiation is followed up by an auction. Restrictions can be specified for each period. The system manager, market facilitator and negotiators match very similarly with the system operator, market operator and agents taking part in electricity markets.

However, the main difference relies in the table manager and the negotiators. The system manager and market facilitator lack some of the responsibilities that the system operator and market operator have. Despite of this, the table manager takes on these tasks. Such as verifying transactions and updating the session's restriction.

The system manager and market facilitator do not exactly fit the responsibilities a system operator and market operator have. This is because the table manager is the one responsible for verifying transactions and updating the session's restriction. Therefore, this requirement is accomplished but by another actor. Therefore, it will be the agent the one with whom agents will negotiate energy reduction prices. Moreover, more characterization will be needed from negotiators to indicate their generation, consumption and reduction capabilities.

Furthermore, the negotiation strategies will be the ones available and do not vary for this case study.

The new features the simulator must include are the following:

- Negotiators must specify the energy they can generate, consume and/or reduce, with their limit price, at each period. Additionally, they must specify the type of generation and consumption energy they are able to offer.
- The invitation process must take into account the new negotiator's features.
- Consumers and the table manager must be able to negotiate energy reduction offers with the negotiators.
- The table manager must include the reduction offers in the matching process during the auction phase.

5.2.3. Design

This section will describe and discuss all the design changes carried out in order to adapt the current simulator to the case study.

In the local electricity market, the registration process a bit more complex than the generic market due to the fact that there are more conditions that have to be taken into account before a negotiator undertakes a role. The interaction done during a registration will not change. However, the reasoning behind this now takes into account the type of generation and consumption as well as the ability to reduce their consumption.

Table 5.1 displays the conditions to be for a negotiator to undertake a role in a surplus or deficit session.

Table 5.1: Conditions to be met in order to undertake a role in a surplus or deficit session.

Role	Generation	Gen. type	Consumption	Con. type	Reduction
	Surplus session				
Prosumer	> 0	normal	> 0	either	-
Generator	0	normal	0	-	-
Consumer	0	-	> 0	either	-
	Deficit session				
Prosumer	> 0	either	> 0	normal	≥ 0
Prosumer	> 0	either	≥ 0	normal	> 0
Generator	> 0	either	0	normal	0
Consumer	0	-	≥ 0	normal	> 0
Consumer	0	-	> 0	normal	≥ 0

In order to comply with the new analyzed requirements, the table manager, the seller and the buyer roles have been updated. Firstly, the table manager now has the necessary knowledge to know when a reduction offer is acceptable. Secondly, sellers and buyers now know the type of energy consumption or generation they can offer. Lastly, buyers are able to perform reduction offers. To make it easier to see, the changes made to each of these roles has been highlighted.

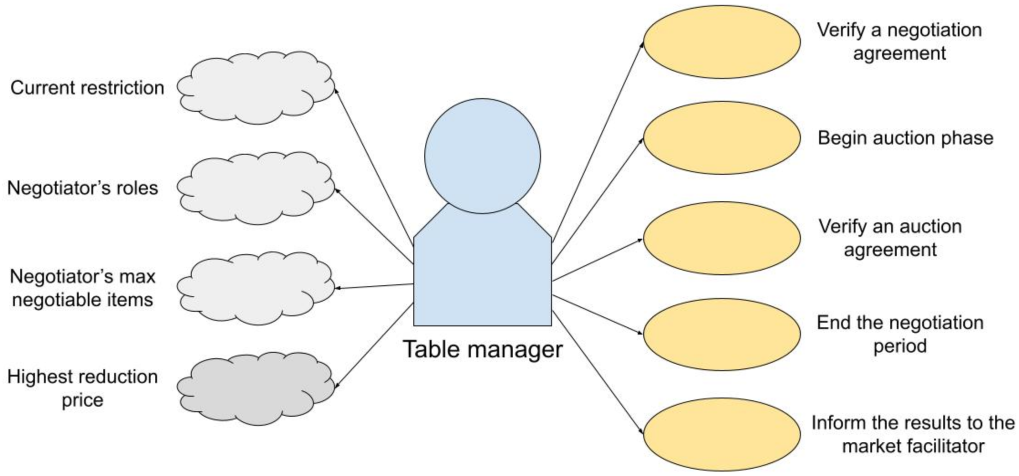

Figure 5.3: Diagram of the updated table manager role.

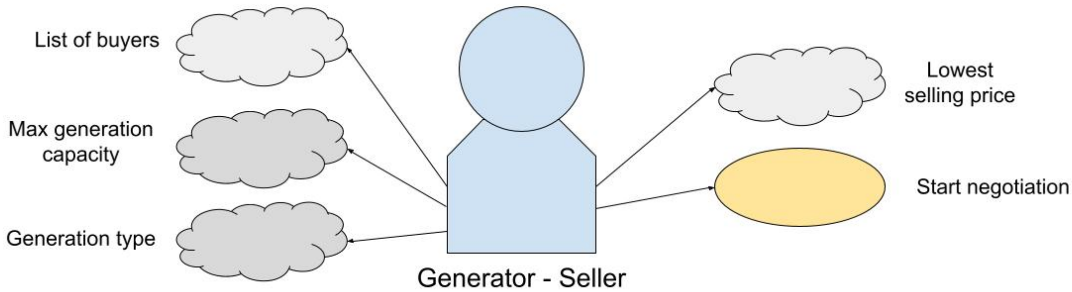

Figure 5.4: Diagram of the updated seller role.

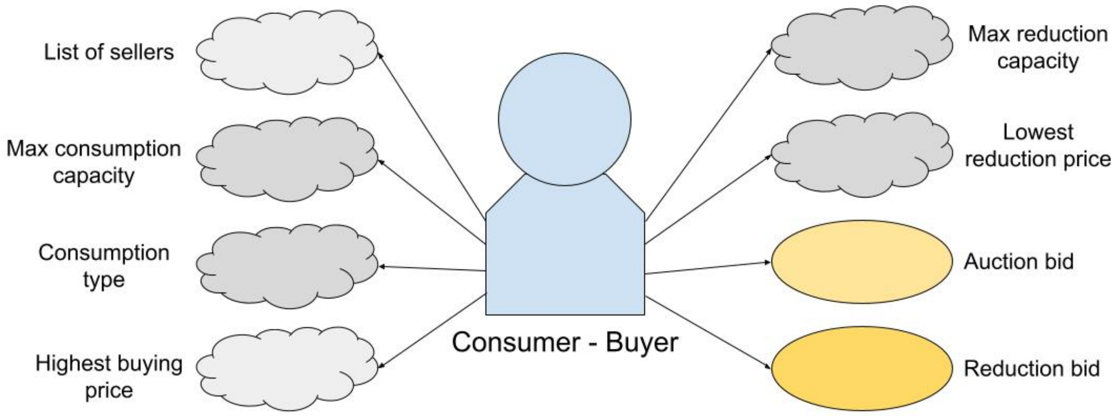

Figure 5.5: Diagram of the updated buyer role.

Furthermore, the class diagram had to be modified to include the classes' new information. Despite this, relations between classes have not changed neither new classes have been added. The modifications carried out are only related to class attributes from the TablaManager class and TakesPart relation. Therefore, figure 5.6 only will display the changes regarding to the initial uml class diagram in figure 4.9.

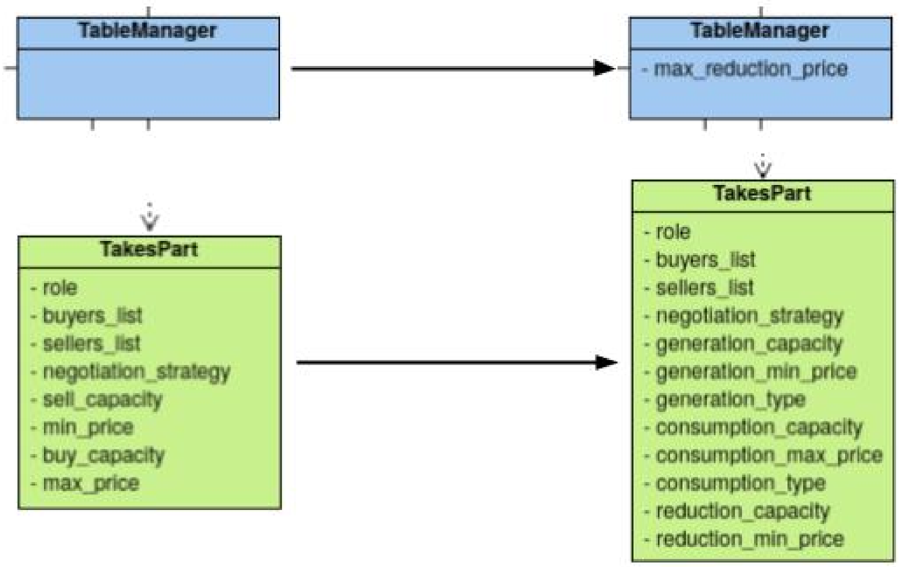

Figure 5.6: Diagram of the updated buyer role.

The table manager must include its new attribute *max_reduction_price* in order to know when to accept an energy reduction offer. After discussing this parameter's value, the decision of setting it at a constant value of 0.4 was made.

The takes part relation is the one that had the most modifications, as the energy to be bought and sold has additional characterization and now agents are able to make energy reduction offers. Therefore, the *sell_capacity* and *buy_capacity* were renamed to *generation_capacity* and *consumption_capacity*, respectively. Moreover, the rest of the attributes are represented by: *generation_type*, *consumption_type*, *reduction_capacity* and *reduction_min_price*.

In relation to the input data, it is only necessary to include new attributes for the correct characterization of the agents. The previous configuration remains the same. Moreover, the output data will need to specify the new type of transaction, the energy reductions, and the time when that kind of agreements are reached.

Due to the changes in the auction step, one interaction diagram must be modified to incorporate the new required behavior. Negotiators, particularly, buyers with reduction capabilities may now negotiate a reduction price with the table manager. This is the reason why the auction phase's diagram needs to be updated.

Additionally, the matching process will now need to take into account the normal bids and energy reduction bids. Although, the matching will include these new agreements, the matching conditions will remain the same.

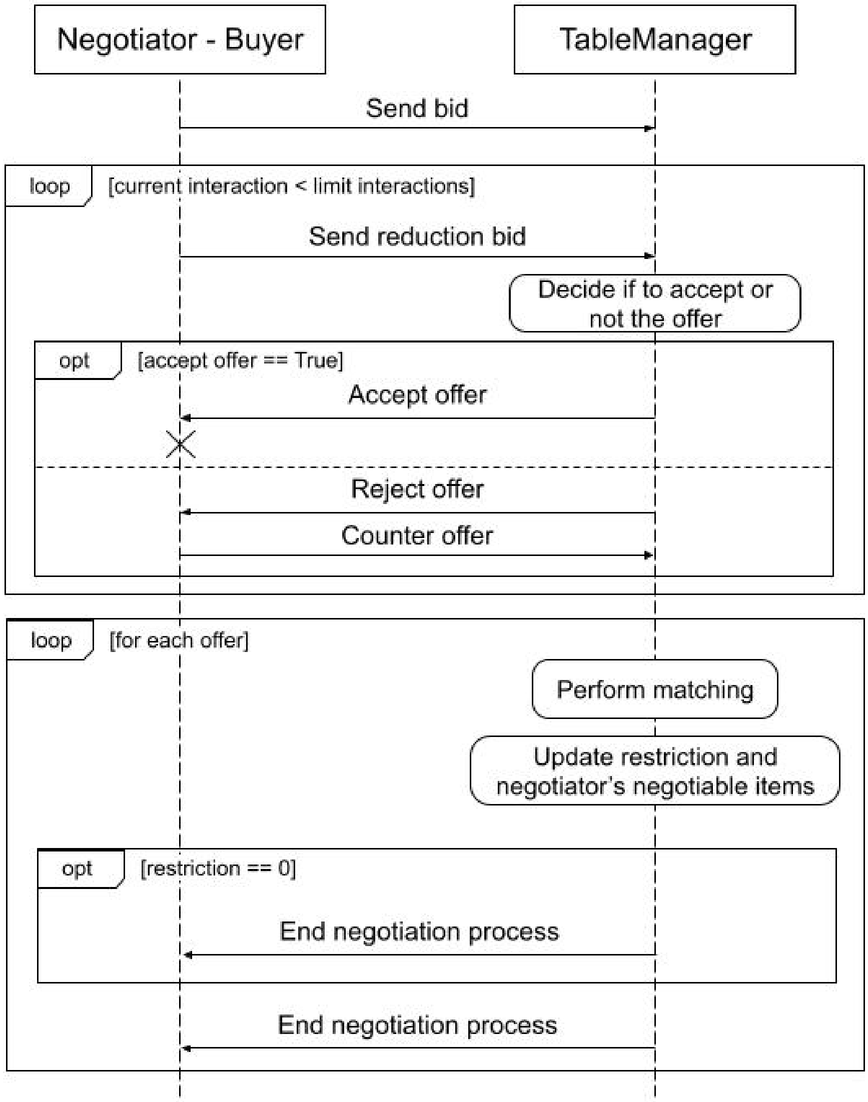

Figure 5.7: Interaction diagram displaying the updated auction phase.

5.2.4. Implementation

This section describes the additional implementation carried out for the local electricity market case study. In order to comply with the case study's market design the generic market simulator software has been forked. This fork includes the implementation specific to the local electricity market.

The first file that is modified is the *utils.py* file. The changes performed include adjusting the parse the negotiator's new attributes from the market specification XLS file to the market properties JSON file as shown in figure 4.25. Moreover, the negotiator's registration verification method was updated to comply with the rules specified in table 5.1.

Secondly, the *negotiator.py* and *tableManager.py* were modified. The changes made introduced new functionalities and attributes, so that they can be able negotiate reduction offers.

5.2.5. Validation

The core functionalities of the generic market simulator have been validated previously,this section will focus on the validation of the new implementations related to the local electricity market.

Validation tests

An experiment has been conducted in order to carry out the validation of the flexible local electricity market simulator's implementation. This experiment will cover all the new functionality to verify the software works as expected.

The simulated market is composed of 2 negotiation periods with a different restriction for each one, this will be enough to test the new registration conditions shown in table 5.1 and the consumer's capability to agree to reduce its energy consumption. Additionally, the maximum number of interactions during the bilateral negotiation and the reduction negotiation is set to 20. Moreover, the table manager's maximum price to pay for the reduction of energy is 20.

Table 5.2: Table of the restrictions in each period for the electricity market simulator validation.

Period	Restriction
1	-inf
2	inf

The validation experiment will composed of 2 generators, 2 consumers and 2 prosumers which will be taking part in the market. Moreover, 3 of these agents are able to reduce their energy consumption. Their individual energy capabilities and price limits are specified in table 5.3.

Table 5.3: Table of the constant parameters of the negotiator agents for the electricity market simulator validation.

	Agent 0	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Gen. capacity	10	5	0	0	10	5
Gen. type	normal	extra	-	-	normal	extra
Min. sell price	23	25	-	-	17	22
Con. capacity	0	0	10	5	10	5
Con. type	-	-	normal	extra	normal	extra
Max. buy price	-	-	20	27	30	22
Red. capacity	5	0	5	0	0	5
Min. red. price	15	-	19	-	-	22
Strategy	Linear	Linear	Linear	Linear	Linear	Linear

Results

The goal of having a deficit and surplus period is to analyze if the registration period is consistent with the market's design. Additionally, this allows to verify that reduction agreements only occur in deficit periods. Table 5.4 displays the agreements reached during the negotiation sessions.

The experiment's results show that energy reduction agreements were only allowed in deficit periods. The infinite restriction allows negotiations to continue until all negotiators have used up all their capacity or agreements are no longer economically viable. This ensures the auction step is reached and makes possible to reach reduction agreements. In the end, it can be affirmed the implementation of this functionality complies with its proposed design.

Furthermore, the type of energy generated or consumed and the agent's capabilities also has to be taken into account in the registration period. The negotiators taking part in each negotiation period agrees with the conditions stated in table 5.1. For example, agents 3 and 5 cannot act as consumers in period 1 and agents 1 and 5 cannot act as generators in period 2.

Given all agreements reached are technically and economically feasible, it can be concluded this test allows the validation of the proposed market's implementation. The next section will proceed with the case study's simulation.

Table 5.4: Table that shows the results from the flexible local electricity market validation test.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Quantity	Price
1	-inf	negotiation	Agent 0	Agent 3	5	22.661
		negotiation	Agent 5	Agent 4	5	28.393
		negotiation	Agent 0	Agent 4	5	28.393
		negotiation	Agent 4	Agent 2	5	18.929
		auction	Agent 1	Agent 2	5	15.0
		reduction	-	Agent 0	5	18.214
		reduction	-	Agent 2	5	19.0
2	inf	negotiation	Agent 0	Agent 3	5	22.661
		negotiation	Agent 4	Agent 3	5	24.107
		negotiation	Agent 0	Agent 4	10	27.929
		negotiation	Agent 4	Agent 2	5	18.929

5.3 Simulation

Once the system is validated, the next step is to expose it to a real life scenario in order to see its behavior in the wild. It is important to validate the software and verify all the implementation was done correctly. However, it is even more so to test an artificial intelligence system in a real environment.

With respect to the agent specification, it was carried out by the Institute of Energy Engineering. The full specification of the agents can be found in [61]. Since, this work's focus is not to calculate these values, it will only show examples of the agent specifications for a better understanding of what information was used to create the market's properties file.

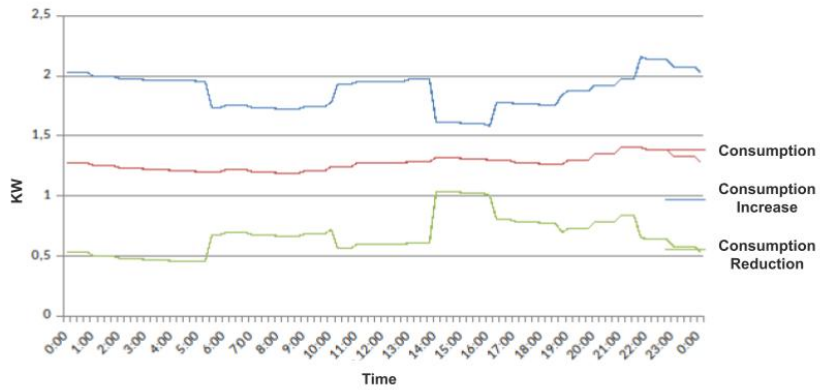

Figure 5.8: Example of an agent's energy consumption specification.

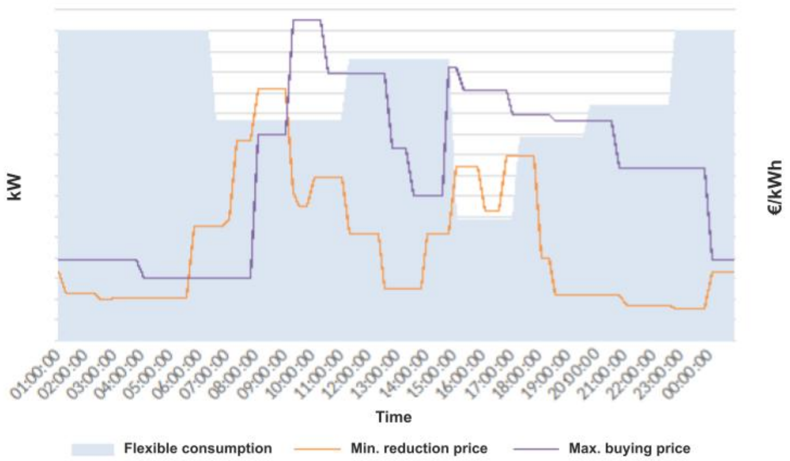

Figure 5.9: Example of the estimated values related to an agent's energy consumption.

Figure 5.8 shows an agent's possible consumption range during a day. Figure 5.9 shows the consumption flexibility during a day. Also, it is displayed the maximum price for which it will consume more energy and minimum price it should be paid to reduce its consumption.

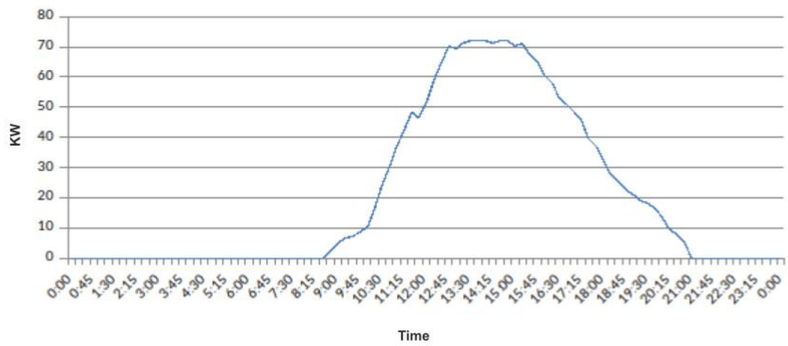

Figure 5.10: Example of an agent's energy generation specification.

Figure 5.11: Example of the estimated values related to an agent's energy generation.

Figure 5.10 and 5.11 both show the amount of power generated during a day. Figure 5.11 adds the minimum price at which it would sell its generated energy.

All this information was used to characterize the flexible local electricity market's negotiators.

5.3.1. Market configuration

Before carrying out the simulation of the market is important to prepare an experimental plan alongside with the corresponding properties files for each test.

The experiments' aim is to study the behavior of the market when exposed to different stimuli in the input data. Following this approach, a base case is defined to later carry out different changes on the input data to later analyze its response them.

There are several parameters common to all simulations. First, the restrictions. The intra-daily market is divided in 15 minute periods. In words words, a day is divided into 15 minute periods and each one of them may has a restriction, this is displayed in figure 5.5. Additionally, the negotiation's maximum number of interactions is set to 15. In total, 18 agents have been specified to take part in this simulation. Moreover, the price reasoning used by agents is the one implemented by default.

Table 5.5: Table of the the local electricity market's negotiation periods and their associated restriction.

Period	Restriction	Period	Restriction	Period	Restriction
00:00	0	08:00	0	16:00	0
00:15	0	08:15	0	16:15	10
00:30	0	08:30	-10	16:30	10
00:45	0	08:45	-10	16:45	10
01:00	0	09:00	-10	17:00	0
01:15	0	09:15	-10	17:15	0
01:30	0	09:30	0	17:30	0
01:45	0	09:45	0	17:45	0
02:00	0	10:00	0	18:00	0
02:15	0	10:15	0	18:15	0
02:30	0	10:30	0	18:30	0
02:45	0	10:45	0	18:45	0
03:00	0	11:00	0	19:00	0
03:15	0	11:15	0	19:15	-15
03:30	0	11:30	0	19:30	-5
03:45	0	11:45	0	19:45	-5
04:00	0	12:00	0	20:00	-5
04:15	0	12:15	0	20:15	0
04:30	0	12:30	0	20:30	0
04:45	0	12:45	0	20:45	0
05:00	0	13:00	30	21:00	0
05:15	0	13:15	30	21:15	0
05:30	0	13:30	30	21:30	0
05:45	0	13:45	30	21:45	0
06:00	0	14:00	0	22:00	0
06:15	0	14:15	0	22:15	0
06:30	0	1:30	0	22:30	0
06:45	0	14:45	0	22:45	0
07:00	-20	15:00	0	23:00	0
07:15	-20	15:15	0	23:15	0
07:30	-20	15:30	0	23:30	0
07:45	0	15:45	0	23:45	0

Table 5.6 displays the experimental plan and which parameters have been modified in each test:

Table 5.6: Table of the experimental plan for the local electricity market simulation.

Exp.	# of Res.	# of Agents	Res.'s value	Agent capabilities	Limit prices	Strategy
0	Base case	Base case	Base case	Base case	Base case	Base case
1	+5 periods	Base case	Base case	Base case	Base case	Base case
2	-5 periods	Base case	Base case	Base case	Base case	Base case
3	Base case	+2 agents	Base case	Base case	Base case	Base case
4	Base case	-2 agents	Base case	Base case	Base case	Base case
5	Base case	Base case	+10%	Base case	Base case	Base case
6	Base case	Base case	-10%	Base case	Base case	Base case
7	Base case	Base case	Base case	+15%	Base case	Base case
8	Base case	Base case	Base case	-15%	Base case	Base case
9	Base case	Base case	Base case	Base case	+5%	Base case
10	Base case	Base case	Base case	Base case	-5%	Base case
11	Base case	Base case	Base case	Base case	Base case	Linear
12	Base case	Base case	Base case	Base case	Base case	Conservative
13	Base case	Base case	Base case	Base case	Base case	Aggressive
14	Base case	Base case	Base case	Base case	Base case	Mixed 1
15	Base case	Base case	Base case	Base case	Base case	Mixed 2
16	Base case	Base case	Base case	Base case	Forced auction	Base case

The negotiation strategy has been modified for experiment 11, 12, 13, 14 and 15. Table 5.7 shows the strategy used by each agent for each experiment:

Table 5.7: Table of the negotiation strategy used by each negotiator over the different experiments.

Agent	Base case	Exp. 11	Exp. 12	Exp. 13	Exp. 14	Exp. 15
0	Linear	Linear	Conservative	Aggressive	Conservative	Aggressive
1	Conservative	Linear	Conservative	Aggressive	Linear	Linear
2	Aggressive	Linear	Conservative	Aggressive	Conservative	Aggressive
3	Aggressive	Linear	Conservative	Aggressive	Conservative	Aggressive
4	Conservative	Linear	Conservative	Aggressive	Conservative	Aggressive
5	Linear	Linear	Conservative	Aggressive	Linear	Linear
6	Linear	Linear	Conservative	Aggressive	Conservative	Aggressive
7	Aggressive	Linear	Conservative	Aggressive	Linear	Linear
8	Linear	Linear	Conservative	Aggressive	Conservative	Aggressive
9	Conservative	Linear	Conservative	Aggressive	Linear	Linear
10	Aggressive	Linear	Conservative	Aggressive	Linear	Linear
11	Linear	Linear	Conservative	Aggressive	Conservative	Aggressive
12	Conservative	Linear	Conservative	Aggressive	Linear	Linear
13	Linear	Linear	Conservative	Aggressive	Conservative	Aggressive
14	Aggressive	Linear	Conservative	Aggressive	Linear	Linear
15	Aggressive	Linear	Conservative	Aggressive	Conservative	Aggressive
16	Conservative	Linear	Conservative	Aggressive	Linear	Linear
17	Linear	Linear	Conservative	Aggressive	Conservative	Aggressive

5.3.2. Results

The visualization proved to be a very convenient tool when analyzing the experiment's results. The following images show a negotiation and some results, both from the base case experiment.

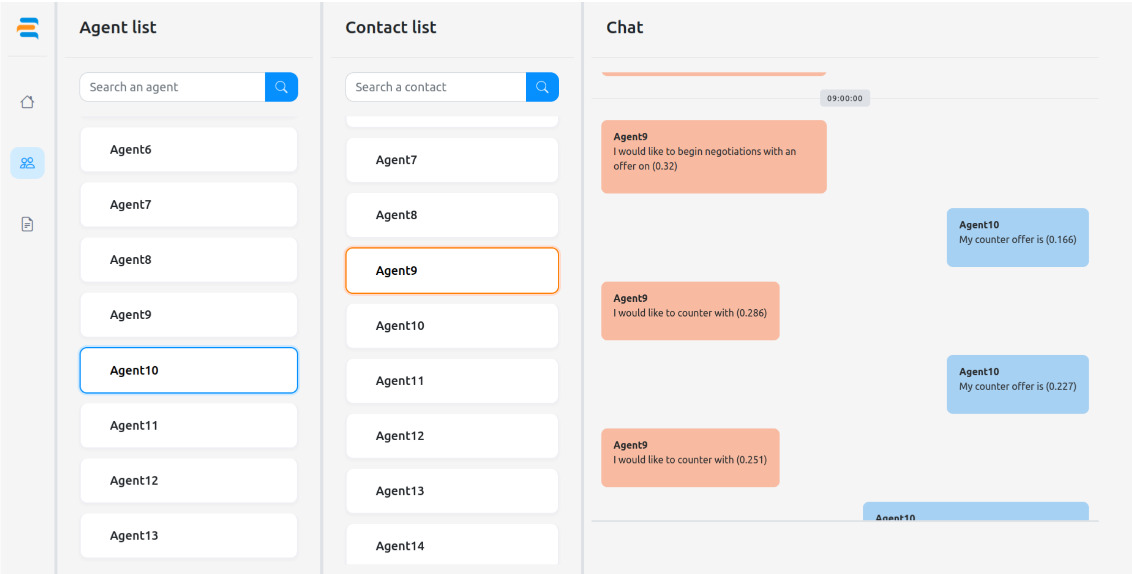

Figure 5.12: Image showing a negotiation from the base case experiment.

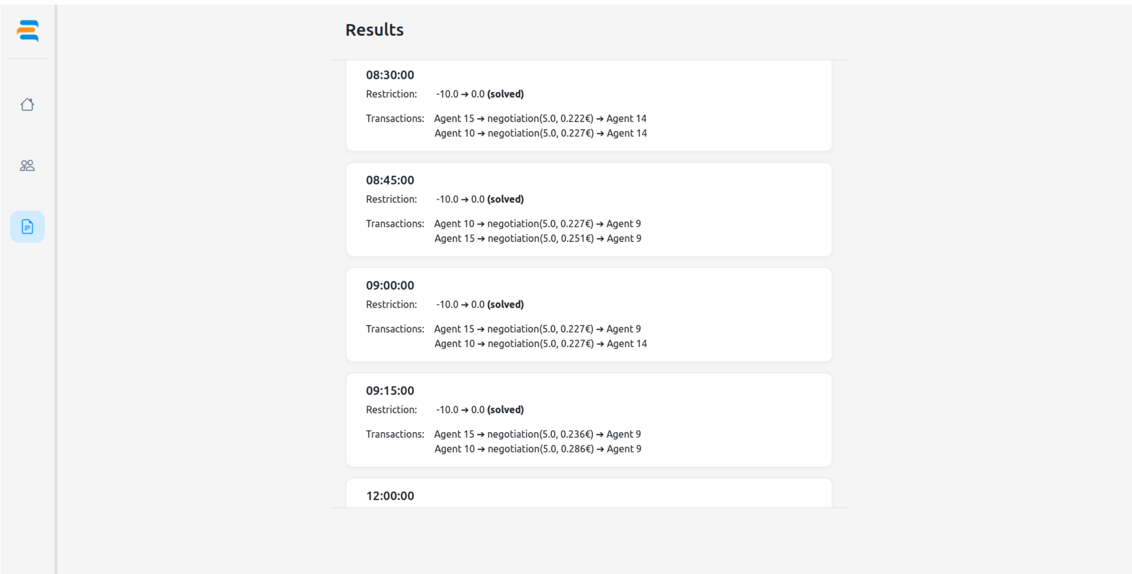

Figure 5.13: Image showing some results of the base case experiment.

Table A.3 shows the results obtained in the negotiation sessions as well as the details of the agreements reached. This table only shows the results of deficit or surplus restrictions since no negotiation process is needed to carry out for zero restriction periods.

Analyzing the internal behavior of the simulator with the visualization tools, it is possible to confirm that the implementation works accordingly to the previously designed market.

Table 5.8: Table of the results from the base case experiment, experiment 0.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	10	0.201
7:15	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
7:30	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 14	5	0.227
8:45	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 10	Agent 9	5	0.251
		negotiation	Agent 15	Agent 14	5	0.227
9:15	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.286
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 6	30	0.093
13:30	30	negotiation	Agent 16	Agent 2	15	0.093
		negotiation	Agent 16	Agent 9	15	0.1
13:45	30	negotiation	Agent 16	Agent 6	30	0.093
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 8	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	10	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 15	Agent 9	5	0.16
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Results analysis

The simulator's main conclusions are drawn out from the comparative analysis of the results obtained from the different experiments. Based on the results obtained in the prepared scenarios, the main differences identified between the proposed cases are presented in the following table:

Table 5.9: Table of the results obtained from the comparative analysis from the different experiments.

Experiment	Restrictions solved (%)	Agreements reached		Avg. price (€/kWh)	Price* (%)
		Bilateral negotiation	Auction		
0	100	32	0	0.165	32.26
1	100	36	0	0.162	34.39
2	100	24	0	0.161	34.73
3	100	31	0	0.170	35.30
4	100	31	0	0.167	34.48
5	100	36	0	0.177	36.32
6	100	32	0	0.168	35.04
7	100	31	0	0.169	35.87
8	100	38	0	0.177	35.16
9	100	31	0	0.176	34.76
10	100	32	0	0.154	35.12
11	100	30	0	0.157	29.87
12	100	32	0	0.156	30.18
13	100	35	0	0.151	31.36
14	100	30	0	0.158	32.97
15	100	31	0	0.165	40.40
16	100	0	30	0.066	151.71

* Percentage position of the final average price. The percentage scale ranges from the minimum sale price (0%) to the maximum purchase price (100%).

The insights obtained from the local electricity market's experiments results are described below:

1. As shown in table 5.9, the simulator responds to the changes done to the input data. These stimuli affects the number of agreements reached, the agents involved in each agreement as well as the amount of energy negotiated and the agreed price.

2. If it is technically viable (the quantity negotiated) and economically viable (the limit prices), all agreements will be performed during the bilateral negotiation. Table 5.9 shows how in the first 15 experiments, all agreements are closed during the first negotiation phase. In order to analyze the behavior of agents in the auction phase, the 16th experiment forced the auction process. This was done by increasing the minimum sale prices above maximum purchase prices. This makes impossible the economic viability of agreements in the first phase, therefore, all agreements occur in the auction phase.

3. All non zero restriction negotiation periods are technically and economically viable and so, as a result, all restriction are solved. Table A.3 shows the results of the base case. If within each negotiation period all the negotiated power is added up, it will sum up to the restriction's value. This happens in all other negotiation sessions from all the experiments.

4. The number of agreements reached in the different experiments changes in a coherent manner with the input data's stimuli:

   - Increase in the number of agreements reached - this occurs when the restriction's value is higher than the agent's capacities or when the agent's energy capacities are lower than the restriction's value. This leads to more agreements of smaller amounts of power. This can be observed from the comparison of results of the base case and experiments 5 through 8.

- Reduction in the number of agreements reached - this is due to the decrease in value of the restriction itself or an increase in the agents' energy capacity, leading to less agreements of greater amounts of power.

5. The settled price in agreements is not affected by the amount of power negotiated. Only the price of an agreement in bargained without taking into consideration the amount of power negotiated. This means that if an agreement is reached it only depends on the economic viability of agents. A clear example of this event occurs in periods 12:15 and 13:00 from the base case. Where for significantly different amounts of power, the agreed price is the same:

Table 5.10: Table of the results from the base case experiment for periods 12:15 and 13:00.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1

6. The agreed prices are closer to the minimum sale price than to the maximum purchase price. Within the possible range of prices, ranging from the lowest sale price (0%) to the highest buying price (100%), and without taking into account experiment 16 (as it can be considered an outlier) the average agreed price in negotiations is 34.263% of the maximum possible value.

7. Experiment 16 shows that in the auction phase, the average agreed price is significantly lower than in bilateral negotiations. This value is lower than the minimum sale price. As shown in table A.3 the percentage relative to the average price would be outside the range between the minimum sale price and the maximum purchase price, with a result of 151.71%. Although, this price is out of the expected range, this is consistent with how negotiators reason their prices.

8. Experiments where all negotiators share the same negotiation strategy lead to agreed prices closer to the minimum selling price. This is visible comparing experiments 0 trough 10 and the 11th, 12th and 13th experiments. Where the average agreed price of these sets of experiments are are 34.442% and 31.503% respectively of the maximum buying price.

9. Changing the economic characterization of agents entails consistent changes in the results obtained from the simulation. If the limit prices are increased, the average agreed prices increases as well. Meanwhile, the reduction of the limit prices reduces the average agreed price. Experiments 9 increases the limit prices by 5% and experiment 10 decreases the limit prices by 5%. This leads to an increase and decrease, respectively, in 6.7% the average agreed price. As the change in price is symmetrical and in equal proportion in both ways, this verifies the coherence in the results obtained. Additionally, it shows how relevant the economic characterization is in this version of the simulator.

10. Some agents have more presence than others in the agreements reached. It has been noticed that negotiators with greater capabilities (energy generation, consumption or reduction) as well as having less restrictive price limits reach more settlements than other agents. An example of this are agents 9 and 16.

After the analysis of the results given by the first version of the simulator, it can be confirmed that it has provided enough knowledge to validate the local electricity market proposal. In order to gain deeper insights about the local electricity market's proposed the design of the generic virtual market simulator should be further evaluated. The following section will proceed to develop the conclusions obtained from the work carried out.

CHAPTER 6
Conclusions

This work has provided a solution to the application area of the generic virtual market simulation. The tools used in the work as well as the technologies have been described. Both developed markets, the generic virtual market and the flexible local electricity market have been developed according to the standard software development process and both platforms have been validated to make sure it complies with their proposal. The result of this work is a fully functional simulator for generic virtual markets alongside its visualization tool. Furthermore, it has been possible to use this software to study the design of a flexible local electricity market.

This chapter presents the achieved objectives and the conclusions obtained from the work carried out within this project. In first place, the general conclusions about the generic virtual market will be presented. These conclusions are based on the development carried out and the validation tests performed to confirm all the implementation is consistent with the work's proposal.

Additionally, the simulator has been adapted and tested in a real life environment. The specific conclusions of the flexible local electricity market will also be presented. These are obtained by the results analysis of the electricity market simulation. These conclusions will help to better understand the current capabilities of the simulator and define future improvements for the simulator.

Once the conclusions have been described, future lines of work will be presented to continue and improve this ambitious project.

6.1 Generic market simulator

The current thesis has detailed the complete software development process to design, develop, implement and validate the generic market simulator based on multi-agent technology and organizational concepts as well as its visualization tool. Therefore, it is possible to say that this work successfully achieved the initially set objectives.

The following conclusions were obtained from this project:

- This work has studied the necessary concepts to correctly implement the proposed market. These includes organizational structures, multi-agent technologies and the SPADE platform.
- The requirements to create a generic virtual market have been studied and analyzed to provide a novel solution in this application area.

- This work has carried out a standard software development process to develop the simulator and its visualization tool.

- All the software's functionality has been validated. In other words, all the implementation works as expected and it is consistent with the proposal made.

6.2 Case study application

The simulation of real life scenarios is a complex task. This final master's project has not been able to offer a final solution to analyze the full behavior of the flexible local electricity market. Nevertheless, the software has been accepted as the first milestone to be improved in the future and has been able to validate the work and proposal made in [61].

Given the results obtained from the simulation and their analysis, the following results are obtained:

- The context in which the simulator needs to be applied has been studied. This has been done to correctly understand the flexible local electricity market proposal and create a coherent adaptation of the generic market simulator. Subsequently, an exclusive software development process has been carried out to comply with the Institute of Energy Engineering's proposal.

- The simulator provides a coherent solution with the specification of the flexible local electricity market. It behaves as expected in the different experiments. However, the agent's behavior has been considered basic as it does not show enough intelligence to resemble what real interactions would look like between actors.

- Negotiations just focus on their economic viability. Negotiator's negotiation strategy only takes into account their economic characterization. The results analysis displays evidence of this. Therefore, negotiations strategies need further characterization to take into account other factors such as the quantity negotiated. This would make the agreed price be more coherent with the products exchanged.

- Negotiator's do not have mechanisms to analyze if an agreement is more profitable than others. If negotiators only try to maximize their profits they would only try to sell energy or reduce their consumption, as these are the only ways to gain profit. Instead, negotiators try to reach agreements at an economically acceptable price, as long as they have power to continue negotiations. This has shown some unexpected behavior. For example, most negotiations occur in the bilateral negotiation, as these are economically feasible and only experiment 16 reaches the auction phase. Additionally, all restrictions are solved in all experiments. This is not necessarily a realistic behavior. Therefore, it is necessary to characterize how negotiators benefit from agreements as buyers and a sellers.

- Given the results obtained from current behavior of the market, there is not enough information to compare the competition between the bilateral negotiation and the auction phase.

The current definition of the flexible local electricity market does not provided enough information about how negotiators should evaluate their negotiated agreements. This has proven to be a key point to further explore the behavior of agents in a given market.

Ultimately, a first simulator version of the market has been developed. Despite not having very intelligent behavior, it is a solid base from which future versions can be develop, incorporating more complex behaviors that emulate more realistic negotiations. In the end, this will enable researchers to obtain deeper insights from market simulations.

6.3 Future Work

This master's thesis provides the starting point for further research in the application of generic virtual market simulations. There are several lines of work which could improve this project, the following points define contributions to develop further the current simulation tool:

- Extend the visualization tool to a full web graphical user interface. This would enable to characterize markets, create and define negotiator agents, load and save simulation properties files, load simulation results and visualize log files if needed.
- Include more characterization possibilities to the market. Such as, opening a market with only a bilateral negotiation or auction and enable the specification of other types of auction.
- Introduce more options to the current negotiation strategies. Create default options, similar to the current ones, and a fully customizable concession curve. This would combine the current step and concession curves and, additionally, enabling agents to maintain a constant price at their initial price and/or their limit price.
- Make negotiation strategies multi-attribute so these can take into account additional information other than the agent's economic characterization. This information includes quantity, the agent's needs and the benefit of an agreement.
- Implement a function to calculate a successful agreement's intrinsic value a apart from its economic profit.
- Modify the agent's behavior so it can evaluate which negotiation phase is more beneficial to take part in.
- Once, the behavior of agents has been adjusted reevaluate the simulator in the same and new case studies.

This thesis is VRAIN's first milestone of research on generic markets simulators for virtual markets. The next step is to continue working on the mentioned points and continue to improve the current solution.

APPENDIX A

Case study's experimental results

This appendix will display all the experimental results from the case study's simulation. This data was used to create table 5.9, the results comparative analysis table.

Table A.1: Table of the results from the base case experiment, experiment 0.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	10	0.201
7:15	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
7:30	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 14	5	0.227
8:45	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 10	Agent 9	5	0.251
		negotiation	Agent 15	Agent 14	5	0.227
9:15	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.286
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 6	30	0.093
13:30	30	negotiation	Agent 16	Agent 2	15	0.093
		negotiation	Agent 16	Agent 9	15	0.1
13:45	30	negotiation	Agent 16	Agent 6	30	0.093
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 8	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	10	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 15	Agent 9	5	0.16
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.2: Table of the results from experiment 1.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
6:45	-10	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
7:00	-20	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	10	0.201
7:15	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	15	0.201
7:30	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 14	5	0.227
8:45	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:15	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.286
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 8	30	0.1
13:15	30	negotiation	Agent 16	Agent 6	30	0.093
13:30	30	negotiation	Agent 16	Agent 2	15	0.093
		negotiation	Agent 16	Agent 9	15	0.1
13:45	30	negotiation	Agent 16	Agent 6	30	0.093
14:00	50	negotiation	Agent 16	Agent 9	50	0.1
16:00	20	negotiation	Agent 16	Agent 8	20	0.112
16:15	10	negotiation	Agent 16	Agent 9	10	0.112
16:30	10	negotiation	Agent 16	Agent 8	10	0.112
16:45	10	negotiation	Agent 16	Agent 8	10	0.112
19:00	10	negotiation	Agent 16	Agent 9	10	0.112
19:15	-15	negotiation	Agent 16	Agent 14	15	0.112
19:30	-5	negotiation	Agent 15	Agent 9	5	0.16
19:45	-5	negotiation	Agent 15	Agent 9	5	0.16
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.3: Table of the results from experiment 2.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	10	0.201
7:15	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 14	5	0.227
8:45	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 6	30	0.093
13:30	30	negotiation	Agent 16	Agent 2	15	0.093
		negotiation	Agent 16	Agent 9	15	0.1
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	10	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 16	Agent 15	5	0.112
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.4: Table of the results from experiment 3.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	10	0.201
7:15	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
7:30	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 9	5	0.251
8:45	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:15	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.286
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 2	Agent 9	5	0.196
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 9	30	0.1
13:30	30	negotiation	Agent 16	Agent 6	30	0.093
13:45	30	negotiation	Agent 16	Agent 9	30	0.1
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 8	10	0.112
16:45	10	negotiation	Agent 16	Agent 8	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	10	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 16	Agent 14	5	0.112
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.5: Table of the results from experiment 4.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	10	0.201
7:15	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	10	0.201
7:30	-20	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.16
		negotiation	Agent 13	Agent 14	10	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 14	5	0.227
8:45	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 9	5	0.227
9:15	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.286
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 6	30	0.093
13:30	30	negotiation	Agent 16	Agent 9	30	0.1
13:45	30	negotiation	Agent 16	Agent 9	30	0.1
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 8	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	10	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 15	Agent 9	5	0.16
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.6: Table of the results from experiment 5.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-22	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	12	0.201
7:15	-22	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	12	0.201
7:30	-22	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	12	0.201
8:30	-11	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 14	5	0.227
		negotiation	Agent 13	Agent 9	1	0.261
8:45	-11	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
		negotiation	Agent 13	Agent 9	1	0.274
9:00	-11	negotiation	Agent 15	Agent 14	5	0.227
		negotiation	Agent 10	Agent 9	5	0.251
		negotiation	Agent 9	Agent 14	1	0.25
9:15	-11	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 13	Agent 9	6	0.334
12:00	5	negotiation	Agent 16	Agent 8	5.5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5.5	0.1
13:00	33	negotiation	Agent 16	Agent 9	33	0.1
13:15	33	negotiation	Agent 16	Agent 6	33	0.093
13:30	33	negotiation	Agent 16	Agent 9	33	0.1
13:45	33	negotiation	Agent 16	Agent 9	33	0.1
16:15	11	negotiation	Agent 16	Agent 8	11	0.112
16:30	11	negotiation	Agent 16	Agent 9	11	0.112
16:45	11	negotiation	Agent 16	Agent 8	11	0.112
19:15	-16.5	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	11.5	0.112
19:30	-5.5	negotiation	Agent 16	Agent 15	5.5	0.112
19:45	-5.5	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 15	0.5	0.112
20:00	-5.5	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 2	Agent 9	0.5	0.213

Table A.7: Table of the results from experiment 6.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-18	negotiation	Agent 15	Agent 14	5	0.165
		negotiation	Agent 10	Agent 14	5	0.179
		negotiation	Agent 13	Agent 14	8	0.201
7:15	-18	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	8	0.201
7:30	-18	negotiation	Agent 15	Agent 14	5	0.166
		negotiation	Agent 10	Agent 14	5	0.185
		negotiation	Agent 13	Agent 14	8	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5	0.222
		negotiation	Agent 10	Agent 14	4	0.227
8:45	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 10	Agent 9	4	0.251
9:00	-10	negotiation	Agent 15	Agent 9	5	0.227
		negotiation	Agent 13	Agent 9	4	0.274
9:15	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 13	Agent 9	4	0.334
12:00	4.5	negotiation	Agent 16	Agent 8	4.5	0.1
12:15	4.5	negotiation	Agent 16	Agent 9	4.5	0.1
13:00	27	negotiation	Agent 16	Agent 9	27	0.1
13:15	27	negotiation	Agent 16	Agent 9	27	0.1
13:30	27	negotiation	Agent 16	Agent 2	15	0.093
		negotiation	Agent 16	Agent 9	12	0.1
13:45	27	negotiation	Agent 16	Agent 9	27	0.1
16:15	9	negotiation	Agent 16	Agent 8	9	0.112
16:30	9	negotiation	Agent 16	Agent 9	9	0.112
16:45	9	negotiation	Agent 16	Agent 8	9	0.112
19:15	-13.5	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	8.5	0.112
19:30	-4.5	negotiation	Agent 16	Agent 15	4.5	0.112
19:45	-4.5	negotiation	Agent 15	Agent 9	4.5	0.16
20:00	-4.5	negotiation	Agent 15	Agent 9	4.5	0.16

Table A.8: Table of the results from experiment 7.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5.75	0.165
		negotiation	Agent 10	Agent 14	5.75	0.179
		negotiation	Agent 13	Agent 14	8.5	0.201
7:15	-20	negotiation	Agent 15	Agent 14	5.75	0.166
		negotiation	Agent 10	Agent 14	5.75	0.185
		negotiation	Agent 13	Agent 14	8.5	0.201
7:30	-20	negotiation	Agent 15	Agent 14	5.75	0.166
		negotiation	Agent 10	Agent 14	5.75	0.185
		negotiation	Agent 13	Agent 14	8.5	0.201
8:30	-10	negotiation	Agent 15	Agent 14	5.75	0.222
		negotiation	Agent 10	Agent 14	4.25	0.227
8:45	-10	negotiation	Agent 15	Agent 9	5.75	0.227
		negotiation	Agent 10	Agent 9	4.25	0.251
9:00	-10	negotiation	Agent 10	Agent 9	5.75	0.251
		negotiation	Agent 15	Agent 14	4.25	0.227
9:15	-10	negotiation	Agent 15	Agent 9	5.75	0.236
		negotiation	Agent 13	Agent 9	4.25	0.286
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 6	30	0.093
13:30	30	negotiation	Agent 16	Agent 9	30	0.1
13:45	30	negotiation	Agent 16	Agent 6	30	0.093
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 8	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	5.75	0.16
		negotiation	Agent 16	Agent 9	9.25	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 15	Agent 9	5	0.16
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.9: Table of the results from experiment 8.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	4.25	0.165
		negotiation	Agent 10	Agent 14	4.25	0.179
		negotiation	Agent 13	Agent 14	11.5	0.201
7:15	-20	negotiation	Agent 15	Agent 14	4.25	0.166
		negotiation	Agent 10	Agent 14	4.25	0.185
		negotiation	Agent 13	Agent 14	11.5	0.201
7:30	-20	negotiation	Agent 15	Agent 14	4.25	0.166
		negotiation	Agent 10	Agent 14	4.25	0.185
		negotiation	Agent 13	Agent 14	11.5	0.201
8:30	-10	negotiation	Agent 15	Agent 14	4.25	0.222
		negotiation	Agent 10	Agent 14	4.25	0.227
		negotiation	Agent 13	Agent 9	1.5	0.261
8:45	-10	negotiation	Agent 15	Agent 9	4.25	0.227
		negotiation	Agent 10	Agent 9	4.25	0.251
		negotiation	Agent 13	Agent 9	1.5	0.274
9:00	-10	negotiation	Agent 15	Agent 9	4.25	0.227
		negotiation	Agent 10	Agent 9	4.25	0.251
		negotiation	Agent 13	Agent 14	1.5	0.241
9:15	-10	negotiation	Agent 15	Agent 9	4.25	0.236
		negotiation	Agent 10	Agent 9	4.25	0.286
		negotiation	Agent 13	Agent 9	1.5	0.334
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 9	30	0.1
13:30	30	negotiation	Agent 16	Agent 2	12.75	0.093
		negotiation	Agent 16	Agent 9	17.25	0.1
13:45	30	negotiation	Agent 16	Agent 9	30	0.1
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 9	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	4.25	0.16
		negotiation	Agent 16	Agent 9	10.75	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-15	negotiation	Agent 15	Agent 9	4.25	0.16
		negotiation	Agent 16	Agent 9	0.75	0.112
20:00	-15	negotiation	Agent 15	Agent 9	4.25	0.16
		negotiation	Agent 16	Agent 9	0.75	0.213

Table A.10: Table of the results from experiment 9.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.158
		negotiation	Agent 10	Agent 14	5	0.189
		negotiation	Agent 13	Agent 14	10	0.213
7:15	-20	negotiation	Agent 10	Agent 14	5	0.189
		negotiation	Agent 15	Agent 14	5	0.188
		negotiation	Agent 13	Agent 14	10	0.213
7:30	-20	negotiation	Agent 10	Agent 14	5	0.189
		negotiation	Agent 15	Agent 14	5	0.188
		negotiation	Agent 13	Agent 14	10	0.213
8:30	-10	negotiation	Agent 15	Agent 14	5	0.225
		negotiation	Agent 10	Agent 14	5	0.238
8:45	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.251
9:15	-10	negotiation	Agent 15	Agent 9	5	0.24
		negotiation	Agent 10	Agent 9	5	0.286
12:00	5	negotiation	Agent 16	Agent 8	5	0.12
12:15	5	negotiation	Agent 16	Agent 9	5	0.12
13:00	30	negotiation	Agent 16	Agent 9	30	0.12
13:15	30	negotiation	Agent 16	Agent 9	30	0.12
13:30	30	negotiation	Agent 16	Agent 9	30	0.12
13:45	30	negotiation	Agent 16	Agent 9	30	0.12
16:15	10	negotiation	Agent 16	Agent 8	10	0.12
16:30	10	negotiation	Agent 16	Agent 9	10	0.12
16:45	10	negotiation	Agent 16	Agent 8	10	0.12
19:15	-15	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	10	0.12
19:30	-5	negotiation	Agent 16	Agent 15	5	0.12
19:45	-5	negotiation	Agent 15	Agent 9	5	0.12
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.11: Table of the results from experiment 10.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.156
		negotiation	Agent 10	Agent 14	5	0.157
		negotiation	Agent 13	Agent 14	10	0.188
7:15	-20	negotiation	Agent 15	Agent 14	5	0.156
		negotiation	Agent 13	Agent 14	15	0.188
		negotiation	Agent 15	Agent 14	5	0.156
7:30	-20	negotiation	Agent 13	Agent 14	15	0.188
8:30	-10	negotiation	Agent 15	Agent 14	5	0.213
		negotiation	Agent 10	Agent 9	5	0.242
8:45	-10	negotiation	Agent 15	Agent 9	5	0.225
		negotiation	Agent 10	Agent 9	5	0.251
9:00	-10	negotiation	Agent 10	Agent 9	5	0.251
		negotiation	Agent 15	Agent 9	5	0.225
9:15	-10	negotiation	Agent 15	Agent 9	5	0.24
		negotiation	Agent 13	Agent 9	5	0.318
12:00	5	negotiation	Agent 16	Agent 2	5	0.08
12:15	5	negotiation	Agent 16	Agent 9	5	0.08
13:00	30	negotiation	Agent 16	Agent 2	15	0.08
		negotiation	Agent 16	Agent 5	10	0.08
		negotiation	Agent 16	Agent 6	5	0.08
13:15	30	negotiation	Agent 16	Agent 6	30	0.08
13:30	30	negotiation	Agent 16	Agent 2	15	0.08
		negotiation	Agent 16	Agent 3	15	0.072
13:45	30	negotiation	Agent 16	Agent 9	30	0.08
16:15	10	negotiation	Agent 16	Agent 9	10	0.112
16:30	10	negotiation	Agent 16	Agent 8	10	0.112
16:45	10	negotiation	Agent 16	Agent 9	10	0.112
19:15	-15	negotiation	Agent 15	Agent 9	5	0.16
		negotiation	Agent 16	Agent 9	10	0.112
19:30	-5	negotiation	Agent 16	Agent 14	5	0.112
19:45	-5	negotiation	Agent 15	Agent 9	5	0.16
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.12: Table of the results from experiment 11.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.132
		negotiation	Agent 10	Agent 15	5	0.179
		negotiation	Agent 13	Agent 14	10	0.211
7:15	-20	negotiation	Agent 10	Agent 15	5	0.185
		negotiation	Agent 15	Agent 14	5	0.135
		negotiation	Agent 13	Agent 14	10	0.212
7:30	-20	negotiation	Agent 10	Agent 15	5	0.185
		negotiation	Agent 15	Agent 14	5	0.135
		negotiation	Agent 13	Agent 14	10	0.2
8:30	-10	negotiation	Agent 15	Agent 14	5	0.185
		negotiation	Agent 10	Agent 14	5	0.239
8:45	-10	negotiation	Agent 15	Agent 9	5	0.185
		negotiation	Agent 10	Agent 15	5	0.239
9:00	-10	negotiation	Agent 15	Agent 8	5	0.185
		negotiation	Agent 10	Agent 9	5	0.226
9:15	-10	negotiation	Agent 15	Agent 9	5	0.211
		negotiation	Agent 10	Agent 9	5	0.279
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 8	30	0.1
13:15	30	negotiation	Agent 16	Agent 9	30	0.1
13:30	30	negotiation	Agent 16	Agent 9	30	0.093
13:45	30	negotiation	Agent 16	Agent 9	30	0.093
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 9	10	0.112
19:15	-15	negotiation	Agent 16	Agent 8	15	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 16	Agent 8	5	0.112
20:00	-5	negotiation	Agent 15	Agent 9	5	0.143

Table A.13: Table of the results from experiment 12.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.14
		negotiation	Agent 10	Agent 15	5	0.178
		negotiation	Agent 13	Agent 15	10	0.227
7:15	-20	negotiation	Agent 10	Agent 15	5	0.178
		negotiation	Agent 15	Agent 14	5	0.14
		negotiation	Agent 7	Agent 15	10	0.237
7:30	-20	negotiation	Agent 15	Agent 14	5	0.14
		negotiation	Agent 10	Agent 15	5	0.178
		negotiation	Agent 13	Agent 14	10	0.193
8:30	-10	negotiation	Agent 15	Agent 14	5	0.178
		negotiation	Agent 10	Agent 15	5	0.242
8:45	-10	negotiation	Agent 15	Agent 9	5	0.191
		negotiation	Agent 10	Agent 15	5	0.242
9:00	-10	negotiation	Agent 15	Agent 7	5	0.178
		negotiation	Agent 10	Agent 9	5	0.242
9:15	-10	negotiation	Agent 15	Agent 9	5	0.222
		negotiation	Agent 10	Agent 9	5	0.276
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 9	30	0.1
13:30	30	negotiation	Agent 16	Agent 9	30	0.094
13:45	30	negotiation	Agent 15	Agent 5	10	0.094
		negotiation	Agent 16	Agent 4	10	0.094
		negotiation	Agent 16	Agent 9	10	0.094
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 9	10	0.112
19:15	-15	negotiation	Agent 16	Agent 9	15	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 16	Agent 9	5	0.112
20:00	-5	negotiation	Agent 15	Agent 9	5	0.15

Table A.14: Table of the results from experiment 13.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.134
		negotiation	Agent 10	Agent 14	5	0.181
		negotiation	Agent 13	Agent 14	10	0.212
7:15	-20	negotiation	Agent 10	Agent 15	5	0.188
		negotiation	Agent 15	Agent 14	5	0.137
		negotiation	Agent 13	Agent 15	10	0.212
7:30	-20	negotiation	Agent 15	Agent 14	5	0.137
		negotiation	Agent 10	Agent 15	5	0.188
		negotiation	Agent 13	Agent 15	10	0.212
8:30	-10	negotiation	Agent 15	Agent 14	5	0.188
		negotiation	Agent 10	Agent 15	5	0.24
8:45	-10	negotiation	Agent 15	Agent 9	5	0.188
		negotiation	Agent 10	Agent 9	5	0.227
9:00	-10	negotiation	Agent 15	Agent 14	5	0.188
		negotiation	Agent 10	Agent 15	5	0.24
9:15	-10	negotiation	Agent 15	Agent 9	5	0.236
		negotiation	Agent 10	Agent 9	5	0.257
12:00	5	negotiation	Agent 16	Agent 4	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 4	10	0.1
		negotiation	Agent 16	Agent 5	10	0.1
		negotiation	Agent 16	Agent 8	10	0.1
13:15	30	negotiation	Agent 16	Agent 5	15	0.1
		negotiation	Agent 16	Agent 4	15	0.1
		negotiation	Agent 16	Agent 8	15	0.1
13:30	30	negotiation	Agent 16	Agent 2	15	0.093
		negotiation	Agent 16	Agent 6	15	0.093
13:45	30	negotiation	Agent 16	Agent 9	30	0.1
16:15	10	negotiation	Agent 16	Agent 5	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 8	10	0.112
19:15	-15	negotiation	Agent 16	Agent 7	15	0.112
19:30	-5	negotiation	Agent 16	Agent 15	5	0.112
19:45	-5	negotiation	Agent 16	Agent 10	5	0.112
20:00	-5	negotiation	Agent 15	Agent 9	5	0.16

Table A.15: Table of the results from experiment 14.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 15	Agent 14	5	0.158
		negotiation	Agent 10	Agent 14	5	0.166
		negotiation	Agent 13	Agent 15	10	0.178
7:15	-20	negotiation	Agent 15	Agent 14	5	0.161
		negotiation	Agent 10	Agent 14	5	0.166
		negotiation	Agent 7	Agent 14	10	0.219
7:30	-20	negotiation	Agent 15	Agent 14	5	0.161
		negotiation	Agent 10	Agent 14	5	0.166
		negotiation	Agent 13	Agent 14	10	0.2
8:30	-10	negotiation	Agent 15	Agent 14	5	0.203
		negotiation	Agent 10	Agent 14	5	0.217
8:45	-10	negotiation	Agent 15	Agent 9	5	0.222
		negotiation	Agent 10	Agent 9	5	0.226
9:00	-10	negotiation	Agent 10	Agent 9	5	0.226
		negotiation	Agent 15	Agent 9	5	0.222
9:15	-10	negotiation	Agent 15	Agent 9	5	0.232
		negotiation	Agent 10	Agent 9	5	0.279
12:00	5	negotiation	Agent 16	Agent 8	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 9	30	0.1
13:15	30	negotiation	Agent 16	Agent 9	30	0.1
13:30	30	negotiation	Agent 16	Agent 9	30	0.093
13:45	30	negotiation	Agent 16	Agent 5	10	0.089
		negotiation	Agent 10	Agent 9	20	0.093
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 9	10	0.112
19:15	-15	negotiation	Agent 16	Agent 9	15	0.112
19:30	-5	negotiation	Agent 16	Agent 14	5	0.112
19:45	-5	negotiation	Agent 16	Agent 14	5	0.112
20:00	-5	negotiation	Agent 15	Agent 9	5	0.158

Table A.16: Table of the results from experiment 15.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	negotiation	Agent 10	Agent 15	5	0.204
		negotiation	Agent 15	Agent 14	5	0.131
		negotiation	Agent 11	Agent 15	8	0.251
		negotiation	Agent 13	Agent 15	2	0.236
7:15	-20	negotiation	Agent 10	Agent 15	5	0.211
		negotiation	Agent 15	Agent 14	5	0.131
		negotiation	Agent 13	Agent 15	10	0.245
7:30	-20	negotiation	Agent 10	Agent 15	5	0.211
		negotiation	Agent 15	Agent 14	5	0.131
		negotiation	Agent 13	Agent 15	10	0.245
8:30	-10	negotiation	Agent 15	Agent 8	5	0.175
		negotiation	Agent 10	Agent 15	5	0.251
8:45	-10	negotiation	Agent 15	Agent 9	5	0.179
		negotiation	Agent 10	Agent 15	5	0.251
9:00	-10	negotiation	Agent 15	Agent 8	5	0.188
		negotiation	Agent 10	Agent 15	5	0.251
9:15	-10	negotiation	Agent 15	Agent 9	5	0.206
		negotiation	Agent 10	Agent 8	5	0.286
12:00	5	negotiation	Agent 16	Agent 4	5	0.1
12:15	5	negotiation	Agent 16	Agent 9	5	0.1
13:00	30	negotiation	Agent 16	Agent 4	10	0.1
		negotiation	Agent 16	Agent 8	20	0.1
13:15	30	negotiation	Agent 16	Agent 8	30	0.1
13:30	30	negotiation	Agent 16	Agent 6	30	0.093
13:45	30	negotiation	Agent 16	Agent 8	30	0.1
16:15	10	negotiation	Agent 16	Agent 8	10	0.112
16:30	10	negotiation	Agent 16	Agent 9	10	0.112
16:45	10	negotiation	Agent 16	Agent 4	10	0.106
19:15	-15	negotiation	Agent 16	Agent 8	15	0.112
19:30	-5	negotiation	Agent 16	Agent 14	5	0.112
19:45	-5	negotiation	Agent 16	Agent 9	5	0.112
20:00	-5	negotiation	Agent 15	Agent 9	5	0.141

Table A.17: Table of the results from experiment 16.

Period	Restriction	Agreement				
		Type	Generator	Consumer	Power (kW)	Price (€/kWh)
7:00	-20	auction	Agent 7	Agent 3	10	0.044
		auction	Agent 10	Agent 3	5	0.044
		auction	Agent 11	Agent 3	5	0.044
7:15	-20	auction	Agent 7	Agent 3	10	0.044
		auction	Agent 10	Agent 3	5	0.044
		auction	Agent 11	Agent 3	5	0.044
7:30	-20	auction	Agent 7	Agent 3	10	0.044
		auction	Agent 10	Agent 3	5	0.044
		auction	Agent 11	Agent 3	5	0.044
8:30	-10	negotiation	-	Agent 16	10	0.074
8:30	-10	negotiation	-	Agent 16	10	0.074
9:00	-10	negotiation	-	Agent 16	10	0.074
9:15	-10	negotiation	-	Agent 16	10	0.074
12:00	5	auction	Agent 0	Agent 16	5	0.083
12:15	5	auction	Agent 0	Agent 16	5	0.083
13:00	30	auction	Agent 0	Agent 16	15.05	0.083
		auction	Agent 1	Agent 16	14.95	0.083
13:15	30	auction	Agent 0	Agent 16	15.68	0.083
		auction	Agent 1	Agent 16	14.32	0.083
13:30	30	auction	Agent 0	Agent 16	16.93	0.083
		auction	Agent 1	Agent 16	13.07	0.083
13:45	30	auction	Agent 0	Agent 16	18.19	0.083
		auction	Agent 1	Agent 16	11.81	0.083
16:15	10	auction	Agent 0	Agent 16	10	0.083
16:30	10	auction	Agent 0	Agent 16	10	0.083
16:45	10	auction	Agent 0	Agent 16	10	0.083
19:15	-15	reduction	-	Agent 16	15	0.051
19:30	-5	reduction	-	Agent 15	5	0.052
19:45	-5	reduction	-	Agent 15	5	0.052
20:00	-5	reduction	-	Agent 15	5	0.052

Bibliography

[1] Netlogo. http://ccl.northwestern.edu/netlogo/., 1999.

[2] Sunil Adhau, Murali Lal Mittal, and Abhinav Mittal. A multi-agent system for distributed multi-project scheduling: An auction-based negotiation approach. *Engineering Applications of Artificial Intelligence*, 25(8):1738–1751, 2012.

[3] Reticular Systems Inc. Alternative Energy Systems Consulting Inc. Using intelligent agents to implement an electronic auction for buying and selling electric power. 1999.

[4] Bo An, Victor R Lesser, David E Irwin, and Michael Zink. Automated negotiation with decommitment for dynamic resource allocation in cloud computing. In *AAMAS*, volume 10, pages 981–988, 2010.

[5] Estefanía Argente Villaplana. *GORMAS: Guías para el desarrollo de Sistemas Multiagente abiertos basados en organizaciones*. PhD thesis, Universitat Politècnica de València, 2008.

[6] Robert Aumann and Sergiu Hart. Handbook of game theory with economic applications. Technical report, Elsevier, 2002.

[7] Amelia Bădică, Costin Bădică, Maria Ganzha, Mirjana Ivanović, and Marcin Paprzycki. Multi-agent spatial sir-based modeling and simulation of infection spread management. In *International Conference on Computational Science*, pages 440–453. Springer, 2021.

[8] Mihai Barbuceanu and Wai-Kau Lo. A multi-attribute utility theoretic negotiation architecture for electronic commerce. In *Proceedings of the fourth international conference on autonomous agents*, pages 239–246, 2000.

[9] Fabio Bellifemine, Federico Bergenti, Giovanni Caire, and Agostino Poggi. Jade—a java agent development framework. In *Multi-agent programming*, pages 125–147. Springer, 2005.

[10] Martin Bichler. *The future of e-markets: multidimensional market mechanisms*. Cambridge University Press, 2001.

[11] Guido Boella and Leendert Van Der Torre. A game-theoretic approach to normative multi-agent systems. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2007.

[12] Olivier Boissier, Jomi Fred Hübner, and Jaime Simao Sichman. Organization oriented programming: From closed to open organizations. In *International Workshop on Engineering Societies in the Agents World*, pages 86–105. Springer, 2006.

[13] Rafael H Bordini, Jomi Fred Hübner, and Michael Wooldridge. *Programming multi-agent systems in AgentSpeak using Jason*. John Wiley & Sons, 2007.

[14] Walter Brenner, Rüdiger Zarnekow, and Hartmut Wittig. *Intelligent software agents: foundations and applications*. Springer Science & Business Media, 2012.

[15] AC da R Costa and Yves Demazeau. Toward a formal model of multi-agent systems with dynamic organizations. In *Proceedings of the International Conference on Multi-Agent Systems, MIT Press, Kyoto, Japan*, volume 431, 1996.

[16] Richard L Daft and Richard L Daft. Teoría y diseño organizacional. 2011.

[17] Rajarshi Das, James E Hanson, Jeffrey O Kephart, and Gerald Tesauro. Agent-human interactions in the continuous double auction. In *International joint conference on artificial intelligence*, volume 17, pages 1169–1178. Lawrence Erlbaum Associates Ltd, 2001.

[18] Mehdi Dastani, Virginia Dignum, and Frank Dignum. Role-assignment in open agent societies. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 489–496, 2003.

[19] MV Dignum et al. *A model for organizational interaction: based on agents, founded in logic*. SIKS, 2004.

[20] MV Dignum et al. A landscape of agent systems in the real world. 2006.

[21] Paul Doran, Valentina Tamma, Terry R Payne, and Ignazio Palmisano. Flexible agreement mechanism for dynamic meaning negotiation. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pages 1599–1600. Citeseer, 2010.

[22] George B Dyson. *Darwin among the machines: The evolution of global intelligence*. Basic Books, 2012.

[23] Peyman Faratin, Carles Sierra, and Nicholas R Jennings. Using similarity criteria to make issue trade-offs in automated negotiations. *artificial Intelligence*, 142(2):205–237, 2002.

[24] Jacques Ferber and Olivier Gutknecht. A meta-model for the analysis and design of organizations in multi-agent systems. In *Proceedings international conference on multi agent systems (Cat. No. 98EX160)*, pages 128–135. IEEE, 1998.

[25] Tim Finin, Richard Fritzson, Don McKay, and Robin McEntire. Kqml as an agent communication language. In *Proceedings of the third international conference on Information and knowledge management*, pages 456–463, 1994.

[26] Association for Computing Machinery. *ACM transactions on autonomous and adaptive systems*. ACM Press, 2006.

[27] LM Gales and BJ Hodge. Teoria de la organizacion - un enfoque estrategico, 2004.

[28] Alfredo Garro, Max Mühlhäuser, Andrea Tundis, Matteo Baldoni, Cristina Baroglio, Federico Bergenti, Paolo Torroni, et al. Intelligent agents: Multi-agent systems. *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, 315, 2018.

[29] Benjamin Gâteau, Olivier Boissier, Djamel Khadraoui, and Eric Dubois. Moiseinst: An organizational model for specifying rights and duties of autonomous agents. *EUMAS*, 2005:484–485, 2005.

[30] Wafa Ghonaim. A multi-agent smart e-market design at work for shariah compliant islamic banking. *International Journal of Economics and Management Engineering*, 11(12):2807–2812, 2018.

[31] A Giret, V Julián, and VJ Botti. Agentes software y sistemas multi-agente: Conceptos, arquitecturas y aplicaciones. aplicaciones industriales de los sistemas multiagentes, 2005.

[32] Jorge Gonzalez-Palacios and Michael Luck. Towards compliance of agents in open multi-agent systems. In *International Workshop on Software Engineering for Large-Scale Multi-agent Systems*, pages 132–147. Springer, 2006.

[33] Ehsan Hejazi. Multi-agent machine learning in self-organizing systems. *Information Sciences*, 581:194–214, 2021.

[34] Jomi F Hübner, Jaime S Sichman, and Olivier Boissier. S-moise+.

[35] Ivar Jacobson, Grady Booch, and James Rumbaugh. The unified modeling language. *University Video Communications*, 1996.

[36] Nicholas R Jennings, Peyman Faratin, Alessio R Lomuscio, Simon Parsons, Carles Sierra, and Michael Wooldridge. Automated negotiation: prospects, methods and challenges. *International Journal of Group Decision and Negotiation*, 10(2):199–215, 2001.

[37] Maria Gabriela Juarez Juarez. Un mercado virtual genérico basado en sistemas multi-agente. Master's thesis, Universitat Politècnica de València, Valencia, Spain, 2018.

[38] Sarit Kraus and Ronald C Arkin. *Strategic negotiation in multiagent environments*. MIT press, 2001.

[39] Sarit Kraus, Katia Sycara, and Amir Evenchik. Reaching agreements through argumentation: a logical model and implementation. *Artificial Intelligence*, 104(1-2):1–69, 1998.

[40] D Krokuo. Estrategias de entrada en el emergente mercado electrónico chino, 2016.

[41] Mohamed Haitam Laarabi and Raffaele Bruno. A generic software framework for carsharing modelling based on a large-scale multi-agent traffic simulation platform. In *International Workshop on Agent Based Modelling of Urban Systems*, pages 88–111. Springer, 2016.

[42] Zhengping Li, Cheng Hwee Sim, and Malcolm Yoke Hean Low. A survey of emergent behavior and its impacts in agent-based systems. In *2006 4th IEEE international conference on industrial informatics*, pages 1295–1300. IEEE, 2006.

[43] Michael Luck, Mark d'Inverno, et al. A normative framework for agent-based systems. *Computational & Mathematical Organization Theory*, 12(2):227–250, 2006.

[44] Xudong Luo, Nicholas R Jennings, Nigel Shadbolt, Ho-fung Leung, and Jimmy Homan Lee. A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments. *Artificial Intelligence*, 148(1-2):53–102, 2003.

[45] Pattie Maes, Robert H Guttman, and Alexandros G Moukas. Agents that buy and sell. *Communications of the ACM*, 42(3):81–ff, 1999.

[46] Bireshwar Dass Mazumdar and Shubhagata Roy. Multi-agent paradigm for b2c e-commerce. In *Artificial Intelligence and Machine Learning in Business Management*, pages 29–52. CRC Press, 2021.

[47] Lucas Silveira Melo, Raimundo Furtado Sampaio, Ruth Pastôra Saraiva Leão, Giovanni Cordeiro Barroso, and José Roberto Bezerra. Python-based multi-agent platform for application on power grids. *International transactions on electrical energy systems*, 29(6):e12012, 2019.

[48] Bernard Moulin, Walid Chaker, Jimmy Perron, Patrick Pelletier, Jimmy Hogan, and Edouard Gbei. Mags project: Multi-agent geosimulation and crowd simulation. In *International Conference on Spatial Information Theory*, pages 151–168. Springer, 2003.

[49] H Juergen Mueller. Negotiation principles. *Foundations of distributed artificial intelligence*, 9:211, 1996.

[50] Oliver Obst. Spark-a generic simulator for physical multiagent simulations. *Computer Systems Science and Engineering*, 20(5):347–356, 2005.

[51] Julian Padget and Axel Polleres. Mallow workshop on coordination, organization, institutions and norms in agent systems & on-line communities (coin@ mallow2009). *Multi-Agent Logics, Languages, and Organisations*, page 73, 2009.

[52] Javier Palanca, Andrés Terrasa, Vicente Julian, and Carlos Carrascosa. Spade 3: Supporting the new generation of multi-agent systems. *IEEE Access*, 8:182537–182549, 2020.

[53] J Peiró. Psicología de la organización, universidad nacional de educación a distancia. *Psicología. Quinta edición, Madrid*, 1991.

[54] Steve Phelps, Kai Cai, Peter McBurney, Jinzhong Niu, Simon Parsons, and Elizabeth Sklar. Auctions, evolution, and multi-agent learning. In *Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning*, pages 188–210. Springer, 2005.

[55] Katia Potiron, Amal El Fallah Seghrouchni, and Patrick Taillibert. *From fault classification to fault tolerance for multi-agent systems*, volume 8. Springer, 2013.

[56] Howard Raiffa. *The art and science of negotiation*. Harvard University Press, 1982.

[57] Anand S Rao. Agentspeak (l): Bdi agents speak out in a logical computable language. In *European workshop on modelling autonomous agents in a multi-agent world*, pages 42–55. Springer, 1996.

[58] Stephen P Robbins. *Comportamiento organizacional*. Pearson educación, 2004.

[59] Andrew Rollings and Ernest Adams. *Andrew Rollings and Ernest Adams on game design*. New Riders, 2003.

[60] Jeffrey S Rosenschein and Gilad Zlotkin. *Rules of encounter: designing conventions for automated negotiation among computers*. MIT press, 1994.

[61] Carlos Sanchís Gómez. Diseño de un mercado de electricidad local para la explotación óptima de recursos energéticos flexibles maximizando el beneficio de los agentes involucrados. Master's thesis, Universitat Politècnica de València, Valencia, Spain, 2021.

[62] Tuomas W Sandholm. Distributed rational decision making. *Multiagent systems: a modern approach to distributed artificial intelligence*, pages 201–258, 1999.

[63] Gabriel Santos, Tiago Pinto, Hugo Morais, Tiago M Sousa, Ivo F Pereira, Ricardo Fernandes, Isabel Praça, and Zita Vale. Multi-agent simulation of competitive electricity markets: Autonomous systems cooperation for european market modeling. *Energy Conversion and Management*, 99:387–399, 2015.

[64] Gabriel Santos, Tiago Pinto, Isabel Praça, and Zita Vale. Mascem: Optimizing the performance of a multi-agent system. *Energy*, 111:513–524, 2016.

[65] Sergey Satunin and Eduard Babkin. A multi-agent approach to intelligent transportation systems modeling with combinatorial auctions. *Expert Systems with Applications*, 41(15):6622–6633, 2014.

[66] Carles Sierra, Vicent Botti, and Sascha Ossowski. Agreement computing. *KI-Künstliche Intelligenz*, 25(1):57–61, 2011.

[67] Jose M Such, Ana García-Fornes, Agustín Espinosa, and Joan Bellver. Magentix2: A privacy-enhancing agent platform. *Engineering Applications of Artificial Intelligence*, 26(1):96–109, 2013.

[68] Katia P Sycara. Persuasive argumentation in negotiation. *Theory and decision*, 28(3):203–242, 1990.

[69] William Thomson. Cooperative theory of bargaining i: Classical. In *Cooperation: Game-Theoretic Approaches*, pages 9–24. Springer, 1997.

[70] Isaac Guzmán Valdivia. *Reflexiones en torno al orden social*. Editorial Jus, 1983.

[71] Hal R Varian. *Microeconomic analysis*. Number 338.5 V299m 1992. WW Norton, 1992.

[72] Equipo Vertice. E-commerce. aplicación y desarrollo. *Malaga: Vértice*, 2010.

[73] John A Wagner, John R Hollenbeck, and Luisa Reyes Lopez. *Comportamiento organizativo: Consiguiendo la ventaja competitiva*. Thomson, 2004.

[74] William E Walsh and Michael P Wellman. A market protocol for decentralized task allocation. In *Proceedings International Conference on Multi Agent Systems (Cat. No. 98EX160)*, pages 325–332. IEEE, 1998.

[75] Gerhard Weiss. *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT press, 1999.

[76] Michael Wooldridge. *An introduction to multiagent systems*. John wiley & sons, 2009.

[77] Michael Wooldridge, Jörg P Müller, and Milind Tambe. Agent theories, architectures, and languages: a bibliography. pages 408–431, 1995.

[78] Franco Zambonelli, Nicholas R Jennings, and Michael Wooldridge. Organisational abstractions for the analysis and design of multi-agent systems. In *International Workshop on Agent-Oriented Software Engineering*, pages 235–251. Springer, 2000.

[79] Yang Zhou, Zhengyu Shi, and Libo Wu. Green policy under the competitive electricity market: An agent-based model simulation in shanghai. *Journal of Environmental Management*, 299:113501, 2021.