electronics

MDPI

Article

A Comprehensive Modeling of the Discrete and Dynamic Problem of Berth Allocation in Maritime Terminals

Sami Mnasri [1,2,*] and Malek Alrashidi [1,*]

[1] Department of Computer Science, Applied College, University of Tabuk, Tabuk 47512, Saudi Arabia
[2] CNRS-IRIT (RMESS), University of Toulouse (UT2J), 31000 Toulouse, France
* Correspondence: smnasri@ut.edu.sa (S.M.); mqalrashidi@ut.edu.sa (M.A.)

Abstract: In this study, the discrete and dynamic problem of berth allocation in maritime terminals, is investigated. The suggested resolution method relies on a paradigm of optimization with two techniques: heuristic and multi-agent. Indeed, a set of techniques such as the protocol of negotiation named contract net, the multi-agent interactions, and Worst-Fit arrangement technique, are involved. The main objective of the study is to propose a solution for attributing m parallel machines to a set of activities. The contribution of the study is to provide a detailed modeling of the discrete and dynamic berth allocation problem by establishing the corresponding models using a multi-agent methodology. A set of numerical experiments are detailed to prove the performance of the introduced multi-agent strategy compared with genetic algorithm and tabu search.

Keywords: dynamic and discrete berth allocation; scheduling; parallel machines; multi-agent negotiation; heuristics

check for updates

Citation: Mnasri, S.; Alrashidi, M. A Comprehensive Modeling of the Discrete and Dynamic Problem of Berth Allocation in Maritime Terminals. *Electronics* **2021**, *10*, 2684. https://doi.org/10.3390/electronics10212684

Academic Editor: Javid Taheri

Received: 16 September 2021
Accepted: 1 November 2021
Published: 3 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Presentation and Motivations for the Berth Allocation

The problem of scheduling ships at berths, called Berth Allocation Problem (BAP), is a problem that consists of assigning the berths to the ships/vessels in maritime terminals, with a specific planning horizon.

Indeed, as shown in Figure 1, a set of vessels are present at the terminal on dates that are known in advance. Precise schedules must be assigned for the attribution of the berths (a problem of assignment) to the vessels (a problem of scheduling) with handling activities. Different constraints/restrictions (which will be detailed in the rest of the paper) should be considered such as the length of the quay and the handling time of the containers makes the berths critical resources and involves wait-time for access into the berth. However, to increase the productivity in maritime ports, reducing berthing costs and waiting time is essential. The dynamic and discrete berth allocation problem (DDBAP) is a real-world practical engineering problem.

For shipowners, the time at the berth is a period without generation of revenue [1]. When assigning berths, we must take into account the space available to accommodate the ship and the future needs of incoming ships. Planning must therefore be carried out over a long enough period to avoid queue.

The main decision made when berthing is the choice of "where" and "when" the ships should birth. Therefore, planning must identify the time and locations for arrivals of ships. The purpose of this planning is to reduce the overall cost, in terms of time, of the berth, knowing the estimated time of arrival, and the duration of operations that represent the temporal aspect of the problem. The problem can therefore be modeled in a two-dimensional space.

The quay can be represented by a horizontal line having a predetermined length. The ships can be represented by a rectangle where the rectangle length indicates the length of the ship (with marginal safety spaces) and the rectangle width represents the duration of the handling operation [2]. The set of obtained rectangles must be positioned in the

"decision space" with no overlaps while respecting a set of constraints. As a result, the ship berthing problem becomes a constrained two-dimensional bin packing problem.

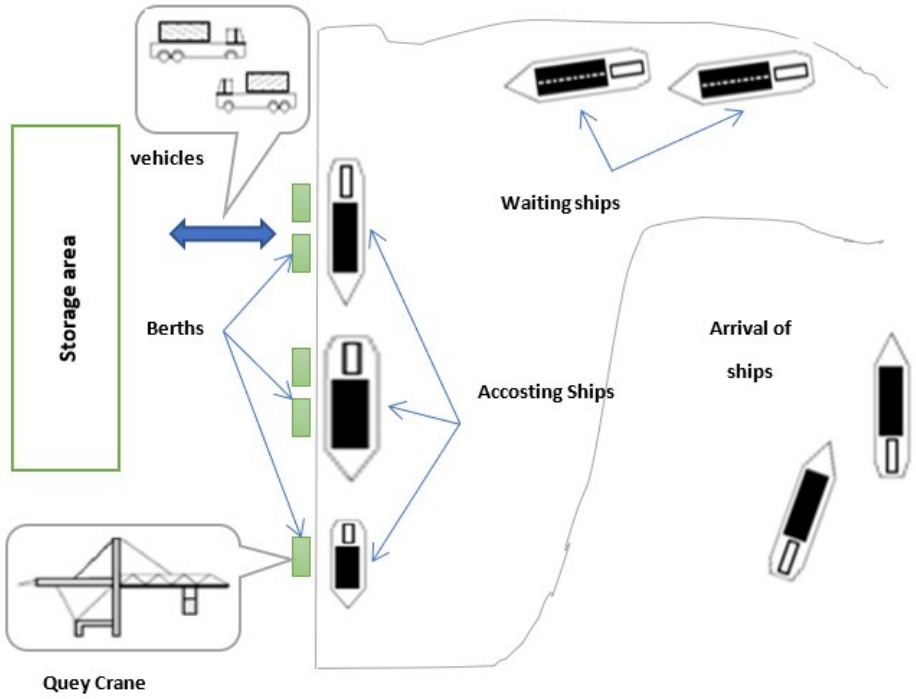


Figure 1. Graphical illustration of the process of berthing of ships.

As shown in Figure 2, the aim of the DDBAP problem is to establish an assignment and a berth allocation that minimizes the overall time of service of all the ships. This service time is the time separating the arrival of the ship on the port to the end of handling activities [3]. On the other hand, there are several other objectives which differ according to the variant and the constraints taken into account during the formulation of the problem.

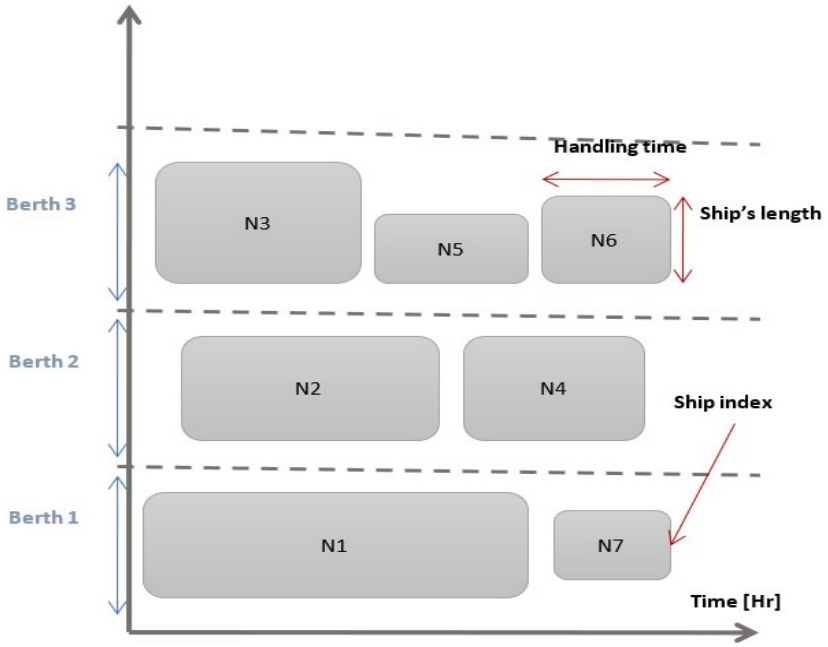


Figure 2. Spatiotemporal representation of a berthing operation with 7 vessels and 3 berths.

DDBAP belongs to real-world engineering problems. This type of problems is complex and needs intelligent optimization methods to resolve it [4,5]. Indeed, DDBAP is categorized as "NP-Hard" [6] given the complexity of the considered constraints of the berth, which includes several factors and parameters. Even if there is a single berth, the problem will be reduced to the problem of minimizing total execution time (or minimizing other parameters depending on the intended purpose) on one machine.

Indeed, the BAP can be modeled as a 2D cutting stock problem having several constraints to satisfy, such as needed time for handling containers, the waiting time for berthing, the occupation of quay cranes and terminal trucks. This means it is unresolvable using deterministic algorithms and exact solvers such as CPLEX [7]. Besides, unexpected events such as weather conditions, schedule modification, or equipment problems necessitate fast real-time decision-making and may make the problem of berthing more challenging and harder to solve. For these reasons, heuristic and approximative methods and algorithms should be used to resolve complex problems [8] such as the BAP and other terminal problems. In this regard, numerous heuristics were applied to optimize the berthing.

The contributions proposed by this study are as follows:

- The discrete and dynamic berth allocation problem is introduced and formulated as an integer linear problem with a set of constraints based on real world assumptions.
- An efficient multi-agent system (MAS) is introduced to resolve the introduced DDBAP. The proposed multi-agent modeling is based on Multi-Agent-Systems-Engineering (O-MaSE), a modeling methodology dedicated for agent-based dynamic contexts and systems. O-MaSE relies on a set of specific diagrams which are detailed and explained for the context of the DDBAP.
- The proposed multi-agent architecture is implemented and developed. Then, a set of experiments are achieved using the Jade multi-agent platform to assess the behavior of the proposed system with other heuristic methods such as the genetic algorithms.

2. Related Works

Amultitude of methods are used by several researchers to solve berth planning problem. Among the most widely used methods we cite tabu search, Branch and Bound, genetic algorithms, heuristic, hybrid and evolutionary algorithms, linear programming, and multi-agent systems. Table 1 gives a summary of the most relevant studies resolving the problem of berth planning.

Table 1. Illustration and discussion of relevant studies investigating the berth planning problem.

Type of Berth Problem	Author/Year	Used Approach	Considered Objective(s)	Deductions/Discussion of the Study
Discrete problem	Cahyono et al., 2020 [9]	Dynamic predictive scheduling using Monte Carlo algorithm.	Minimization of total service time (standby time + handling time).	(+) Dataset from a real terminal (Tanjung Priuk, Jakarta) is used. (+) A dynamic formulation of the berthing process is proposed.
	Lassoued et al., 2019 [10]	Mathematical model was introduced then solved by CPLEX.	Minimization of the global time of service of the vessels.	Resolution of the dynamic berth allocation problem in bulk ports.
	Arram et al., 2015 [11]	A bio-inspired Bird mating algorithm (BMO).	Minimization of the global waiting and processing time of the vessels.	The waiting time indicates the relevance of ships. The performance of the BMO algorithm is compared with other optimizers.
	Schepler et al., 2019 [12]	Probabilistic optimization based on Dynamic programming and Iterated tabu search.	Minimization of the global weighted flow-time.	Instances from the literature were used to evaluate the algorithms. Obtained results indicate the adaptability of the introduced algorithm to both reactive and proactive phases regarding the needed computation times.

Table 1. *Cont.*

Type of Berth Problem	Author/Year	Used Approach	Considered Objective(s)	Deductions/Discussion of the Study
Continuous problem	Iris et al., 2017 [13]	Integer linear programming	Identifying berthing positions in partitioned continuous berth. Identifying berthing start and end time. Determining quay cranes for ships	(+) The state-of-the-art methods are investigated for the dynamic and the continuous berth allocation and quay crane problem. (−) A larger benchmark for the problem should be used.
	Chen et al., 2017 [14]	Genetic Algorithms	Minimizing the penalty cost.	The distance of deviation from the reference point of berth and the delay of leaving time are considered in the modeled problem.
	Nourmohammad zadeh et al., 2020 [15]	Pareto Simulated Annealing (PSA) and a multi-Objective Simulated Annealing (MOSA) algorithm	Minimization of the deviations from target locations. Minimization of the departure delays of the vessels.	The introduced PSA and MOSA algorithm outperform other traditional methods (tests achieved using exact solver and with ϵ-constraint).
Simultaneous berth and quay crane allocation problem	Abou kasm et al., 2019 [16]	Optimization modeling	Minimize the needed time for handling of the containers.	Three terminal problems are integrated and simultaneously resolved: the quay crane scheduling problem (QCSP), the quay crane assignment problem (QCAP), and the berth allocation problem (BAP). Real experimental tests were achieved on the terminal of Abu Dhabi.
	Atencio et al., 2018 [17]	genetic algorithm, and colony optimization and simulated annealing.	Minimization of the total penalty cost.	Experimental tests were achieved on a real case in bulk ports, and using additional instances.
	Chu et al., 2020 [18]	ϵ-constraint based two-phase iterative algorithm.	Minimization of the global turnaround time of ships. Minimization of the global penalty cost of maintenance of the quay cranes.	Additional constraints are considered, such as the preventive activities of maintenance. Small size and large size instances of the problem are considered.

According to the investigated studies, the BAP is a relevant operation to carefully enhance in maritime ports. However, it raises numerous issues. The most used classification of the BAP in these studies relies on three categories (continuous, discrete, and hybrid) according to the layout of the quay. The continuous BAP assumes that the quay is continuous with no specific positions for the berths. The discrete BAP supposes that the quay is composed of a set of berths where each berth receives one ship at a time. In addition, the investigated studies highlight the most important objectives and constraints to be taken into consideration when establishing the berth allocation schedule. They concern essentially the characteristics of vessels (type, speed, capacity, and compatibility with berths). Besides, it is noticed that the majority of research resolves the berth allocation problem using deterministic algorithms. However, handling the containers in terminals is a dynamic and complex operation with a set of stochastic factors such as delays and accidents. Hence, recent studies begin focusing on new factors such as the incertitude of vessel arrival or the weather condition's influence on the BAP. Moreover, the integration of the BAP with other related problems in the terminal, such as the quay crane and container stocking, remain insufficient. The following is a look at recent studies focusing on combined QCAP and BAP in a dynamic and uncertain context:

In [19], the authors introduce a data-driven optimization algorithm to resolve the tactical berth allocation problem (TBAP). The algorithm aims to minimize the cost of berthing. A comparison of the suggested algorithm with other models for different levels of time uncertainty is achieved. However, the scalability of the proposed model is not guaranteed for large-size instances of the TBAP.

The study in [20] suggests combining the problems of vessel sequencing and berth planning. An integer linear model and a column generation algorithm were presented. The latter relies on a set-partitioning concept to compute the lower bound of the solution. The findings of the study highlights the performance of the proposed algorithm compared to different heuristic techniques such as the simulated annealing and the genetic algorithm.

Despite a detailed formulation being given in the study, tests were conducted on a small set (55 vessels and 43 berths). The time of execution is also a drawback of the suggested model (three minutes were necessary to resolve this small-size set).

In [21], there was a combined problem of QCAP and BAP. The aim was to obtain the best berthing positions, time, and quay cranes for a set of waiting vessels with the minimum cost regarding the time of departure of vessels. A K-means based clustering algorithm was used to decompose the uncertainty set into several non-overlapping clusters. The results shows that the introduced system is more efficient than a deterministic approach in handling uncertainties. However, different constraints should be added such as the penalty and capacity constraints.

The study in [22] suggest to simultaneously resolve the quay crane and the tactical berth allocation problems. A linear modeling was presented with different realistic constraints. Then a heuristic model was applied on a problem instance of 10 vessels and 100 berths. The pseudo-code of the algorithm was given. However, additional objectives should be considered such as the degree of berth use, the relocation of containers and the yard occupancy.

Authors in [23] propose a study to resolve the BAP taking into consideration the effect of weather circumstances. The approach used was relied on a two-stage optimizer which identifies the needed handling time then proposes a mixed-integer programming for the problem. For large-size instances of the BAP, a particle swarm optimizer (PSO) and machine learning algorithm were implemented. However, in this study, the BAP is resolved separately from the quay crane and yard assignment problems.

In [24], a real dynamic berth allocation model was implemented in the terminal of Paranaguá in Brazil. Two optimizers were combined: a local search dynamic programming with a genetic algorithm. Then two heuristics were used: an elimination process procedure and a reduced neighborhood for unpromising areas in the search space. The problem complexity was investigated and the behaviors of the genetic and local search were compared with the two heuristics performance on the real data from the Paranaguá terminal. However, small instances for experiments were used.

Authors in [25] introduces a multi-user model in the port of Cartagena in Spain to resolve the BAP by reducing the delays of serving ships. The indicated port was a liquid bulk terminal characterized by a little number of quays. An optimizer was developed to simultaneously enhance the arrival time and the berth assignment. Despite the fact that the study is based on real data and is multi-user, the behavior of the latter is not taken into consideration.

The study in [26] addresses the problem of berth allocation in automotive terminals. Car flow constraints were formulated in terms of traffic fluidity and collision risk in an integer linear model. The aim was to simultaneously enhance the car flow crossing effect and the overall handling time. A Complexity analysis was given. However, there is a non-conformity between the modeling assumptions and those used in the experimental tests. Moreover, important constraints, such as the capacity and positions of storage yards, are not considered in the suggested formulation.

The study in [27] highlights the interest of the used policy for optimizing the berth allocation in multimodal terminals. The proposed approach relied on a learning system which was used to resolve Markov decision problems. However, the suggested system relies only on simulations and the allocation-of-berths problem is not solved simultaneously with the quay cranes one.

In [28], authors investigate the problem of the movement of the truck yard in seaports. The aim was to study how the berth allocation and the container stacking influences the distance traveled by the trucks. The used approach relied on a hybrid tabu and genetic method for simultaneously resolving the container stacking and the berth and yard allocation problems. However, numerous constraints are not considered in this study such as the unloading process which influences the route planning of trucks. Hence, a tradeoff between the processes of loading and unloading can be interesting.

3. Mathematical Formulation of the DDBAP

The dynamic formulation assumes that ships can arrive at the port before or after the start of the berth assignment process and so we can extend the planning to reach possible horizons. The goal is to minimize berthing and service waiting times. The dynamic problem is formulated as follows:

$$\text{Minimizing} \sum_{i\in B}\sum_{j\in V}\sum_{k\in O}\{((\mathrm{T}-\mathrm{k}+1)\mathrm{Cij}+\mathrm{Si}-\mathrm{Aj}\}\mathrm{Xijk}+\sum_{i\in B}\sum_{j\in V}\sum_{k\in O}(\mathrm{T}-\mathrm{k}+1)\mathrm{Yijk} \quad (1)$$

The constraints are as follows:

$$\sum_{i\in B}\sum_{k\in O}\mathrm{Xijk}=1 \text{ for } j\in V \quad (2)$$

$$\sum_{j\in V}Xijk \le 1 \text{ i} \in B, k \in O \quad (3)$$

$$\sum_{i\in V}\sum_{k\in Pk}(\mathrm{Cil\ Xilm}+\mathrm{Yilm})+\mathrm{Yijk}-(Aj-\mathrm{S}i)\ \mathrm{Xijk} >= 0 \text{ For } i\in B, j\in Wi, k\in O \quad (4)$$

$$\mathrm{Yijk} \ge 0 \text{ for } i\in B, j\in V, k\in O \quad (5)$$

$$\mathrm{Xijk}\ \epsilon\ \{0,1\} \text{ for k } \epsilon \text{ O, j } \epsilon \text{ V, i } \epsilon \text{ B} \quad (6)$$

with:

- i from 1 to I in B is the set of the available berths.
- j from 1 to T in V is the set of the available ships.
- k from 1 to T in O is the set of the available service orders.
- Si is the time in which the location i becomes free and may be added to the schedule.
- Aj the time of arrival of j.
- Cij the needed time for handling the vessel j in the location i.
- Xijk is equal to 1 if the vessel j is handled as the kth ship in the location i, 0 else.
- Pk sub-set from O where Pk = {p | p < k € O}
- Wi sub-set of ships such as Aj $\ge$ Si.
- Yijk free time (from location i) separating the arrival of the kth ship from the departure of the (k − 1)th ship when the vessel j is served as the kth ship.

The aim of the objective function (1) is to minimize the overall time for each ship. The constraints (2), (3), and (6) relate to the assignment of the ship. The constraint (4) guarantees to handle the ships only after its arrival. We assume that Aj, Si, and Cij are integer. Yijk indicates the priority that relates the arrival of the vessel and its handling. To explain that the service is performed only after the arrival, Yijk is defined as the difference of time separating the beginning of the service of the boat j from the time of departure of its predecessor.

4. O-MaSE Modeling Methodology

There are several studies that are interested in the methodology of analysis and design of MAS. However, there is no standard method because each research group conducts different method withdifferent aspects. Indeed, the Multi-Agent-Systems-Engineering O-MaSE is an object-oriented methodology which aims to design a society of agents relying on an organizational meta-model [29]. Being of clear presentation, easy to use and recognized by various works, we propose to follow this methodology and the associated platform AgentTool [30] to design our System.

The development process of this methodology covers the analysis, design, and implementation phase using the AgentTool:

- The analysis phase: it is made up of the following three stages:
  - Identify goals: a "goal" is considered as an objective of the system. The goals are structured and organized according to their importance.

- Determine the desired behaviors of the system and describe the exchanged messages.
- Define roles: each goal is transformed into a role and this role will be played by an agent.

- The design phase: composed of four stages:
  - Create agent's classes: agent's classes are identified from roles.
  - Build conversations: define protocols between agents.
  - Assemble the classes of agents: This step is concerned with the creation of internal classes of agents according to the chosen architectural model (Belief-Desire-Intention or others).
  - Define the structure of the system: this structure is presented by the O-MaSE deployment diagrams.
- The implementation phase: AgentTool facilitates the development of applications modeled by this methodology. It allows the verification, code generation, and reuse of components.

The O-MaSE methodology contains models such as the model of goals, of roles, of agents, of protocols and of plans. Figure 3 shows the process for developing these diagrams.

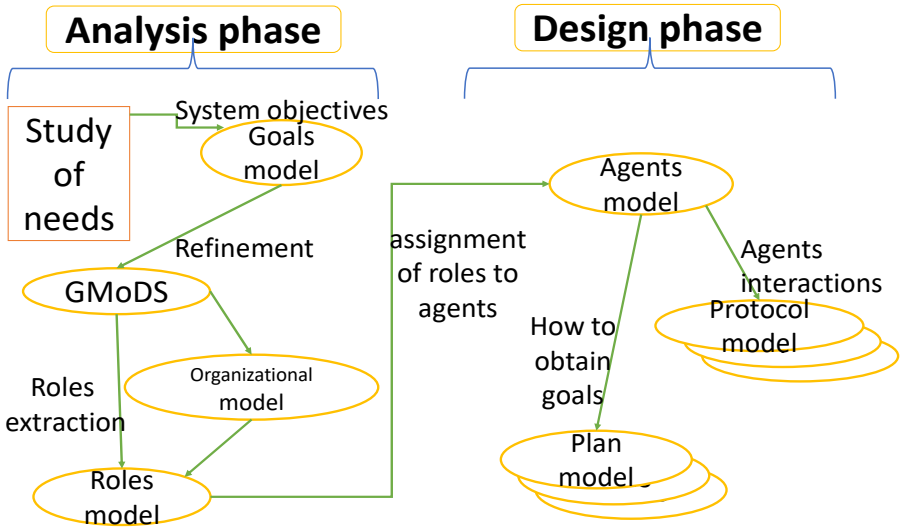


Figure 3. General Bi-phase process of O-MaSE.

We propose the following multi-agent architecture which includes the following agents: the berth agent, the scheduler agent, the ship agent, and the objects (containers). The scheduling agent is the mediator between the berth and ship agents, who communicate with each other to find a solution while respecting the constraints.

4.1. Analysis Phase

This phase concerns the study of needs, it mainly covers the identification of goals and the definition of roles. We present the models of goals, organization and roles.

4.1.1. Goal Model

It is, as defined on the official AgentTools website [30]; an AND/OR tree of goals. Goals in this tree can have any number of attributes and can have priorities and trigger its relationships. The occurred events may also be added if one goal is triggered from another.

The main purpose of our system is to schedule and assign ships to berths. To achieve this assignment, the system must create and be able to manage vessel and post agents, assist and serve vessels requesting berthing, and seek and provide a berthing position and ultimately build a close-to-optimal solution. Figure 4 shows the goal tree.

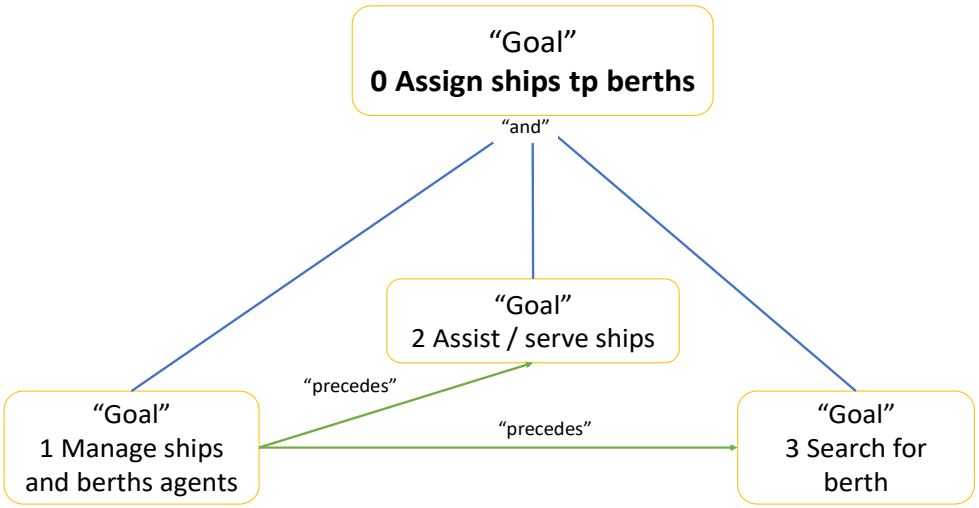


Figure 4. Goal model of O-MaSE.

According to DeLoach [29], the task of goal refinement is to capture the goal model and its dynamic aspect and to define the goals with a method named Assign Precede Trigger Analysis. We then obtain a Goal Model for Dynamical Systems (GMoDS) refined-version. If goal A should be accomplished before goal B is pursued, then goal A precedes goal B. The new goals are generally produced to satisfy new events. Indeed, goal refinement is employed to capture the constraints between goals, to identify the goals to create and to document the required parameters. After refinement of the goals, we will have the diagram of Figure 5.

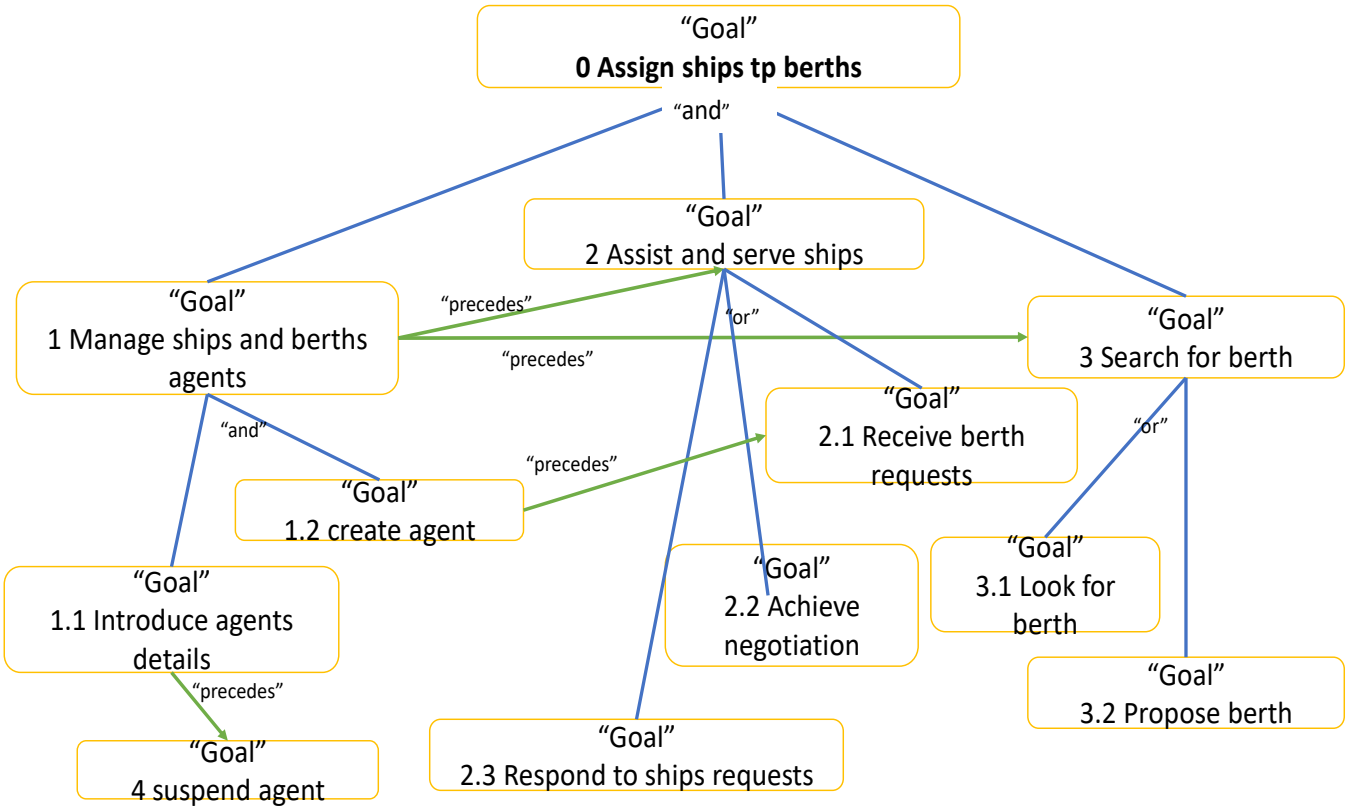


Figure 5. Refined goal model of O-MaSE.

4.1.2. Organizational Model

The organizational model defines the relationships between roles. According to the same definition of the official AgentTools website [30], the organizational model illustrates the interactions between the external actors and the organization. Organizational models are static and include services, protocols, goals, organizations, and actors. They have as input the derived GMoDS model. The purpose of this activity is to determine the relationships of the organization (the system) with the external actors, which may be agents, actors, or organizations. To identify the organization interfaces, several classes from external entities may be checked to decide whether the organization should communicate with them. When the organization is a sub-organization of a bigger organization, the communications between the agents in the higher-level organization and the sub-organization identify the initial communications with this secondary organization.

In the case of our system, the interactions are formulated as protocols identified as a set of arrows going from the trigger of the communication to its destination, which can be the system or the external actor. Figure 6 shows our organizational model.

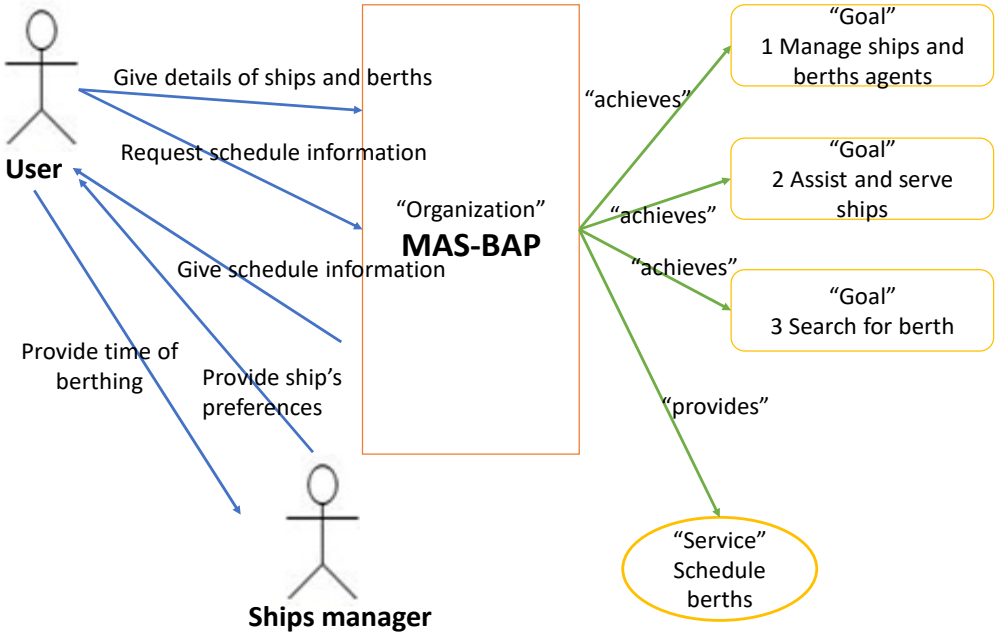


Figure 6. Organizational model of O-MaSE.

4.1.3. Role-Model

The role-model determines all the roles in the organization and the communications between them and with the external actors. The purpose of role modeling is to assign each sheet goal of the organizational goal template to a predefined role. A simple role is often created for each sheet goal. However, sometimes it is necessary to allow a simple role to achieve more than a goal. Indeed, the designer should also determine the communications between roles and with the external actors. The communications with the external actors may be extracted from the organizational model [29].

The official AgentTools website [30] defines this diagram as a representer of all the organization roles with other goals and with the communication protocols from the organization. The models of roles are therefore static and can include actors, roles, goals, possibilities, services, and protocols; and relationships between the entities include 'require', 'provide' or 'inheritance'. Figure 7 illustrates our role model (protocols between roles are not specified so as not to clutter the diagram).

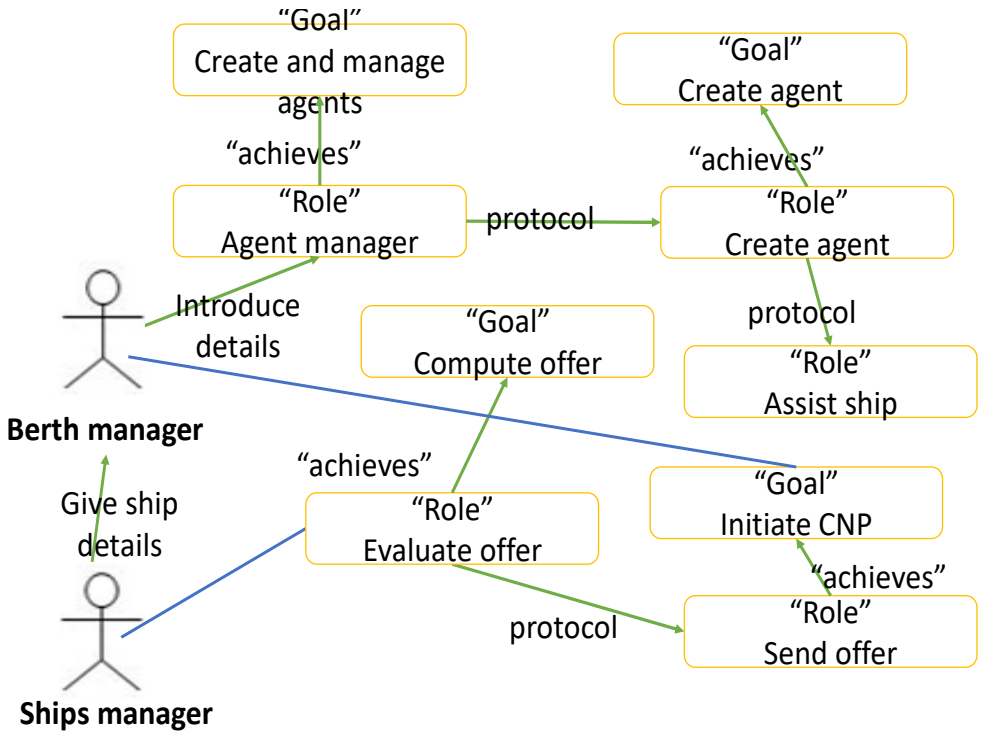


Figure 7. Role model of O-MaSE.

The development of the role model uses the organizational model and GMoDS described earlier in the process. The main task then is to determine the roles necessary for the organization and the communications between these actors. These interactions are presented as protocols. Actors in the organizational model must appear as actors in the role model. Thus, protocols between the organization and the actors should link an actor and a specific role in the role model.

Besides, a role is assigned to each "leaf" from the tree of the GMoDS goal. We use the term "achieves" on the roles to indicate the goals achieved by the latter roles. Hence, a role may achieve more than one goal while a specific role is supposed to achieve at least one goal.

At this stage all the roles are defined and the analysis stage is completed. Based on the models determined in the phase of analysis, we approach the second phase of the methodology.

4.2. The Design Phase

In this phase, we are interested in the modeling of agents, their internal behavior and their social behavior towards other agents. We present the models of protocol, plan, agent, and many other O-MaSE diagrams.

4.2.1. Agent Behaviors

For our system, we have at least three plans since it has three types of agents. We can also model a plan for each role played by the agent since each role achieves a goal. Figure 8 shows the plan diagram corresponding to the goal "launch a CNP". Figure 9 describes the plan diagram corresponding to the "manage agents" goal.

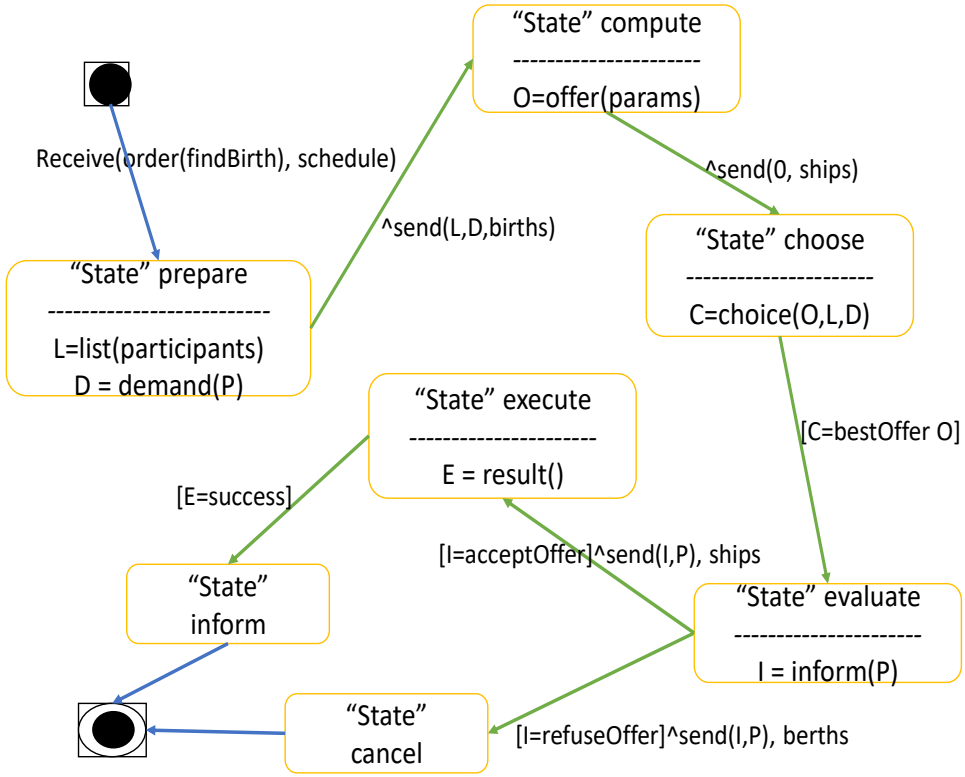


Figure 8. Model of the satisfaction plan for the goal "launching the CNP".

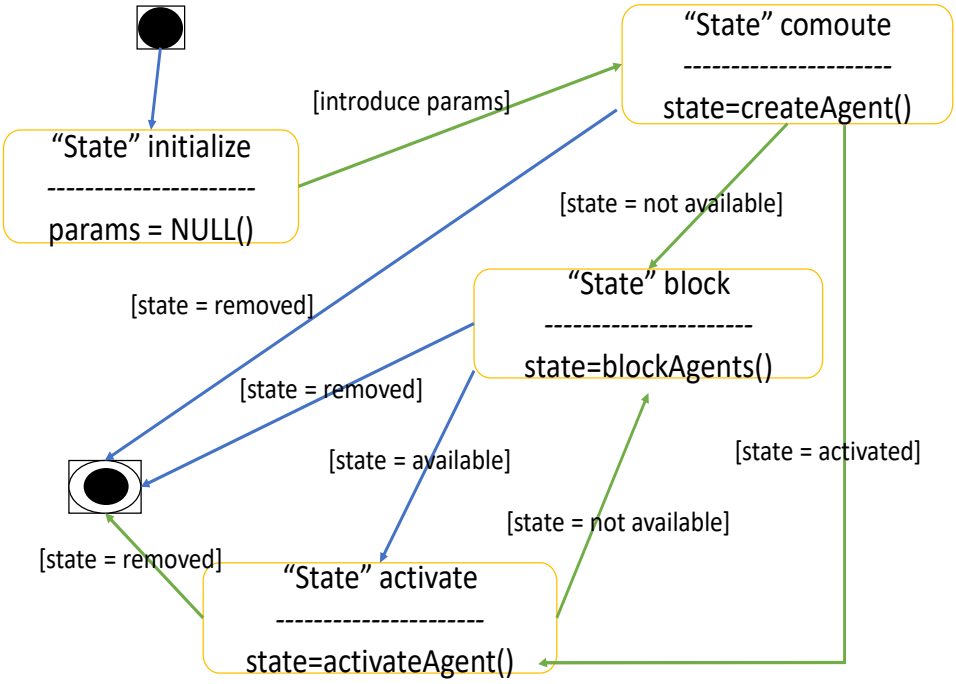


Figure 9. Model of "the satisfaction plan" for the goal "managing agents".

In what follows, we investigate the interactions and communications between the agents of the system.

4.2.2. Interactions and Communications between Agents

Interactions and communications between agents are described using protocol models which allow modeling the protocols determined in the model of roles and in the model

of agents. The model details are presented in the form of messages exchanged between the external actors and the agents. As defined by the official AgentTools website [30], this model represents the diagrams of protocols as shown in AgentUML. It illustrates the orders of messages sent between organizations, roles, and external actors. It can include iterative and alternative structures of the system.

The purpose of protocol models is to determine the specifications of communication between the roles. The protocols are being indicated in the organization, role, or agent models; hence, we can decide which protocols we will specify. If the protocols of the role model are indicated by the protocol models, the agent having these roles must inherit the protocols.

Figure 10 illustrates the protocol diagram representing the interactions between the scheduling agent and the ship agent and shows the roles involved in negotiation.

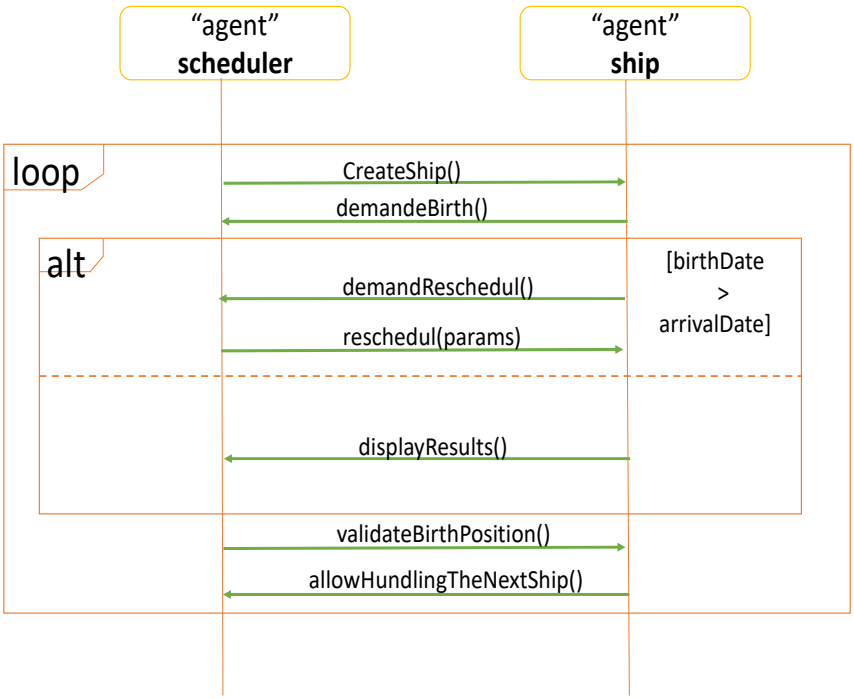


Figure 10. Protocol model between the "scheduling agent" and the "ship agent".

5. Experimental Results

To assess the efficiency of our method, we compare our results with other methods. Figure 11 shows the communications between the different agents of the system.

5.1. Comparison with Branch-and-Bound and Genetic Algorithm

5.1.1. Obtained Results for Small Size Instances of DellAmico and E. Mokotoff

In [31], the authors introduce a Branch-and-Bound strategy relying on the computation of upper and lower bound, and a set of dominance criteria. In [32], the authors suggest a simplex and Branch-and-Bound method using the same instances. In our study, we compare our method only with the Dell'Amico results in [31] since their found results surpass those in [32] for all instances. Figure 12 illustrates the obtained results.

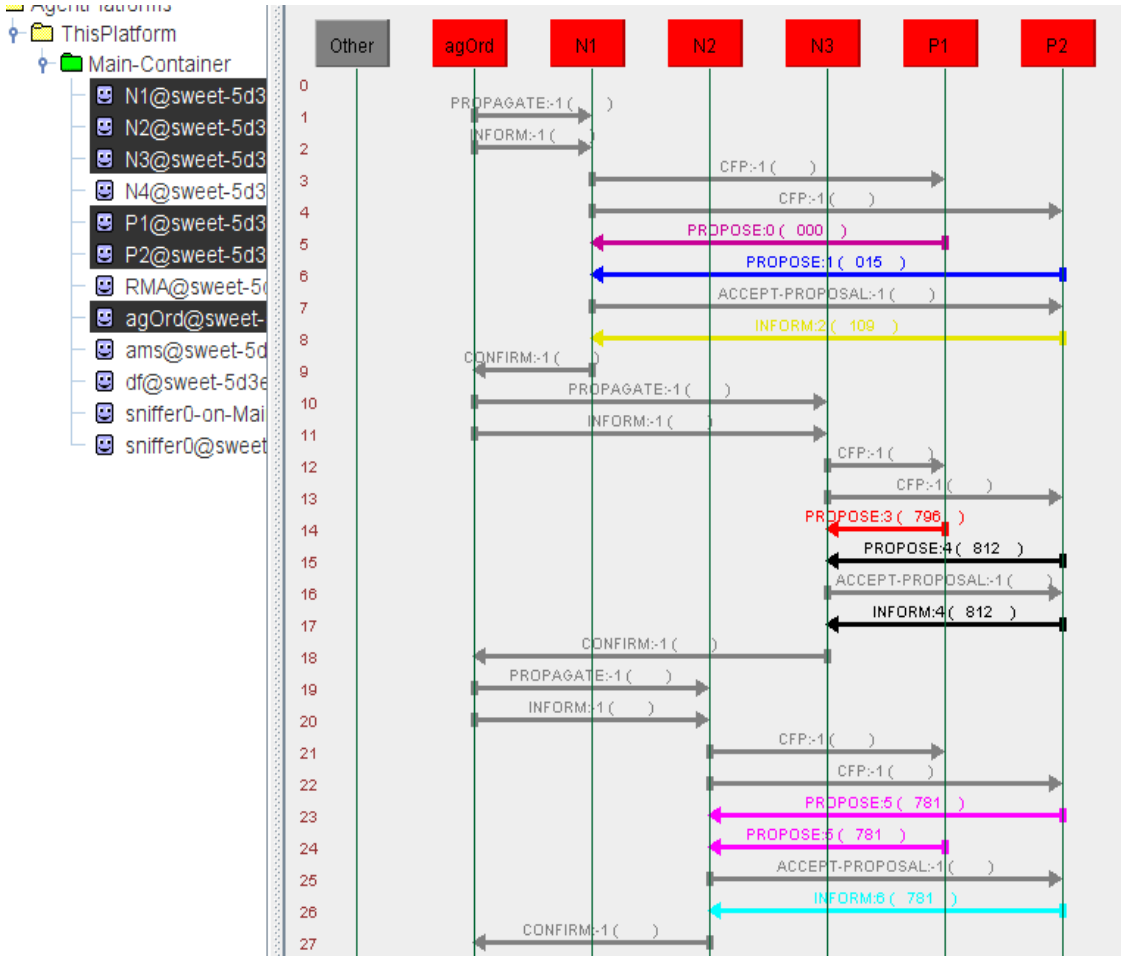


Figure 11. Messages exchanged between the agents of the proposed multi-agent system.

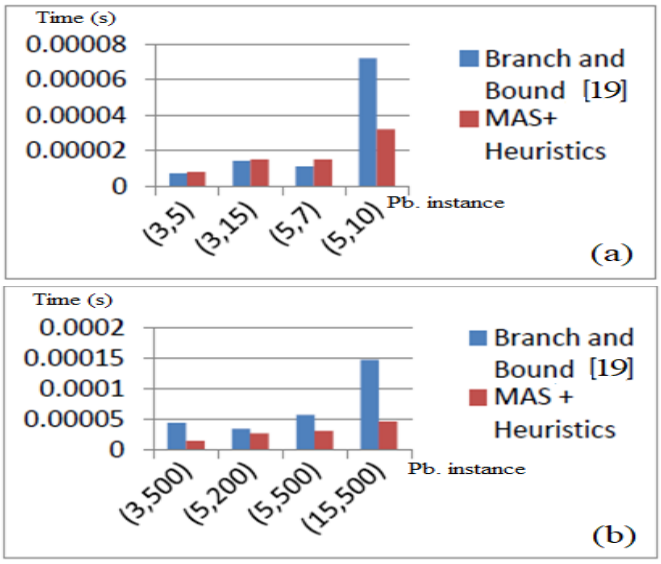


Figure 12. Obtained results; (**a**) for small size instances, (**b**) for big size instances.

Dell'Amico execution time is better than ours only for the small size instances. This constatation is explained by the fact that the authors in [31] use an exact algorithm (branch and bound) that solves the problem faster than our algorithm. Indeed, we use a heuristic algorithm that needs a lot of messages to be exchanged between the agents which generates

an additional execution time for small size problems. However, for large size instances, the results indicate the superiority of our method compared to [31] regarding the execution time. Hence, multi-agent negotiation may be an efficient heuristic method to resolve complex engineering problems.

5.1.2. Obtained Results for Big Size Instances of DellAmico and E. Mokotoff

Table 2 illustrates the results of the same studies for a larger number of instances ($n \leq 200$ and $m \leq 100$). The number of tasks (vessels) is represented by n and M is the number of machines (berths).

Table 2. Results for large number of instances of DellAmico and E. Mokotoff.

		Mokotoff		Dell'Amico		Multi-Agent Approach (Our System)		Genetic Algorithm [33]	
M	***n***	**Time (In Seconds)**	**Non-Resolved Instances**	**Time (In Seconds)**	**Non-Resolved Instances**	**Time (In Seconds)**	**Non-Resolved Instances**	**Time (in Seconds)**	**Non-Resolved Instances**
3	20	0.12	0	0.000014	0	0.000015	0	0.00491	0
3	50	0.10	0	0.000017	0	0.000013	0	0.00582	0
3	100	0.22	0	0.000013	0	0.0000124	0	0.00611	0
3	200	0.12	0	0.000020	0	0.000014	0	0.00638	0
5	20	0.57	0	0.000753	0	0.00042	0	0.00518	0
5	50	0.35	0	0.000021	0	0.000019	0	0.00542	0
5	100	0.67	0	0.000026	0	0.000022	0	0.00566	0
5	200	1.23	0	0.000034	0	0.000027	0	0.00597	0
15	20	0.01	0	0.000004	0	0.000025	0	0.00643	0
15	50	864.47	2	0.000062	0	0.000039	0	0.00681	0
15	100	329.66	0	0.000059	0	0.000037	0	0.00729	0
15	200	348.79	0	0.000104	0	0.000041	0	0.00783	0
100	200	2042.08	0	0.058836	0	0.00078	0	0.00852	0

5.1.3. Comparing with Adriana and Celso and the Genetic Algorithm

In [34], Adriana A. and Celso C.R, they introduce a hybrid tabu search algorithm to solve a problem of the P||Cmax which has, mathematically, the same decision parameters of the berth allocation problem. The aim in both the two problems is to identify a specific set M of tasks (vessels) to assign to specific number n of machines (berths) having a predefined capacity (makespan). As indicated in Table 3, n is set between 10 and 1000 and M is set between 5 and 25. Three intervals of time (Pi:) are used in the experiments (up to one hundred, up to one thousand, and up to ten thousand). Table 3 presents the results after applying different approaches.

Table 3. Comparison of the execution time of our system with genetic and tabu search algorithms.

Pi	**M**	***n***	**Genetic Algorithm**	**Tabu Search (Adriana and Celso)**	**Multi-Agent Approach (Our System)**
1–1000	5	100	0.00985	0.01	0.000022
1–10,000	5	10	0.00368	0.22	0.000036
1–10,000	5	50	0.00492	0.15	0.000019
1–10,000	5	100	0.00643	0.01	0.000022

5.2. Performance Analysis of Genetic Algorithms

The behavior of the genetic algorithm used is assessed by computing the objective value representing the cost of berthing (in terms of handling time) according to the number of generations (Table 4). It is concluded from Table 4 that the performance of the tested genetic algorithm depends on the problem size and the number of generations. It is also noticed that increasing the number of generations to more than 500 has no effect on the quality of the solution due to its convergence.

Table 4. Performance of the genetic objective value (in terms of cost) with different number of generations.

Nbr of Generations		10	30	70	100	150
Population size	20	95.09	95.01	94.39	92.28	89.13
	50	93.42	93.39	93.16	91.16	86.22
	100	81.28	81.12	80.72	78.71	76.16
	200	73.31	73.34	72.28	70.14	68.01
	500	44.46	43.17	41.17	40.28	37.74
	700	42.42	42.19	40.74	38.24	35.97
	1000	39.39	37.21	35.15	33.59	29.45

6. Conclusions

In this study, we introduced the dynamic and discrete problem of berth allocations. We presented a scheduling solution to this problem. Then we suggested a detailed formulation of this problem including the notation, the constraints, the objective function, and the decision variables. Afterwards, we introduced a detailed modeling of the multi-agent approach proposed to solve the DDBAP, using the O-MaSE methodology by providing the corresponding diagrams. Numerical results show the superiority and effectiveness of our approach compared to other paradigms. As a future directive of our study, the aim is to apply our method on other complex engineering problems such as the deployment of objects on Internet of Things networks [35].

Author Contributions: S.M. and M.A. contributed equally to this article. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Giallombardo, G.; Moccia, L.; Salani, M.; Vacca, I. The tactical berth problem with quay crane assignment and transshipment-related quadratic yard costs. In Proceedings of the European Transport Conference (ETC), Noordwijkerhout, The Netherlands, 6 October 2008.
2. Wang, F.; Lim, A. A stochastic beam search for the berth allocation problem. *Decis. Support Syst.* **2007**, *42*, 2186–2196. [CrossRef]
3. Cordeau, J.F.; Laporte, G.; Legato, P.; Moccia, L. Models and Tabu Search Heuristics for the Berth Allocation Problem. *J. Transp. Sci.* **2005**, *39*, 526–538. [CrossRef]
4. Mnasri, S.; Zidi, K. A heuristic approach based on the multi-agents negotiation for the resolution of the DDBAP. In Proceedings of the 4th International Conference on Metaheuristics and Nature Inspired Computing, Sousse, Tunisia, 27–31 October 2012.
5. Nasri, N.; Mnasri, S.; Val, T. 3D node deployment strategies prediction in wireless sensors network. *Int. J. Electron.* **2019**, *107*, 808–838. [CrossRef]
6. Lenstra, J.K.; Kan, R.; Brucker, A.H.G. Complexity of machine scheduling problems. *Ann. Discret. Math.* **1997**, *1*, 343–362.
7. Hassanat, A.B.; Alkasassbeh, M.; Al-Awadi, M.; Alhasanat, E.A. Colour-based lips segmentation method using artificial neural networks. In Proceedings of the 2015 6th International Conference on Information and Communication Systems (ICICS), Amman, Jordan, 7–9 April 2015; pp. 188–193. [CrossRef]
8. Hassanat, A.B.A. Furthest-Pair-Based Decision Trees: Experimental Results on Big Data Classification. *Informations* **2018**, *9*, 284. [CrossRef]
9. Cahyono, R.T.; Flonk, E.J.; Jayawardhana, B. Discrete-Event Systems Modeling and the Model Predictive Allocation Algorithm for Integrated Berth and Quay Crane Allocation. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1321–1331. [CrossRef]
10. Lassoued, R.; Elloumi, A. The discrete and dynamic berth allocation problem in bulk port. In Proceedings of the 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT), Paris, France, 23–26 April 2019; pp. 1976–1980. [CrossRef]
11. Arram, A.; Nazri, M.Z.A.; Ayob, M.; Abunadi, A. Bird mating optimizer for discrete berth allocation problem. In Proceedings of the 2015 International Conference on Electrical Engineering and Informatics (ICEEI), Denpasar, Indonesia, 10–11 August 2015; pp. 450–455. [CrossRef]

12. Schepler, X.; Absi, N.; Feillet, D.; Sanlaville, E. The stochastic discrete berth allocation problem. *EURO J. Transp. Logist.* **2019**, *8*, 363–396. [CrossRef]
13. Iris, Ç.; Lam, J.S.L. Models for continuous berth allocation and quay crane assignment: Computational comparison. In Proceedings of the 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, 10–13 December 2017; pp. 374–378. [CrossRef]
14. Chen, L.; Huang, Y. A dynamic continuous berth allocation method based on genetic algorithm. In Proceedings of the 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE), Beijing, China, 17–19 August 2017; pp. 770–773. [CrossRef]
15. Nourmohammadzadeh, A.; Voß, S. A robust multiobjective model for the integrated berth and quay crane scheduling problem at seaside container terminals. *Ann. Math. Artif. Intell.* **2021**, 1–23. [CrossRef]
16. Kasm, O.A.; Diabat, A.; Cheng, T.C.E. The integrated berth allocation, quay crane assignment and scheduling problem: Mathematical formulations and a case study. *Ann. Oper. Res.* **2020**, *291*, 435–461. [CrossRef]
17. Atencio, F.N.; Casseres, D.M. A comparative analysis of metaheuristics for berth allocation in bulk ports: A real world application. *IFAC PapersOnLine* **2018**, *51*, 1281–1286. [CrossRef]
18. Li, Y.; Chu, F.; Zheng, F.; Liu, M. A Bi-Objective Optimization for Integrated Berth Allocation and Quay Crane Assignment with Preventive Maintenance Activities. *IEEE Trans. Intell. Transp. Syst.* **2020**, 1–18. [CrossRef]
19. Xiang, X.; Liu, C. An expanded robust optimisation approach for the berth allocation problem considering uncertain operation time. *Omega* **2021**, *103*, 102444. [CrossRef]
20. Liu, B.; Li, Z.-C.; Sheng, D.; Wang, Y. Integrated planning of berth allocation and vessel sequencing in a seaport with one-way navigation channel. *Transp. Res. Part B Methodol.* **2021**, *143*, 23–47. [CrossRef]
21. Xiang, X.; Liu, C. An almost robust optimization model for integrated berth allocation and quay crane assignment problem. *Omega* **2021**, *104*, 102455. [CrossRef]
22. Thanos, E.; Toffolo, T.; Santos, H.G.; Vancroonenburg, W.; Berghe, G.V. The tactical berth allocation problem with time-variant specific quay crane assignments. *Comput. Ind. Eng.* **2021**, *155*, 107168. [CrossRef]
23. Guo, L.; Wang, J.; Zheng, J. Berth allocation problem with uncertain vessel handling times considering weather conditions. *Comput. Ind. Eng.* **2021**, *158*, 107417. [CrossRef]
24. Bacalhau, E.T.; Casacio, L.; de Azevedo, A.T. New hybrid genetic algorithms to solve dynamic berth allocation problem. *Expert Syst. Appl.* **2021**, *167*, 114198. [CrossRef]
25. Jiménez, P.; Gómez-Fuster, J.M.; Pavón-Mariño, P. HADES: A Multi-Agent Platform to Reduce Congestion Anchoring Based on Temporal Coordination of Vessel Arrivals—Application to the Multi-Client Liquid Bulk Terminal in the Port of Cartagena (Spain). *Appl. Sci.* **2021**, *11*, 3109. [CrossRef]
26. Dkhil, H.; Diarrassouba, I.; Benmansour, S.; Yassine, A. Modelling and solving a berth allocation problem in an automotive transshipment terminal. *J. Oper. Res. Soc.* **2021**, *72*, 580–593. [CrossRef]
27. Cervellera, C.; Gaggero, M.; Maccio, D. Policy Optimization for Berth Allocation Problems. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021; pp. 1–6. [CrossRef]
28. Guo, W.; Ji, M.; Zhu, H. Multi-Period Coordinated Optimization on Berth Allocation and Yard Assignment in Container Terminals Based on Truck Route. *IEEE Access* **2021**, *9*, 83124–83136. [CrossRef]
29. DeLoach, S.A.; Garcia-Ojeda, J.C. O-MaSE: A customisable approach to designing and building complex, adaptive multi-agent systems. *Int. J. Agent Oriented Softw. Eng.* **2010**, *4*, 244–280. [CrossRef]
30. Official Site of AgentTools AT3. Available online: http://agenttool.cis.ksu.edu/ (accessed on 9 May 2021).
31. Dell'Amico, M.; Martello, S. A note on exact algorithms for the identical parallel machine scheduling problem. *Eur. J. Oper. Res.* **2005**, *160*, 576–578. [CrossRef]
32. Mokotoff, E. An exact algorithm for the identical parallel machine scheduling problem. *Eur. J. Oper. Res.* **2004**, *152*, 758–769. [CrossRef]
33. Akinnuwesi, B.A.; Yama, O.B.; Rahman, A.M.; Fashoto, S.G. Berth Allocation Model for Container Terminal using Genetic Algorithm Technique: Case of Apapa Wharf, Lagos, Nigeria. *J. Comput. Sci. Appl.* **2021**, *28*, 1. [CrossRef]
34. Alvim, A.C.F.; Ribeiro, C.C. A Hybrid Heuristic for Bin-Packing and Multiprocessor Scheduling. *J. Heuristics* **2010**. [CrossRef]
35. Mnasri, S.; Nasri, N.; Val, T. The Deployment in the Wireless Sensor Networks: Methodologies, Recent Works and Applications. In Proceedings of the International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN-2014), Sousse, Tunisia, 4–7 November 2014.