

A survey of task allocation techniques in MAS

George Marios Skaltsis¹, Hyo-Sang Shin¹ and Antonios Tsourdos¹

Abstract—Multi-agent systems and especially unmanned vehicles, are a crucial part of the solution to a lot of real world problems, making essential the improvement of task allocation techniques. In this review, we present the main techniques used for task allocation algorithms, categorising them based on the techniques used, focusing mainly on recent works. We also analyse these methods, focusing mainly on their complexity, optimality and scalability. We also refer to common communication schemes used in task allocation methods, as well as to the role of uncertainty in task allocation. Finally, we compare them based on the above criteria, trying to find gaps in the literature and to propose the most promising ones.

Keywords: task allocation, MAS, auction based, optimisation, learning, game theory, metaheuristics

I. INTRODUCTION

It is common knowledge that most of the systems found in nature, are complex distributed systems. Such systems mainly need to communicate and cooperate in order to achieve a common goal, such as improving the performance of each individual inside a group, aiming to achieve optimal overall performance [1]. Therefore, being inspired from nature, the same principles are used in many complex engineered systems. Especially the last fifteen years, a lot of research efforts have been focused on multi-agent systems that can perform better a lot of tasks that a single agent sometimes was unable to perform. The agent can be a physical entity such as UAVs, UGVs or UUVs, generally types of robots, but even computer resources such as processors, or a computer program [2].

There are a lot of reasons why the scientific community has focused their attention to MAS. Some tasks, especially the distributed ones, may not be able to be performed by a single agent due to their complexity and prerequisites. Moreover, the existence of multiple agents improves the performance and credibility of the executed tasks, since more agents can cooperate to accomplish faster the same task and the system is more robust to agents' losses or malfunctions. Also, the cost might be reduced, since many cheaper and sometimes disposable agents can be used instead of an expensive one [3].

But, while using a multi-agent system for the fulfilment of several tasks, the problem of division of labour arises,

*This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-19-1-7032. Any opinions finding and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

¹ George Marios Skaltsis, Hyo-Sang Shin and Antonios Tsourdos are with the School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield MK43 0AL, UK (email: george.skaltsis, h.shin, a.tsourdos@cranfield.ac.uk)

namely which task will be assigned to which agent, what type of communication will the agents have and generally how the behaviour of each agent will be defined in order to have an optimal and robust performance [3], [4]. The answer to all of these questions are the task allocation techniques. Finding an optimal or near optimal solution to the problem of task allocation in MAS is a quite difficult procedure that has been proven to be NP hard in the general case [5], [6]. Some of the main goals of task allocation, except for achieving the overall optimal system performance, can be the minimization of the execution time of the tasks, the minimization of the time some agents stay inactive, to maximize the number of the tasks completed on a specific amount of time, to maximize the reliability of the task allocation procedure, namely the successful completion of the tasks, etc. [7]. Because the optimal overall performance is a vague concept, that is difficult to be quantified and might depend to the perception of each agent, the concept of utility is used, namely an estimation of the value or cost of the task allocation procedure to the system's performance [4].

Task allocation on its first steps was static, but because the real environments are dynamic environments, the field of dynamic task allocation has become a big field of research the last years. In dynamic task allocation the system can deal with online changes in tasks or the environment, having more robust performance [8]. The algorithms used can be centralised or decentralised, depending on the communication structure of the agents and also homogeneous or heterogeneous agents could be used. In the first applications of task allocation techniques, mainly homogeneous agents were assumed, because of the smaller computational burden of the corresponding algorithms. But, in real world applications, frequently, heterogeneous agents are needed. For example, in robotic systems different types of sensors might exist, or different types of robots might needed for different tasks of the same problem. Even though heterogeneity increases computational cost, its necessity in many applications, has urged researches to develop plenty of task allocation algorithms for heterogeneous MAS [9], [10].

The main techniques used for solving the problem of task allocation in MAS are auction (or market) based approaches, game theory based approaches, optimisation based approaches (heuristic algorithms, metaheuristic algorithms etc.), and machine learning techniques. Depending on the technique used, an optimal, or almost always, an approximate solution can be found and a different degree of scalability, complexity and adaptability of the problem will exist. The tasks or the applications of task allocation in MAS include search and rescue missions (SAR) [11]–[14], military op-

erations such as attack or surveillance [15]–[18], physical disasters management [11], [12], [19]–[22] where unmanned systems are mainly used, but also crowd-sourcing platforms usage, cloud computing [23]–[28], smart grids, resources allocation in manufacturing [29]–[32] and others.

II. DIFFERENT COMMUNICATION SCHEMES OF TASK ALLOCATION TECHNIQUES

There are two main categories of algorithms for solving task allocation problems, the centralised and the decentralised algorithms.

A. Centralised task allocation

The centralised algorithms are a category of algorithms that have been researched a lot in the past. The main concept is that there is a central coordinator agent, that has communication channels with all the other agents. This agent manages the negotiations of agents, if existed and decides about the tasks that have to be allocated to the other agents. In these cases, most of the times, there is a global utility function taken into consideration [14], [33], [3], [34].

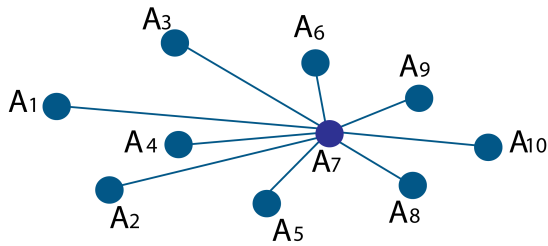


Fig. 1. A centralised system, with agent A_7 being the central coordinator

The advantages of these methods are that they use less system resources and might have lower cost of implementation, but they can be used on a small amount of agents, due to their high computational costs and also they are not adjustable to dynamic environments, therefore they are mainly used for static task allocation. The fact that the tasks are allocated centrally avoids conflicts on task assignments, therefore a consensus stage is not needed and also an optimal solution to the allocation problem can be found. They also lack in robustness, since they are vulnerable to losses of agents and especially the central agent, resulting to the deterioration of the overall performance. Moreover, the fact that all the agents communicate with the central one, limits their scalability [17], [35].

B. Decentralised task allocation

The decentralised algorithms overcome some of the drawbacks of the centralised algorithms, therefore they have attracted the attention of researchers the last few years. In this type of algorithms there is no central coordinator agent, the agents have a local perception of the environment and might negotiate with each other instead of a central agent. As a result, the decision for the task allocation is taken locally, in a distributed way. Every agent, also, might have

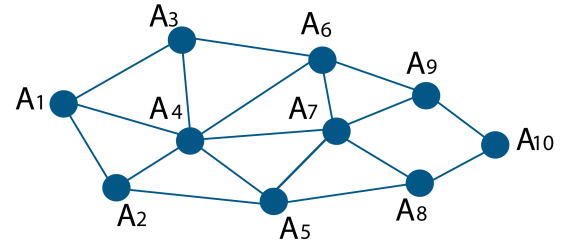


Fig. 2. A decentralised system

its own utility function and an overall utility function might be approximated [14], [33], [3], [34].

The advantages of these methods are that they are robust techniques, since agents' failures have small impact to the overall performance and that are also scalable, due to the low level of communication between the agents. Moreover, they have smaller computational cost than the centralised methods making them ideal for large scale systems, even with small communication bandwidth. The trade-off is that they find a suboptimal (approximate) solution of the task allocation problem and also a consensus algorithm might be necessary, because the local task assignment can cause conflicts between assigned tasks [17], [35].

III. DIFFERENT TYPES OF ALGORITHMS IN TASK ALLOCATION OF MAS

There are a lot of techniques used for task allocation in MAS. Below follows a categorisation of methods used, presenting some main aspects of these methods (see Fig. 3).

A. Auction Based Algorithms

A very big category of algorithms used for task allocation in MAS is the auction based algorithms. This type of algorithms is based on economics and the agents use a negotiation protocol to bid in an auction for tasks, based on the local perception of the environment that they have. This is the reason why sometimes these approaches are also called market based. The agents bid according to the utility or cost they calculate and their goal is to accomplish the highest utility or the lowest cost for the task allocated. A global objective function is optimised, based on the utility functions of the agents. The auctioneer might be a central agent or the auction might be done in a distributed way by the agents of the system and the auctions, that might take several rounds, can regard one or several tasks [10], [14], [36], [37].

The auction based algorithms have a lot of advantages, such as high solution efficiency, even though they find sub-optimal solutions, since they use aspects of both centralised and decentralised methods and robustness. They also are scalable because they have moderate computational cost or communication burden, as not fully centralised algorithms and they are good for dynamic task allocation, since they can add or remove new tasks from the auction procedure [3].

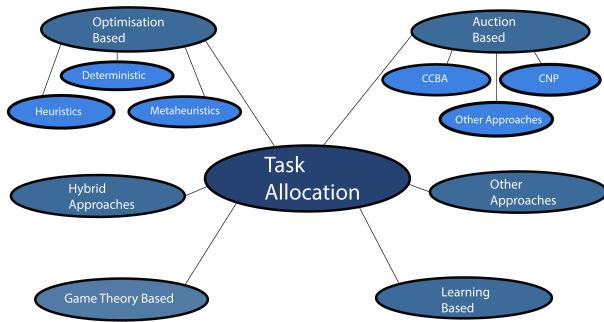


Fig. 3. Task allocation techniques categories

a) *CBBA based algorithms*: The consensus based bundle algorithm (CBBA) is a decentralised algorithm that provides solutions, independent of inconsistencies to the agents' situational awareness, to multi-objective optimization problems, having as cost function the utility each agent perceives for performing bundles of tasks. In the first stage the algorithm uses auctions with greedy heuristics to select the tasks and on the second stage the algorithm applies a consensus based procedure to unravel any overlapping tasks that have occurred. The algorithm is proved to provide suboptimal solutions for the single robot single task task allocation problem (see [4] for a complete taxonomy) and is highly scalable making it suitable for dynamic task allocation applications, since it has polynomial time bidding [38] [39].

The recent approaches found include improvements of the PI (performance impact) algorithm, like PI-MaxAss [14] and [35]. Moreover, other techniques are improvements of the CBBA algorithm, like modified CCBBA [38], G-CBBA [40] and [41].

b) *CNP based techniques*: The Contract Net Protocol (CNP) developed by Smith [42] was the first negotiation platform used in task allocation problems and constitutes the base for numerous task allocation algorithms. It is a standardised protocol, that can allocate the tasks to the most appropriate agents and at the same time it is capable of task reassignment where needed [43]. On the other side, CNP has the message congestion problem, incommoding sometimes the negotiation procedure between the agents. Unlike other approaches, such as pheromone based approaches, CNP depends considerably to the communication by messages between agents and the computational cost of these messages can be very high, deteriorating the communication efficiency and the systems performance [44].

Some recent CNP based approaches include [45], [46], [11], [27], [44]. Moreover, an auction based approach that does not belong to the aforementioned categories is the (FMC-TA) [47].

B. Game theory based

In game theory based approaches, the agents are assumed to be players that take specific actions and the task allocation scheme is the strategy that they should follow. The reward that the players take, depending on their actions, at the end of the game is called the payoff. When the players have

chosen the best strategy, then they will not wish to change their strategy, because this is the best outcome they could accomplish, reaching a condition called Nash equilibrium [48].

Games can be divided into two main categories, the cooperative and the non-cooperative games. In the cooperative games the agents are cooperating or forming coalitions before taking their specific actions, affecting their general strategy and utilities. One example of cooperative games is the coalition formation game. In non cooperative games, agents choose their actions and their strategy individually, meaning that agents are selfish and want to reach the highest payoff. Some examples include Bayesian games, non-cooperative differential games, sub-modular games, etc. [49].

Some recent game theory based approaches include [50], [20], [51], [52], [53], [54], [55].

C. Optimisation based techniques

Optimisation is the field of applied mathematics aiming to find a solution to a specific problem, from a set of possible solutions, minimising the cost or maximising the profit, of a certain cost or objective function. This cost function which is optimised according to some constraints, determines the aim of the system. There are a lot of optimisation techniques that can be either deterministic or stochastic [3], [56]. Deterministic methods do not consider randomness, meaning that the path to the solution will be the same, if the same starting point is used. Deterministic methods include techniques such as graphical methods, graph based methods, sequential programming, linear programming, mixed integer linear programming (MILP), etc. Stochastic methods or metaheuristics are methods that include a randomness to the calculations. Metaheuristics include evolutionary algorithms, swarm intelligence, simulated annealing, etc. Moreover, heuristic algorithms are algorithms that are used to find fast and quality solutions to difficult optimisation problems that a deterministic method would have unbearable computational cost. These methods provide approximate solutions though [57].

a) *Deterministic optimisation based*: One optimisation algorithm that is frequently used as the base for developing new task allocation algorithms is the Hungarian algorithm [58]. The Hungarian algorithm, treats the problem of task allocation as a combinatorial optimisation problem, using graph theory and solves the problem in polynomial time. The algorithm computes an estimate of each agents utility, thus maximising the overall utility. But this is computationally expensive and sometimes of lower value when high uncertainties are present to the system, therefore a lot of improvements to the algorithm have been proposed [59]. Some recent approaches include [60], [61] and [62].

b) *Metaheuristics*: Metaheuristics include several methods like swarm intelligence, genetic algorithms, simulated annealing and others. Swarm intelligence has been widely used in task allocation of MAS and is a category of biologically inspired algorithms, mainly from animals with social behaviour, such as insect colonies, school of fish, flocks of

birds, etc. [63]. These animals demonstrate efficient division of labour, due to the specialisation of the members of a team, leading to colony efficiency [64]. Even though the agents might be quite simple, they can accomplish complex tasks as a whole due to their cooperation, leading to robust, efficient and low cost solutions [65]. On the other side, these algorithms some times assign unnecessary tasks to the agents, cause conflicts and have slow global response to environment variations [63]. Mainly the methods used are divided between threshold based and probabilistic methods.

In the threshold based methods, such as response threshold method [66], the agents decide their actions regarding the tasks depending on the values that take some monitored quantities and the value of the threshold. The threshold can be fixed or variable and the agents might have only local or global information about that quantity. In probabilistic methods, the agents change task randomly, based on probabilities calculated with environmental observations or historic data. Also, a stimulus might be used and a task might be chosen when the stimulus is high for the specific task [67].

Some recent metaheuristic based methods for task allocation include the modified distributed bees algorithm [63], dynamic ant colony's labor division [17], distributed immune multi-agent algorithm [68], improved QPSO [69], Hierarchical Task Assignment and Path Finding method [70], Multi-Objective Multi-Humanoid robots Task Allocation [71] and other techniques like [72], [73], [15].

c) Heuristics: Recent heuristic based approaches include Lazy max-sum algorithm [19], average Hamiltonian partition - multiple traveling salesperson algorithm [74], One-To-Many Bipartite Matching [75], nearest-neighbor based Clustering And Routing approach [76] and [77].

D. Learning based approaches

It is very difficult to predict the future disturbances that an agent might have to deal with, especially where there are not specific mathematical models that describe the behaviour of the environment, which is dynamic for real applications. Therefore, one solution is the agents to learn to confront such disturbances, taking into consideration their past actions and the actions of other agents, leading to higher system efficiency [78], [79], [80].

A typically used machine learning technique is reinforcement learning, where the agents use their experience to learn how to act in different states of the environment. The environment is, usually, formed as a Markov Decision Process (MDP) and the agents optimize a cost or reward function in order to learn from the environment. A frequently used RL method is Q-learning, which is a model free RL method and help the agents find optimal solutions in MDPs. [78], [79]. RL has a lot of advantages, including handling uncertainties in the environment, real time implementation (for well trained networks) and dealing with different tasks [16]. On the other side, especially in large scale complex systems, most RL algorithms require high computational power [81].

The learning based approaches found include the decentralised self-organising map based approach in [82], the stochastic reinforcement learning based algorithm in [12], the graph based multi-agent reinforcement learning method [83], a MARL with enhanced hill climbing search approach [84], a Q-learning based fast task allocation algorithm [16], the Task Allocation Process using Cooperative Deep Reinforcement Learning strategy [79] and a MARL soft Q - learning based method [85].

E. Hybrid approaches

Except for the above approaches for solving the problem of task allocation there are also some other approaches that combine some of the above methods and they are called hybrid approaches.

In [86] an optimisation and an auction based approach are combined, while in [87] a market based method is combined with a game theory based one. Furthermore, [88], [89] and [13] are a market based and metaheuristic combination and [90] is a market based and learning combination. In [91] an evolutionary algorithm with a greedy algorithm are combined, while in [92] a game theoretic based approach is combined with a learning algorithm.

IV. KEY FACTORS OF THE TASK ALLOCATION PROCEDURE

Some basic criteria for the evaluation of the task allocation procedure in MAS are the computational complexity of the algorithms used, the optimality of the solutions and the scalability of the approach used. Moreover, the capability of the algorithms to handle uncertainties, as well as the effectiveness of the communication procedure, play a significant role to the overall system performance.

A. Complexity, optimality and scalability

The factors that affect the computational cost of task allocation are the complexity of the algorithm used, the frequency that these algorithms are used and also the computational cost of the communication method needed between the agents (the bits of information that the agents need to exchange for achieving successful task allocation) [93], [94].

Another key factor is the optimality of the solutions found. When we refer to the optimality of the task allocation procedure, we mean that the solution found has the highest overall utility possible, constrained by the systems' characteristics, like noise, uncertainty and inaccuracy of the information provided to the agents. The frequency that an algorithm is executed in order to find dynamic and not static solutions as well as the proportion of the tasks that can be reassigned, affect the solution quality [4]. Moreover, as more and more complex tasks and higher number of agents are employed to the task allocation schemes, scalability of the algorithms is crucial to their effectiveness.

a) CBBA based: The CBBA based approaches presented, that are improvements of the CBBA and PI algorithms, demonstrate better efficiency and scalability than the baseline CBBA method, but with the disadvantage of

TABLE I
COMPLEXITY OF SOME REPRESENTATIVE TASK ALLOCATION ALGORITHMS

Category	Algorithm	Complexity
CBBA based	PI-MaxAss [14]	$\mathcal{O}(n_t^3)$
	modified CCBBA algorithm [38]	$\mathcal{O}(\Theta M I_{max} n_t 2^{I_{max}} (2^{I_{max}} + M^2) + n_a n_t I_{max} 2^{I_{max}})$
Game theory based	Apollonius circle-based Active Pursuer Check (AAPC) [52]	$\mathcal{O}(n_a^2)$
	GRAPE [50]	$\mathcal{O}(d_G n_t n_a^2)$
Metaheuristics	MO-MHTA [71]	$\mathcal{O}(n_t K + 3n_t^2 + \text{Max}(H^2 \log^{L-2} H, H^2, L))$
Heuristics	lazy max-sum [19]	$\mathcal{O}(n_a \log n_a)$
	AHP-mTSP [74]	$\mathcal{O}(n_a^2)$
	nCAR [76]	$\mathcal{O}(n_a^3)$
	OTMaM [75]	$\mathcal{O}(n_a n_t)$
Hybrid	simplified MILP with iterative scheduling [86]	$\mathcal{O}(n_a^3)$
	CBBA based with Ant Colony System (ACS) and greedy [89]	$\mathcal{O}(n_t^3)$

higher computational cost. Indicatively, the computational complexity of the algorithm PI-MaxAss [14] is equivalent to $\mathcal{O}(n_t^3)$, where n_t is the number of tasks. Moreover the complexity of the modified CCBBA algorithm [38] is $\mathcal{O}(\Theta M I_{max} n_t 2^{I_{max}} (2^{I_{max}} + M^2) + n_a n_t I_{max} 2^{I_{max}})$, where Θ is the maximum number of iterations needed before convergence, I_{max} is the maximum number of sensors per task, n_a is the number of agents, n_t is the number of tasks and M is the planning horizon.

b) CNP based: Generally the CNP based techniques are very good in reallocating tasks, but are highly dependent to the communication procedure between agents, causing often high computational cost. Also another problem of CNP is the observed message congestion. The improved CNP algorithms presented, have higher efficiency and smaller computational cost than baseline CNP. But, even though there are some approaches that try to confront the message congestion problem, e.g. [44], this is still an open field in research.

c) Game theory based: The game theory approaches presented, are more efficient than baseline approaches, with better suboptimal (near optimal) solutions. Moreover, some game theoretic algorithms have better efficiency than market based approaches. As for the complexity, the Apollonius circle-based Active Pursuer Check (AAPC) [52], has complexity of $\mathcal{O}(n_a^2)$, where n_a is the number of pursuers. The complexity of GRAPE algorithm, based on an anonymous hedonic game [50], is bounded by $\mathcal{O}(d_G n_t n_a^2)$, even though in most of the cases is much less, where d_G is the graph diameter of the network, n_t is the number of tasks and n_a is the number of agents. As for the communication complexity of each agent it is $\mathcal{O}(|N_i| n_a)$, where $|N_i|$ is the number of agents that the agent i communicates.

d) Heuristics: There are a lot of techniques for solving DCOP problems. The techniques providing optimal solutions usually have an exponential coordination burden and heuristic based techniques have lower coordination cost, but provide suboptimal solutions. Some of the proposed techniques demonstrated higher efficiency and smaller computational cost than some genetic and market based approaches [19]. The lazy max-sum approach [19] has a message passing

complexity of $\mathcal{O}(n_a \log n_a)$, but if we consider all agents to all tasks assignments, the complexity rise to $\mathcal{O}(n_t^{n_a})$. For the AHP-mTSP algorithm [74] (average Hamiltonian partition, multiple traveling salesperson problem), which finds suboptimal solutions, every iteration has complexity of $\mathcal{O}(n_a^2)$ and an average run time of $n_a^{2.11} n_t^{0.33}$, for n_a agents and n_t tasks. Moreover, the centralised heuristic nearest-neighbour based Clustering And Routing (nCAR) approach [76] has computational cost of $\mathcal{O}(n_a^3)$, where n_a is the number of agents. The OTMaM technique [75], which is suitable for large scale systems, has a time complexity of $\mathcal{O}(n_a n_t)$, where n_a is the number of agents and n_t is the number of tasks.

e) Metaheuristics: The metaheuristics techniques are low cost, robust and efficient, but sometimes can cause conflicts between tasks, allocate unnecessary tasks to agents and have slow response to environmental variations. The presented algorithms have lower complexity and improved scalability compared to baseline ones. But, some of them where suboptimal or assumed no failures to the communication procedure. Moreover, some of the algorithms have higher scalability and better performance than some greedy and market based (e.g. CNP) approaches. For the MO-MHTA algorithm [71] the overall worst-case complexity is $\mathcal{O}(n_t K + 3n_t^2 + \text{Max}(H^2 \log^{L-2} H, H^2, L))$, where n_t is the number of tasks, H is the number of reference points on the hyperplane, L is the number of the objectives and K is the number of clusters created.

f) Learning based: The learning based approaches and especially the reinforcement learning approaches, generally have high efficiency, might be online implementable and have good behaviour to environmental disturbances. We noticed that a lot techniques have better performance than baseline simulated annealing, hill climbing and greedy algorithms. Moreover, it was noticed higher efficiency than Frontier based and the Hungarian method. Even though some methods had smaller computational cost than auction based methods, the computational cost and the increase in dimensionality was still a problem in other reinforcement learning methods.

TABLE II
COMMUNICATION TYPE OF SOME REPRESENTATIVE TASK ALLOCATION ALGORITHMS

Category	Algorithm	Communication
CBBA based	Extended PI [35]	Mesh, row and hybrid (row-tree) communication schemes were used and had similar performance for extended PI. The row communication scheme had the higher percentage of problems solved with the best solution, followed by hybrid and mesh approaches.
	PI-MaxAss [14]	No communication cost assumed. The agents communicate once in every iteration
	modified CCBBA [38]	The number of algorithm iterations in order to achieve convergence is proportional to the communication throughput (number of messages exchanged). In this allocation scheme it is assumed that all agents can communicate with each other for achieving consensus and with trivial communication cost. This is a simplification, since communication throughput cost in decentralised market based task allocation algorithms can be high.
	Cluster first strategy CBBA [41]	Better solution quality with both mesh communication, row communication or circular communication scheme. Higher convergence rate with mesh scheme, decreasing with limited communication.
	MDMA-CCM with SAS [45]	Improved CNP communication scheme.
Other auction based	FMC-TA [47]	Each agent to whom a task has been allocated, communicates with the other agents with whom it shares the tasks.
Game theory based	GRAPE [50]	Assumes strongly connected communication network.
	MOCFF [51]	A social network is used for the communication procedure. The agents communicate with their neighbours (defined by a fixed range) and through them with all the other agents using the same procedure.
Metaheuristics	DIMAA [68]	The blackboard communication scheme is used, namely the location, target point and damage information of each agent will be added to a shared file of the blackboard with other relevant information. The agents communicate with neighbouring agents only.
	HTAPF [95]	The agents communicate with each other by using a probabilistic communication model (broadcast communication), without re-broadcasting.
Learning	Stochastic RL cellular learning automata [12]	A hybrid communication method is used (explicit and implicit).
Hybrid	Potential games TVCBLL [92]	Each agent communicates with other agents inside a communication radius and receives information about tasks inside this range, therefore chooses tasks only from this area.
	MSMA [87]	A social network is used for communication, where the agents communicate with their neighbours (in a specific range) and through them with the rest agents. The structure of the network changes, due to the agents' movements
	Auction based and pheromone map [13]	The communication was based on the Pheromone Map Model, where agents set virtual markers referring to mission and network states, that are 'sensed' from the other agents. This approach reduces direct agent communication.

g) Hybrid: The use of Hybrid approaches is a very good solution, since two techniques can be combined, exploiting their advantages and achieving higher efficiency or smaller computational cost than baseline methods or than using one method only. In [86], where a simplified MILP program and an iterative scheduling algorithm with a multi-agent bidding is used, the computational complexity of the iterative scheduler is $\mathcal{O}(n_a^3)$, where n_a is the subset of agents. Moreover, in the lower levels of this scheduler a GSTP algorithm is used, increasing more the overall complexity. In [89], where a CBBA based approach, is combined with the Ant Colony System (ACS) algorithm and a greedy based strategy is used for the inclusion phase of the CBBA, the worst case computational complexity is $\mathcal{O}(n_t^3)$, where n_t is the number of survivors (tasks).

In table I a summary of the complexity of the aforementioned algorithms is presented. As we can see most of the methods have polynomial time complexity. The higher computational cost have the CBBA based algorithms, as well

as some hybrid approaches. On the other side, the heuristic based approaches and the game theory based have the less complexity.

B. Communication

The communication between agents is a very important factor for the performance of their coordination. The goal is the agents to exchange important information regarding their state, as well as the environment surrounding them, using the smallest amount of bandwidth available and without overloading the communication network [12]. The communication of the agents can be explicit or implicit. Explicit or direct communication, is the exchange of messages between the agents, using a communication network and dedicated network protocols. Most of the existed coordination methods use this type of communication. The implicit method refers to getting information about the other agents of a multiagent system through the environment, using sensors that the agents are equipped with. Implicit communication is active if

TABLE III
COMPARISON OF THE MAIN TASK ALLOCATION METHODS

Algorithm category	Efficiency	Scalability	DTA	Computational cost
CBBA Based	★★★	★★	★★	★★★★
CNP Based	★★	★★★	★★	★★★★
Game Theory	★★★	★★★	★★★	★★
Deterministic Optimisation	★★★★	★	★★	★★★★
Heuristics	★★★	★★★	★★	★
Metaheuristics	★★★	★★★	★★★	★★★
Learning	★★★★	★★★	★★★★	★★★

the agents communicate using the information other agents leave in the environment (biology inspired techniques) and passive if the agents use their sensors to perceive the changes happening to their environment [96].

The explicit communication style has generally higher accuracy than the implicit case, with the disadvantage of higher communication load, especially for larger scale systems. The implicit case, even though lacks in accuracy, has better stability and is more fault tolerant. Therefore a mixture of these methods is a very good idea for exploiting their advantages, leading to better overall system performance [96]. In table II are presented the communication techniques of some characteristic algorithms for task allocation. As we see, some frequently used techniques are the social network technique, the blackboard scheme, the pheromone map and generally graph based techniques.

C. Uncertainty

Task allocation techniques that take into consideration uncertainty, are very useful for the implementation of highly efficient and robust task allocation in real life applications. Most of the techniques so far, especially the distributed ones, that is more difficult, than the centralised ones, to incorporate uncertainty, do simplifying assumptions about the environment. Uncertainty can regard sensor inaccuracies, agents' failures, environmental disturbances etc. [97] [98]. According to previous research, reliability should be taken into consideration a priori, because by neglecting the possibility of failure, the performance decreases (suboptimal performance) [99]. For example, in [100] the authors found that the usage of the Asynchronous Consensus Based Bundle Algorithm (ACBBA) in environments with uncertainty in the communication procedure (realistic lossy network environment), creates inefficient task assignments, especially for a large number of agents. Therefore the algorithms' performance varies compared to the theoretically expected performance.

In [99] is studied the problem of uncertainty (generally failure of elements of the task allocation procedure) in multi-agent systems, using a heuristic approach and non Markovian states. Their conclusion is that making simplifying assumptions such as Markovian states can lead to results that are not a fair representation of the systems' performance.

Moreover, they proved that in some categories of problems, the usage of more sophisticated heuristics that describe better the physical environment and the uncertainties occurring, led to an increase in performance. In [97] the authors developed an improved version of the performance impact (PI) algorithm with improved robustness, by dealing with uncertain environments. Three robust PI variants are proposed that use Monte Carlo sampling to sample uncertain variables from a Gaussian distribution. The proposed methods decrease the failure rate under uncertainties and the number of unallocated tasks, compared to baseline CBBA and PI, but increase the computational complexity making them unreliable for time-critical applications.

Therefore, incorporating uncertainty can be very useful in a lot of applications leading to better performance. But always there is the danger of the higher computational complexity, therefore always there should be a balance between efficiency, robustness and the convergence time, depending on the computational power available and the specific needs of every application.

V. CONCLUSION

In table III there is a summary of some main performance characteristics of the main task allocation techniques, categorised in a scale from one (low value) to four (very high value). We see that the CBBA and CNP based techniques have generally high computational cost, making them inappropriate for large scale systems. Moreover deterministic optimization techniques have also extremely high cost and low scalability, making them unsuitable too for medium to large scale systems, even though they have very good efficiency. On the other side, heuristic and game theory approaches have very low cost, making them ideal for providing fast solutions with moderate to good degree of efficiency. These approaches can also be used in large scale systems, since they have very good scalability. Metaheuristics and learning approaches have moderate cost, good efficiency and scalability and can be used in medium scale and some times in larger scale environments, depending on the specific problem. Especially learning techniques are very good in dynamic task allocation and dynamic environments.

As technology of MAS systems evolves and the computational power is increasing every year, the need for

implementation of improved task allocation algorithms in real environments is imperative. Such environments have high uncertainties, complex tasks and might require real time implementation of the algorithms used. Because of the adaptability to such environments, RL methods are a promising field of research in task allocation, that is widely researched by the scientific community the last few years. Moreover, game theory and metaheuristic approaches are also promising for such systems. As noted in [101] the combination of RL and game theory based techniques improves RL in the multiagent case (MARL), therefore the combination of game theory and RL based techniques is very promising for task allocation methods as well.

REFERENCES

- [1] D. Charbonneau and A. Dornhaus, "When doing nothing is something. How task allocation strategies compromise between flexibility, efficiency, and inactive agents," *Journal of Bioeconomics*, vol. 17, no. 3, pp. 217–242, 2015.
- [2] Y. Rizk, M. Awad, and E. W. Tunstel, "Cooperative heterogeneous multi-robot systems: A survey," *ACM Computing Surveys*, vol. 52, no. 2, 2019.
- [3] A. Khamis, A. Hussein, and A. Elmogy, "Multi-robot Task Allocation: A Review of the State-of-the-Art," in *Studies in Computational Intelligence*, 2015, vol. 604, pp. 31–51. [Online]. Available: http://link.springer.com/10.1007/978-3-319-18299-5_2
- [4] B. P. Gerkey and M. J. Matarić, "A formal analysis and taxonomy of task allocation in multi-robot systems," *International Journal of Robotics Research*, vol. 23, no. 9, pp. 939–954, 2004.
- [5] S. Kartik and C. Siva Ram Murthy, "Task allocation algorithms for maximizing reliability of distributed computing systems," *IEEE Transactions on Computers*, vol. 46, no. 6, pp. 719–724, jun 1997. [Online]. Available: <http://ieeexplore.ieee.org/document/600888/>
- [6] A. Cornejo, A. Dornhaus, N. Lynch, and R. Nagpal, "Task allocation in ant colonies," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8784, 2014, pp. 46–60.
- [7] Y. Jiang, "A Survey of Task Allocation and Load Balancing in Distributed Systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 585–599, 2016.
- [8] N. Excelente-Toledo, C.B., Jennings, "The Dynamic Selection of Coordination Mechanisms," *Autonomous Agents and Multi-Agent Systems*, vol. 9, pp. 55–85, 2004.
- [9] L. E. Parker, "THE EFFECT OF HETEROGENEITY IN TEAMS OF 100+ MOBILE ROBOTS," Tech. Rep., 2003. [Online]. Available: http://web.eecs.utk.edu/~leparker/publications/NRL03_Parker.pdf
- [10] —, "Multiple Mobile Robot Systems," in *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 921–941. [Online]. Available: http://link.springer.com/10.1007/978-3-540-30301-5_41
- [11] N. Hooshangi and A. A. Alesheikh, "Developing an agent-based simulation system for post-earthquake operations in uncertainty conditions: A proposed method for collaboration among agents," *ISPRS International Journal of Geo-Information*, vol. 7, no. 1, 2018.
- [12] M. Khani, A. Ahmadi, and H. Hajary, "Distributed task allocation in multi-agent environments using cellular learning automata," *Soft Computing*, vol. 23, no. 4, pp. 1199–1218, 2019.
- [13] R. S. de Moraes and E. P. de Freitas, "Distributed Control for Groups of Unmanned Aerial Vehicles Performing Surveillance Missions and Providing Relay Communication Network Services," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 92, no. 3-4, pp. 645–656, 2018.
- [14] J. Turner, Q. Meng, G. Schaefer, A. Whitbrook, and A. Soltoggio, "Distributed Task Rescheduling with Time Constraints for the Optimization of Total Task Allocations in a Multirobot System," *IEEE Transactions on Cybernetics*, vol. 48, no. 9, pp. 2583–2597, 2018.
- [15] J. Schwarzkrock, I. Zacarias, A. L. Bazzan, R. Q. de Araujo Fernandes, L. H. Moreira, and E. P. de Freitas, "Solving task allocation problem in multi Unmanned Aerial Vehicles systems using Swarm intelligence," *Engineering Applications of Artificial Intelligence*, vol. 72, no. September 2017, pp. 10–20, 2018.
- [16] X. Zhao, Q. Zong, B. Tian, B. Zhang, and M. You, "Fast task allocation for heterogeneous unmanned aerial vehicles through reinforcement learning," *Aerospace Science and Technology*, vol. 92, pp. 588–594, 2019. [Online]. Available: <https://doi.org/10.1016/j.ast.2019.06.024>
- [17] H. Wu, H. Li, R. Xiao, and J. Liu, "Modeling and simulation of dynamic ant colony's labor division for task allocation of UAV swarm," *Physica A: Statistical Mechanics and its Applications*, vol. 491, pp. 127–141, 2018. [Online]. Available: <http://dx.doi.org/10.1016/j.physa.2017.08.094>
- [18] H. Liang, H. Qiang, and Z. Feng, "Research on Capability Characteristics Modeling and Cooperative Fire Strike Planning for Unmanned Ground Vehicles," in *2019 2nd International Conference on Artificial Intelligence and Big Data, ICAIBD 2019*. IEEE, 2019, pp. 136–141.
- [19] J. Parker, A. Farinelli, and M. Gini, "Lazy max-sum for allocation of tasks with growing costs," *Robotics and Autonomous Systems*, vol. 110, pp. 44–56, 2018. [Online]. Available: <https://doi.org/10.1016/j.robot.2018.08.015>
- [20] M. C. Lim and H. L. Choi, "Improving computational efficiency in crowded task allocation games with coupled constraints," *Applied Sciences (Switzerland)*, vol. 9, no. 10, 2019.
- [21] T. L. Baggio and R. H. Bordini, "Allocating structured tasks in heterogeneous agent teams," *Computational Intelligence*, vol. 35, no. 1, pp. 124–155, 2019.
- [22] P. Ghassemi, D. DePauw, and S. Chowdhury, "Decentralized Dynamic Task Allocation in Swarm Robot Systems for Disaster Response," *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pp. 83–85, 2019. [Online]. Available: <http://arxiv.org/abs/1907.04394>
- [23] A. Chatterjee, M. Borokhov, L. R. Varshney, and S. Vishwanath, "Efficient and Flexible Crowdsourcing of Specialized Tasks with Precedence Constraints," *IEEE/ACM Transactions on Networking*, vol. 26, no. 2, pp. 879–892, 2018.
- [24] A. Jyoti and M. Shrimali, "Dynamic provisioning of resources based on load balancing and service broker policy in cloud computing," *Cluster Computing*, vol. 23, no. 1, pp. 377–395, 2019. [Online]. Available: <https://doi.org/10.1007/s10586-019-02928-y>
- [25] Y. Cheng and G. Xu, "A Novel Task Provisioning Approach Fusing Reinforcement Learning for Big Data," *IEEE Access*, vol. 7, pp. 143 699–143 709, 2019.
- [26] W. Housseini, O. Mosbahi, M. Khalgui, Z. Li, L. Yin, and M. Chetto, "Multiagent Architecture for Distributed Adaptive Scheduling of Reconfigurable Real-Time Tasks With Energy Harvesting Constraints," *IEEE Access*, vol. 6, pp. 2068–2084, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8171204/>
- [27] Q. Baert, A. C. Caron, M. Morge, and J. C. Routier, "Fair multi-agent task allocation for large datasets analysis," *Knowledge and Information Systems*, vol. 54, no. 3, pp. 591–615, 2018.
- [28] Y. Kong, M. Zhang, and D. Ye, "A belief propagation-based method for task allocation in open and dynamic cloud environments," *Knowledge-Based Systems*, vol. 115, pp. 123–132, 2017. [Online]. Available: <http://dx.doi.org/10.1016/j.knosys.2016.10.016>
- [29] R. Khanizad and G. Montazer, "Optimal allocation of human resources based on operational performance of organizational units using fuzzy game theory," *Cogent Engineering*, vol. 5, no. 1, pp. 1–20, 2018. [Online]. Available: <http://doi.org/10.1080/23311916.2018.1466382>
- [30] N. Yussupova and D. Rizvanov, "Decision-Making Support in Resource Management in Manufacturing Scheduling," *IFAC-PapersOnLine*, vol. 51, no. 30, pp. 544–547, 2018. [Online]. Available: <https://doi.org/10.1016/j.ifacol.2018.11.269>
- [31] J. Hanke, O. Kosak, A. Schiendorfer, and W. Reif, "Self-organized resource allocation for reconfigurable robot ensembles," in *International Conference on Self-Adaptive and Self-Organizing Systems, SASO*, vol. 2018-Sept. IEEE, 2019, pp. 110–119.
- [32] J. Wan, B. Chen, S. Wang, M. Xia, D. Li, and C. Liu, "Fog Computing for Energy-Aware Load Balancing and Scheduling in Smart Factory," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4548–4556, 2018.
- [33] K. Zhang, E. G. Collins, and D. Shi, "Centralized and distributed task allocation in multi-robot teams via a stochastic clustering auction," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 7, no. 2, 2012.
- [34] B. Xie, J. Chen, and L. Shen, "Cooperation Algorithms in Multi-Agent Systems for Dynamic Task Allocation: A Brief Overview,"

- Chinese Control Conference, CCC*, vol. 2018-July, pp. 6776–6781, 2018.
- [35] A. Whitbrook, Q. Meng, and P. W. Chung, “Reliable, Distributed Scheduling and Rescheduling for Time-Critical, Multiagent Systems,” *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 2, pp. 732–747, 2018.
 - [36] M. Bernardine Dias, R. Zlot, N. Kalra, and A. Stentz, “Market-based multirobot coordination: A survey and analysis,” *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1257–1270, 2006.
 - [37] A. R. Mosteo and L. Montano, “A Survey of multi-robot task allocation,” *Caai Transactions on Intelligent Systems*, vol. 2008, no. 02, pp. 1–27, 2008.
 - [38] X. Gallud and D. Selva, “Agent-based simulation framework and consensus algorithm for observing systems with adaptive modularity,” *Systems Engineering*, vol. 21, no. 5, pp. 432–454, 2018.
 - [39] H. L. Choi, L. Brunet, and J. P. How, “Consensus-based decentralized auctions for robust task allocation,” *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 912–926, 2009.
 - [40] K. S. Kim, H. Y. Kim, and H. L. Choi, “A Bid-Based Grouping Method for Communication-Efficient Decentralized Multi-UAV Task Allocation,” *International Journal of Aeronautical and Space Sciences*, vol. 21, no. 1, pp. 290–302, 2020. [Online]. Available: <https://doi.org/10.1007/s42405-019-00205-1>
 - [41] X. Chen, P. Zhang, F. Li, and G. Du, “A cluster first strategy for distributed multi-robot task allocation problem with time constraints *,” *2018 WRC Symposium on Advanced Robotics and Automation, WRC SARA 2018 - Proceeding*, pp. 83–89, 2018.
 - [42] Smith, “The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver,” *IEEE Transactions on Computers*, vol. C-29, no. 12, pp. 1104–1113, dec 1980. [Online]. Available: http://www.csupomona.edu/~ftang/courses/CS599-DI/notes/papers/contract_net_protocol.pdf
<http://ieeexplore.ieee.org/document/1675516/>
 - [43] A. Liekna, E. Lavendelis, and A. Grabovskis, “Experimental analysis of contract net protocol in multi-robot task allocation,” *Applied Computer Systems*, vol. 13, no. 1, pp. 6–14, 2013.
 - [44] W. L. Yeung, “Efficiency of task allocation based on contract net protocol with audience restriction in a manufacturing control application,” *International Journal of Computer Integrated Manufacturing*, vol. 31, no. 10, pp. 1005–1017, 2018. [Online]. Available: <https://doi.org/10.1080/0951192X.2018.1493227>
 - [45] B. Du and S. Li, “A new multi-satellite autonomous mission allocation and planning method,” *Acta Astronautica*, vol. 163, pp. 287–298, oct 2019. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0094576518312475>
 - [46] N. Hooshangi and A. Asghar Alesheikh, “Agent-based task allocation under uncertainties in disaster environments: An approach to interval uncertainty,” *International Journal of Disaster Risk Reduction*, vol. 24, no. February, pp. 160–171, 2017. [Online]. Available: <http://dx.doi.org/10.1016/j.ijdr.2017.06.010>
 - [47] S. A. Nelke, S. Okamoto, and R. Zivan, “Market clearing-based dynamic multi-agent task allocation,” *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 1, Jan. 2020. [Online]. Available: <https://doi.org/10.1145/3356467>
 - [48] V. Singhal and D. Dahiya, “Distributed task allocation in dynamic multi-agent system,” *International Conference on Computing, Communication and Automation, ICCCA 2015*, pp. 643–648, 2015.
 - [49] M. E. Mkiramweni, C. Yang, J. Li, and Z. Han, “Game-Theoretic Approaches for Wireless Communications with Unmanned Aerial Vehicles,” *IEEE Wireless Communications*, vol. 25, no. 6, pp. 104–112, 2018.
 - [50] I. Jang, H. S. Shin, and A. Tsourdos, “Anonymous Hedonic Game for Task Allocation in a Large-Scale Multiple Agent System,” *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1534–1548, 2018.
 - [51] B. Xie, X. Gu, J. Chen, and L. C. Shen, “A multi-responsibility-oriented coalition formation framework for dynamic task allocation in mobile-distributed multi-agent systems,” *International Journal of Advanced Robotic Systems*, vol. 15, no. 6, pp. 1–15, 2018.
 - [52] V. R. Makkapati and P. Tsiotras, “Optimal Evading Strategies and Task Allocation in Multi-player Pursuit-Evasion Problems,” *Dynamic Games and Applications*, vol. 9, no. 4, pp. 1168–1187, dec 2019.
 - [53] X. Fu, J. Zhang, L. Zhang, and S. Chang, “Coalition formation among unmanned aerial vehicles for uncertain task allocation,” *Wireless Networks*, vol. 25, no. 1, pp. 367–377, 2019.
 - [54] J. Jesus Roldan, J. Del Cerro, and A. Barrientos, “Should We Compete or Should We Cooperate? Applying Game Theory to Task Allocation in Drone Swarms,” in *IEEE International Conference on Intelligent Robots and Systems*, 2018, pp. 5366–5371.
 - [55] W. Dai, H. Lu, J. Xiao, and Z. Zheng, “Task Allocation Without Communication Based on Incomplete Information Game Theory for Multi-robot Systems,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 94, no. 3–4, pp. 841–856, 2019.
 - [56] M. Badreldin, A. Hussein, and A. Khamis, “A Comparative Study between Optimization and Market-Based Approaches to Multi-Robot Task Allocation,” *Advances in Artificial Intelligence*, vol. 2013, pp. 1–11, 2013. [Online]. Available: <https://www.hindawi.com/archive/2013/256524/>
 - [57] J. Odili, M. Nizam, M. Kahar, and A. Noraziah, “A comparative evaluation of swarm intelligence techniques for solving combinatorial optimization problems,” no. June, pp. 1–11, 2017.
 - [58] H. W. Kuhn, “The Hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, vol. 2, no. 1–2, pp. 83–97, mar 1955. [Online]. Available: <http://journals.lww.com/00001813-199806000-00001> <http://doi.wiley.com/10.1002/nav.3800020109>
 - [59] L. Liu and D. A. Shell, “Assessing optimal assignment under uncertainty: An interval-based algorithm,” *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 936–953, jun 2011. [Online]. Available: <http://journals.sagepub.com/doi/10.1177/0278364911404579>
 - [60] S. Yoon and J. Kim, “Efficient multi-agent task allocation for collaborative route planning with multiple unmanned vehicles,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3580–3585, 2017. [Online]. Available: <https://doi.org/10.1016/j.ifacol.2017.08.686>
 - [61] —, “Fleet size optimization and collaborative route planning for multi-vehicle task allocation,” in *AIAA Scitech 2019 Forum*, no. January, 2019, pp. 1–13.
 - [62] E. F. Flushing, L. M. Gambardella, and G. A. Di Caro, “Simultaneous task allocation, data routing, and transmission scheduling in mobile multi-robot teams,” in *IEEE International Conference on Intelligent Robots and Systems*, vol. 2017-Sept, 2017, pp. 1861–1868.
 - [63] I. Tkach, A. Jevtić, S. Y. Nof, and Y. Edan, “A modified distributed bees algorithm for multi-sensor task allocation,” *Sensors (Switzerland)*, vol. 18, no. 3, 2018.
 - [64] E. Bonabeau, G. Theraulaz, and J. L. Deneubourg, “Fixed response thresholds and the regulation of division of labor in insect societies,” *Bulletin of Mathematical Biology*, vol. 60, no. 4, pp. 753–807, 1998.
 - [65] Y. Tan and Z. yang Zheng, “Research Advance in Swarm Robotics,” pp. 18–39, mar 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.dt.2013.03.001>
<https://linkinghub.elsevier.com/retrieve/pii/S221491471300024X>
 - [66] G. Theraulaz, E. Bonabeau, and J. L. Deneubourg, “Response threshold reinforcement and division of labour in insect societies,” *Proceedings of the Royal Society B: Biological Sciences*, vol. 265, no. 1393, pp. 327–332, 1998.
 - [67] L. Bayindir, “A review of swarm robotics tasks,” *Neurocomputing*, vol. 172, pp. 292–321, 2016.
 - [68] Y. Miao, L. Zhong, Y. Yin, C. Zou, and Z. Luo, “Research on dynamic task allocation for multiple unmanned aerial vehicles,” *Transactions of the Institute of Measurement and Control*, vol. 39, no. 4, pp. 466–474, 2017.
 - [69] F. Zhang, “Intelligent task allocation method based on improved QPSO in multi-agent system,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 2, pp. 655–662, 2020. [Online]. Available: <http://dx.doi.org/10.1007/s12652-019-01242-0>
 - [70] M. Roshanzamir, M. A. Balafar, and S. N. Razavi, “A new hierarchical multi group particle swarm optimization with different task allocations inspired by holonic multi agent systems,” *Expert Systems with Applications*, vol. 149, p. 113292, 2020. [Online]. Available: <https://doi.org/10.1016/j.eswa.2020.113292>
 - [71] S. Saeedvand, H. S. Aghdasi, and J. Baltes, “Robust multi-objective multi-humanoid robots task allocation based on novel hybrid meta-heuristic algorithm,” *Applied Intelligence*, vol. 49, no. 12, pp. 4097–4127, 2019.
 - [72] A. Mukherjee, P. Goswami, Z. Yan, L. Yang, and J. J. Rodrigues, “ADAI and Adaptive PSO-Based Resource Allocation for Wireless Sensor Networks,” *IEEE Access*, vol. 7, pp. 131 163–131 171, 2019.
 - [73] J. Guerrero, J. J. Miñana, O. Valero, and G. Oliver, “Indistinguishability operators applied to task allocation problems in multi-agent systems,” *Applied Sciences (Switzerland)*, vol. 7, no. 10, 2017.

- [74] I. Vandermeulen, R. Grofi, and A. Kolling, "Balanced task allocation by partitioning the multiple traveling salesperson problem," *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 3, pp. 1479–1487, 2019.
- [75] A. Dutta and A. Asaithambi, "One-to-many bipartite matching based coalition formation for multi-robot task allocation," in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2019-May, 2019, pp. 2181–2187.
- [76] C. Sarkar, H. S. Paul, and A. Pal, "A Scalable Multi-Robot Task Allocation Algorithm," in *Proceedings - IEEE International Conference on Robotics and Automation*. IEEE, 2018, pp. 5022–5027.
- [77] J. Bae, J. Lee, and W. Chung, "A heuristic for task allocation and routing of heterogeneous robots while minimizing maximum travel cost," in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2019-May, 2019, pp. 4531–4537.
- [78] Y. T. Tian, M. Yang, X. Y. Qi, and Y. M. Yang, "Multi-robot task allocation for fire-disaster response based on reinforcement learning," *Proceedings of the 2009 International Conference on Machine Learning and Cybernetics*, vol. 4, no. July, pp. 2312–2317, 2009.
- [79] D. B. Noureddine, A. Gharbi, and S. B. Ahmed, "Multi-agent deep reinforcement learning for task allocation in dynamic environment," *ICSOF 2017 - Proceedings of the 12th International Conference on Software Technologies*, no. Icsotf, pp. 17–26, 2017.
- [80] T. S. Dahl, M. J. Mataric, and G. S. Sukhatme, "A Machine Learning Method for Improving Task Allocation in Distributed Multi-Robot Transportation," in *Complex Engineered Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, vol. 2006, pp. 307–337. [Online]. Available: http://link.springer.com/10.1007/3-540-32834-3_14
- [81] A. Majkowska, D. Zydek, and L. Koszałka, "Task Allocation in Distributed Mesh-Connected Machine Learning System: Simplified Busy List Algorithm with Q-Learning Based Queuing," in *Advances in Intelligent Systems and Computing*, ser. Advances in Intelligent Systems and Computing, R. Burduk, K. Jackowski, M. Kurzynski, M. Wozniak, and A. Zolnierok, Eds. Heidelberg: Springer International Publishing, 2013, vol. 226, pp. 763–772. [Online]. Available: http://link.springer.com/10.1007/978-3-319-00969-8_75
- [82] G. Best and G. A. Hollinger, "Decentralised self-organising maps for the online orienteering problem with neighbourhoods," in *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*. IEEE, aug 2019, pp. 139–141. [Online]. Available: <https://ieeexplore.ieee.org/document/8901053/>
- [83] T. Luo, B. Subagdja, D. Wang, and A.-h. Tan, "Multi-Agent Collaborative Exploration through Graph-based Deep Reinforcement Learning," in *2019 IEEE International Conference on Agents (ICA)*. IEEE, oct 2019, pp. 2–7. [Online]. Available: <https://ieeexplore.ieee.org/document/8929168/>
- [84] S. E. Hammami, H. Afifi, H. Mougla, and A. Kamel, "Drone-Assisted Cellular Networks: A Multi-Agent Reinforcement Learning Approach," in *IEEE International Conference on Communications*, vol. 2019-May, 2019.
- [85] J. Han, Z. Zhang, and X. Wu, "A real-world-oriented multi-task allocation approach based on multi-agent reinforcement learning in mobile crowd sensing," *Information (Switzerland)*, vol. 11, no. 2, 2020.
- [86] M. J. Bays and T. A. Wettergren, "Partially-Decoupled Service Agent - Transport Agent Task Allocation and Scheduling," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 94, no. 2, pp. 423–437, 2019.
- [87] B. Xie, S. Chen, J. Chen, and L. C. Shen, "A mutual-selecting market-based mechanism for dynamic coalition formation," *International Journal of Advanced Robotic Systems*, vol. 15, no. 1, pp. 1–10, 2018.
- [88] J. Zhang, G. Wang, and Y. Song, "Task assignment of the improved contract net protocol under a multi-agent system," *Algorithms*, vol. 12, no. 4, 2019.
- [89] F. Zitouni, S. Harous, and R. Maamri, "A Distributed Approach to the Multi-Robot Task Allocation Problem Using the Consensus-Based Bundle Algorithm and Ant Colony System," *IEEE Access*, vol. 8, pp. 27 479–27 494, 2020. [Online]. Available: <https://ieeexplore.ieee.org/document/8981979/>
- [90] J. Turner, G. Schaefer, Q. Meng, and A. Soltoggio, "Distributed strategy adaptation with a prediction function in multi-agent task allocation," *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 1, no. Aamas, pp. 739–747, 2018.
- [91] J. Zhou, X. Zhao, X. Zhang, D. Zhao, and H. Li, "Task allocation for multi-agent systems based on distributed many-objective evolutionary algorithm and greedy algorithm," *IEEE Access*, vol. 8, pp. 19 306–19 318, 2020.
- [92] H. Wu and H. Shang, "Potential game for dynamic task allocation in multi-agent system," *ISA Transactions*, mar 2020. [Online]. Available: <https://doi.org/10.1016/j.isatra.2020.03.004> <https://linkinghub.elsevier.com/retrieve/pii/S0019057820301129>
- [93] B. Gerkey and M. Mataric, "Multi-robot task allocation: analyzing the complexity and optimality of key architectures," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 3, no. Icara. IEEE, 2003, pp. 3862–3868. [Online]. Available: <http://ieeexplore.ieee.org/document/1242189/>
- [94] Z. Huang and K. Yi, "The communication complexity of distributed epsilon-approximations," *SIAM Journal on Computing*, vol. 46, no. 4, pp. 1370–1394, 2017.
- [95] D. Albani, W. Hönig, N. Ayanian, D. Nardi, and V. Trianni, "Summary: Distributed task assignment and path planning with limited communication for robot teams," *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 3, pp. 1770–1772, 2019.
- [96] Z. Yan, N. Jouandeau, and A. A. Cherif, "A Survey and Analysis of Multi-Robot Coordination," *International Journal of Advanced Robotic Systems*, vol. 10, no. 12, p. 399, dec 2013. [Online]. Available: <http://journals.sagepub.com/doi/10.5772/57313>
- [97] A. Whitbrook, Q. Meng, and P. W. Chung, "A robust, distributed task allocation algorithm for time-critical, multi agent systems operating in uncertain environments," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, ser. Lecture Notes in Computer Science, S. Benferhat, K. Tabia, and M. Ali, Eds., vol. 10351 LNCS, no. 2015. Cham: Springer International Publishing, 2017, pp. 55–64. [Online]. Available: <http://link.springer.com/10.1007/978-3-319-60045-1> http://link.springer.com/10.1007/978-3-319-60045-1_8
- [98] —, "Addressing robustness in time-critical, distributed, task allocation algorithms," *Applied Intelligence*, vol. 49, no. 1, pp. 1–15, 2019.
- [99] J. M. Gregory, S. Al-Hussaini, and S. K. Gupta, "Heuristics-Based Multi-Agent Task Allocation for Resilient Operations," in *2019 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSR 2019*. IEEE, 2019, pp. 270–277.
- [100] M. Rantanen, J. Modares, N. Mastrorade, F. Ghanei, and K. Dantu, "Performance of the asynchronous consensus based bundle algorithm in lossy network environments," *Proceedings of the IEEE Sensor Array and Multichannel Signal Processing Workshop*, vol. 2018-July, pp. 311–315, 2018.
- [101] Y. Lu and K. Yan, "Algorithms in multi-agent systems: A holistic perspective from reinforcement learning and game theory," 2020.