Cooperative Heterogeneous Multi-Robot Systems: A Survey

YARA RIZK and MARIETTE AWAD, American University of Beirut
EDWARD W. TUNSTEL, United Technologies Research Center

The emergence of the Internet of things and the widespread deployment of diverse computing systems have led to the formation of heterogeneous multi-agent systems (MAS) to complete a variety of tasks. Motivated to highlight the state of the art on existing MAS while identifying their limitations, remaining challenges, and possible future directions, we survey recent contributions to the field. We focus on robot agents and emphasize the challenges of MAS sub-fields including task decomposition, coalition formation, task allocation, perception, and multi-agent planning and control. While some components have seen more advancements than others, more research is required before effective autonomous MAS can be deployed in real smart city settings that are less restrictive than the assumed validation environments of MAS. Specifically, more autonomous end-to-end solutions need to be experimentally tested and developed while incorporating natural language ontology and dictionaries to automate complex task decomposition and leveraging big data advancements to improve perception algorithms for robotics.

CCS Concepts: • **Computing methodologies** → **Planning and scheduling**; **Multi-agent planning**; **Robotic planning**; **Multi-agent systems**;

Additional Key Words and Phrases: Multi-robot systems, cooperation, heterogeneous agents, task decomposition, task allocation, coalition formation, decision-making models, perception

ACM Reference format:
Yara Rizk, Mariette Awad, and Edward W. Tunstel. 2019. Cooperative Heterogeneous Multi-Robot Systems: A Survey. *ACM Comput. Surv.* 52, 2, Article 29 (April 2019), 31 pages.
https://doi.org/10.1145/3303848

1 INTRODUCTION

The dynamic and unpredictable nature of the world we live in makes it difficult to design one autonomous robot that can efficiently adapt to all circumstances. Therefore, robots of various shapes, sizes, and capabilities such as unmanned aerial vehicles (UAVs), unmanned ground vehicles (UGVs), humanoids, and others have been designed to cooperate with each other and humans to successfully accomplish complex tasks. Allowing these diverse connected devices, expected to surpass $20 billion by 2020 with the emergence of the Internet of things (IoT) [39], to cooperate will significantly increase the spectrum of automated tasks. Integrating these devices in areas such as

This work was supported in part by the National Council of Scientific Research in Lebanon (CNRS-L) and in part by the University Research Board (URB) at the American University of Beirut.
Authors' addresses: Y. Rizk and M. Awad, American University of Beirut, Riad El Solh Street, Beirut, Beirut 1107 2020, Lebanon; emails: {yar01, mariette.awad}@aub.edu.lb; E. W. Tunstel, United Technologies Research Center, 411 Silver Ln, East Hartford, CT 06108, USA; email: tunstel@ieee.org.

0360-0300/2019/04-ART29 $15.00
https://doi.org/10.1145/3303848

health care, transportation systems, emergency response systems, household chores, and elderly care, among others, will make smart cities even smarter.

In this work, we discuss the literature on automating complex tasks using heterogeneous multi-robot systems (MRS), after a brief overview of the more general multi-agent system (MAS) field. We present the main components of a workflow to automate MRS: task decomposition, coalition formation, task allocation, perception, and MAS planning and control, survey existing work in each area, and identify some remaining challenges and possible future research directions.

However, many aspects of heterogeneous MAS are not covered in this survey. They include, but are not limited to, credit assignment, which studies reward distribution among agents [226]; consensus, which investigates protocols that ensure agent agreement under different circumstances [102, 237]; and containment control, which is a type of consensus in leader-follower models [236] and robot hardware design. Although communication protocols for robot–robot communication that enable agent cooperation through information exchange [30, 56] and the information flow problem that seeks to design efficient information exchange strategies [30] are an important component of MAS, we do not include them in this survey. Instead, readers are invited to review References [30, 56].

Few end-to-end frameworks have been presented for MRS; the most notable is swarmanoids [55], which accomplished search and retrieval tasks using a swarm of three types of robots. Many solutions still required significant human intervention to achieve complex tasks. Furthermore, individual aspects of MRS have been tackled, such as giving robots access to information on the cloud [175] and simultaneous coalition formation and task allocation [233]. However, more end-to-end testing on IoT-aided robotics [77] and MRS applications should be conducted to achieve more progress. Natural language ontology and dictionaries could help automate complex task decomposition and big data advancements could be leveraged to improve perception and, consequently, decision making.

Two of the closest surveys to our work were published almost a decade ago; one covered MRS coordination, including task allocation, decomposition, and resource distribution [54]. However, it focused on market-based approaches and did not include work on coalition formation and decision-making models. A more recent survey discussed existing MRS architectures, communication schemes, swarm robotics, task allocation, and learning [157] in applications like foraging, formation control, cooperative object manipulation and displacement, path planning, and soccer [157].

Other surveys had a narrower scope than our work and focused on a specific research area such as cooperative MAS planning and control models and algorithms [155] and distributed consensus in MAS [235]. Some surveys focused on MRS and their assigned tasks; Arai et al. identified seven main research areas in MRS, including robot architectures, mapping and exploration, and motion coordination, and discussed state-of-the-art research and challenges in each area [13]. Ota surveyed tasks assigned to MRS, classifying them into point reaching, region sweeping, and compound tasks, in addition to one-time and many-time tasks, i.e., tasks that require multiple iterations for completion [152]. Murray surveyed cooperative control of multi-vehicle systems and their applications in the military, transportation systems and mobile sensor networks [142]. Portugal and Rocha surveyed MRS for patrolling [164], whereas Robin and Lacroix focused on MRS target detection and tracking [164]. Yan et al. covered coordination in mobile MRS that included communication protocols and decision-making algorithms for both motion and task planning [231]. Doriya et al. presented communication and coordination in MRS that included implicit and explicit communication algorithm [56]. Centralized and distributed coordination were covered, including task decomposition, allocation, and execution. Merrick surveyed decision-making algorithms for robotics [136]. Grieco et al. discussed IoT-robotics integration to enhance performance and envisioned scenarios in health care, military, and smart cities [77].

Next, Section 2 defines key terminology in MAS and presents the MRS workflow. Section 3 surveys implemented MRS. Sections 4 to 7 present existing work on each block of the workflow before discussing remaining challenges in Section 8 and concluding in Section 9 with final remarks.

2 MULTI-AGENT SYSTEMS

2.1 Intelligent Agents

An intelligent agent is a physical (robot) or virtual (software program) entity that can autonomously perform actions on an environment while perceiving this environment to accomplish a goal [184]. A rational agent seeks to perform actions that result in the best outcome [184]. A cognitive architecture is the "underlying infrastructure for an intelligent agent" [115]: the agent's brain. It consists of perception, reasoning, learning, decision making, problem solving, interaction, and communication. Its evaluation is based on domain-specific performance measures, generality, versatility, rationality, optimality, efficiency, scalability, autonomy, and improvability [115]. Readers are invited to check an extensive survey on existing cognitive architectures in Reference [110].

Many agent categorizations have been proposed in the literature. One categorization distinguishes three types of agents: reactive, deliberative, and hybrid agents. Reactive agents simply react to environmental changes. Their workflow contains two primitives: sense (S) and act (A). Deliberative agents initiate actions without any external trigger and rely on planning. This sense-plan-act or sense-model-plan-act paradigm contains three primitives that are performed sequentially: sense (S), plan (P), and act (A). Hybrid agents perform actions based on a planning algorithm or react to current perceptions. The workflows for these three types of agents are represented in Figure 1. A finer categorization divides agents into the following: simple reflex (react to current sensory input), model-based reflex (keep an internal state of the environment), goal-based (perform actions to complete a goal), and utility-based (maximize a utility function) agents [184]. These four categories are considered learning agents if they learn an element of the environment or their control algorithms' parameters with the help of a critic.

2.2 Multi-agent Systems

MAS are composed of multiple autonomous, interacting agents that have common or conflicting goals and sensory information [196]. They are characterized by decentralized and incomplete information, asynchronous computations, and decentralized control [226]. However, centralized or hybrid systems are also considered MAS. MRS restrict agents to physical robots [13]. MAS has been viewed as an area in distributed artificial intelligence "concerned with coordinated, concurrent action and problem solving" [27], with the second sub-field being distributed problem solving. However, Parker defined distributed intelligence as a group of entities that perform cognitive functions such as reasoning, solving problems, and learning [156]. The term *cognitive computing system* has also been used to refer to MAS and is defined as hardware-software co-optimized architectures composed of diverse intelligent agents that can interact with humans and each other to complete tasks by exploiting each entity's strengths [90]. Mobile cognition implements distributed cognitive computing architectures on mobile platforms such as robots, cars, and smart phones.

MAS evaluation criteria are either domain specific or invariant [47]. Domain-specific criteria quantify performance. For search and rescue, performance measures include the number of rescued persons or extinguished fires [64]. Domain-invariant criteria include solution optimality, algorithm time and space complexity, load balancing, fairness, resource utilization and re-allocation quickness, communication overhead, robustness to noise and agent failures, and scalability [47].

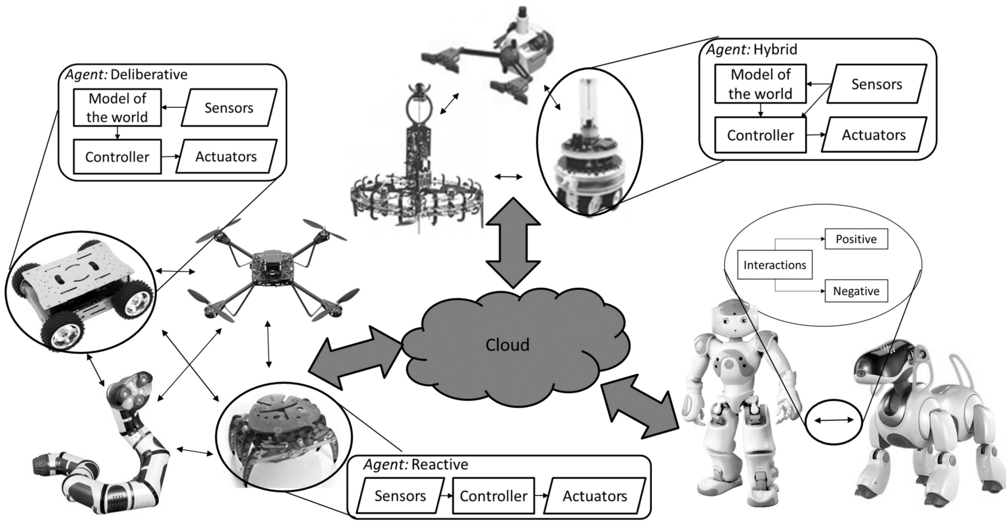

Fig. 1. Three-tier heterogeneous MRS architecture: robot, locally connected MRS, group of MRS connected through the cloud.

MAS have been divided into categories based on multiple criteria. One division, based on agents' diversity and communication capabilities, consists of four classes: homogeneous non-communicative, homogeneous communicative, heterogeneous non-communicative, and heterogeneous communicative [205]. Agent diversity can be from sensory or actuation capabilities, cognition algorithms, or morphology [157]. MAS have also been classified as centralized, hierarchical, decentralized, or hybrid architectures [157]. Considering agent interaction complexity leads to three classes: no direct interaction, simple interaction, and complex conditional interaction [145].

Focusing on interaction types, MAS can exhibit cooperative, competitive, and collaborative interaction based on goals, resources, and agent skills [65, 156]. A broader classification is positive versus negative where agents aid or do not interfere with each other versus actively impede other's agents. We mainly focus on cooperative interaction where agents are aware of other agents and share the same goals, and their individual actions lead to the accomplishment of the common goal. Examples include search and rescue, exploration, classification of a target, and displacing objects, to name a few [142].

Figure 1 represents a heterogeneous MRS architecture with three levels of hierarchy. At the highest level, information about all the robots and the complex task is available in the cloud. Robots can communicate through the cloud and make use of any computational resources and information available in the cloud. The lower level (MRS) contains a subset of robots with an assigned subtask. Interaction between this subset or coalition is local and can be one of the different types of interactions already discussed. Information is gathered from the various robots' sensors and exchanged among them. Finally, the lowest level is the agent, which has access to its sensory input and control of its own actuators. It can communicate with other robots within its coalition and can connect to the cloud. Since the system is heterogeneous, agents are not identical and can have different cognitive architectures (reactive, deliberative, or hybrid) or different physical properties (e.g., UGVs and UAVs). This three-tier architecture can lead to automating complex tasks that could not be automated in simpler MRS frameworks.

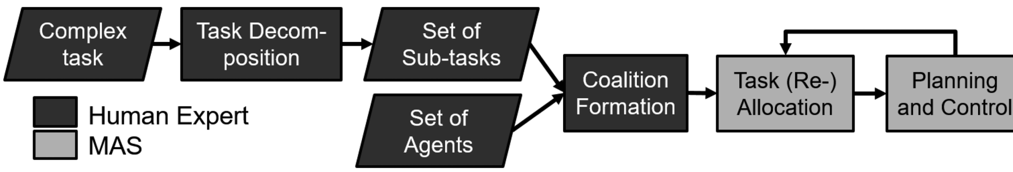

Fig. 2. Workflow for the MRS proposed in Reference [101].

2.3 Task Taxonomy

Cooperative MRS are assigned a wide variety of tasks with varying degrees of complexity. Task complexity determines the difficulty of the task, which affects the number and type of robots needed to complete it and may be composed of multiple simpler sub-tasks. Single-robot tasks can be accomplished by one robot [72], such as small-scale mapping, pick and place, and navigation problems. Multi-robot tasks require multiple cooperating robots [72]. Multi-robot tasks can be further distinguished based on the required level of cooperation for successfully completion, ranging from loosely to tightly coordinated. Loosely coordinated tasks can be decomposed to sub-tasks that can be independently executed with minimum interaction among robots. Examples include large-scale exploration and mapping, hazardous material clean-up, tracking, and surveillance. In such scenarios, the environment can be divided into disjoint areas and the robots operate within their specified areas. Tightly coupled tasks are not decomposable and require coordinated execution with significant interaction among robots. Examples include robot soccer, object transport, and large-scale construction.

2.4 Multi-robot System Workflow

To systematically design MRS capable of accomplishing complex tasks, four main design blocks are identified [32, 101]: (1) task decomposition, which is division of complex tasks into simpler sub-tasks; (2) coalition formation, whic is formation of agent teams; (3) task allocation, which is the assignment of sub-tasks to agent teams for execution; and (4) task execution/planning and control, which is the completion of a task by performing a sequence of actions on the environment.

The literature has adopted varying degrees of automation of the different components, in addition to various inputs and constraints on each component. Figure 2 presents the workflow proposed in Reference [101] that required a human designer to decompose complex tasks to simpler sub-tasks based on the available robots' capabilities and form coalitions from a set of agents. Then task allocation and robot planning and control are autonomously performed by the robot teams. While many researchers have opted to statically assign coalitions in MRS to simplify the design, others have performed dynamic coalition formation.

In Section 3, we present existing MRS that have been developed for various tasks. These systems mostly follow the framework in Figure 2, but some have incorporated an automated coalition formation algorithm. To the best of our knowledge, there are no implementations of fully automated systems, i.e., a complex task was given to the MRS, and the system autonomously decomposed this complex task to sub-tasks, formed coalitions, and assigned and executed these sub-tasks.

This framework, considered a cognitive architecture as per the definition in Reference [110], is generally embedded in software architectures that enable the integration of the various components while abstracting some of the complexity. Agent-based modeling and simulation tools have been developed in the literature to model and test agents and MAS. Such tools are generally classified based on the type of agent they implement (e.g., reactive or mobile), their modeling strength and scalability, their application domain, and the effort they require to implement systems. An exhaustive survey of these tools can be found in Reference [1]. Many software architectures have been developed specifically for robots and MRS [92] that use domain-specific modeling and languages to create more powerful modeling tools. Middleware architectures specific to MRS such as ROS, Orocos, and OpenRTM, surveyed in Reference [128], have also been developed to model communication protocols improve function portability through hardware abstraction [128]. Domain-specific languages for robotics, surveyed in Reference [14], allow even higher levels of abstraction and automation.

2.5 Applications

MAS that combine robotic agents, software agents, and other computing and sensing devices have many applications in smart cities, including health care, domestic services, and intelligent transportation systems. Military applications such as surveillance, navigation, and target tracking have also developed MAS frameworks to efficiently automate these complex tasks. For illustration purposes, we discuss two scenarios in which MAS can be used: emergency response and health care.

MAS can be deployed in emergency response situations. During natural or human-made disasters (wars), robots with diverse capabilities such as snake bots, crawlers, and miniature drones can navigate the debris searching for injured individuals, while larger robots would retrieve these people, and transport UGVs would take them to hospitals. Transport robots would be able to obtain information from the intelligent transportation system for information on fastest routes and the traffic lights can be controlled to direct traffic. Therefore, MRS will need to leverage the cloud and other non-robotic agents to improve the system.

In smart cities, an elderly care system can consist of one or more mobile robots capable of providing round the clock physical care to elderly patients, and a monitoring system capable of recognizing the occurrence of falls or medical emergencies and notifying the mobile robots accordingly. The system can be expanded further by allowing it to communicate with the patient's health care team and relatives for updates and medication delivery. A similar system was envisioned in Reference [11] to improve health care for children with complex conditions but did not include a robot agent. Kraus discussed the capabilities that a robot should possess to effectively aid patients with disabilities [111]. It should develop personalized training regimens, monitor the patient's activities, and intervene and provide encouragement when needed. Woo et al. developed a system consisting of a wireless sensor network and a robot for elderly care situations like the described scenario, but the robot was only used for companionship and did not physically care for the patient [229]. Benavidez et al. proposed an MRS for elderly and disabled home care consisting of cloud computing resources and heterogeneous robots performing various tasks like house monitoring and floor cleaning [23].

3 EXISTING MRS

Many heterogeneous MRS have been proposed to solve a broader set of complex cooperative tasks. These systems allowed heterogeneous robots, i.e., robots with diverse capabilities, such as UAVs and UGVs, to cooperate on complex tasks. They have all taken a step toward deployment in the real world, but they still make some assumptions or simplifications that limit the system's generalization. We discuss some of these systems next, grouping them based on the complexity of the cooperative tasks they execute and their level of automation, from most to least automated. In the first level (least automated), only task execution is automated, while the second level also automates either task allocation or coalition formation but not both. The third level automates both coalition formation and task allocation but does not automate task decomposition. The fourth level automates the entire system. Figure 3 portrays the distribution of existing work across levels of automation and task complexity. Table 1 summarizes the literature on MRS and their tasks, mentioning only those references with a second or greater automation level. "N/A" stands for not

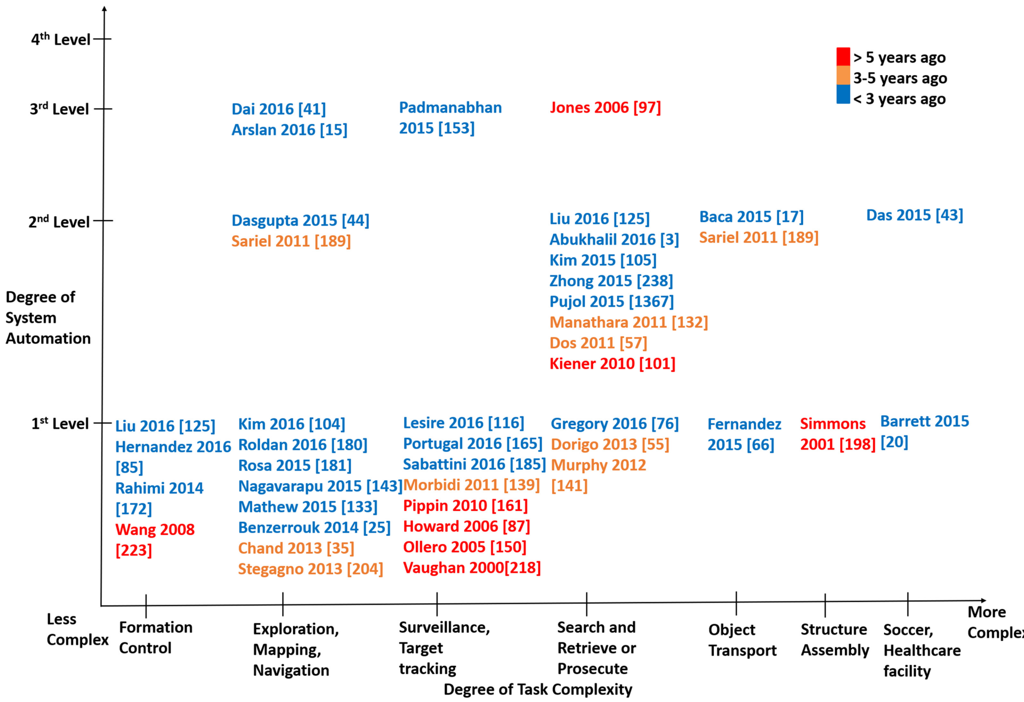

Fig. 3. Comparing existing work based on the task complexity and the degree of system automation.

applicable, i.e., this component of the workflow was not considered in the cited work. To the best of our knowledge, no references were found that automated the entire process (fourth level of automation).

3.1 Third Level of Automation

A few references achieved the third level of automation, where coalition formation, task allocation, and task execution were performed by a MRS without any human interference. The systems used UGVs for treasure hunts [97], UAVs for surveillance [153], collision-free UGV navigation [15], and UGV exploration [41]. Autonomous coalition formation and task allocation algorithms included TraderBots, which is based on auctioning [53, 97], hierarchical clustering [15], ant colony optimization and memetic search algorithm [153], and supervisory control theory [41]. These algorithms were mainly tested in simulation environments or in small-sized experiments but have shown promising results.

3.2 Second Level of Automation

The following references achieved the second level of automation, which can be divided into two main categories: (1) task decomposition and coalitions are predefined by human experts, but task allocation and execution are performed autonomously by MRS, or (2) task decomposition and allocation are performed by human experts, but coalition formation and task execution are performed by MRS.

The first set of references automated task allocation and execution. Applications included health care facility tasks [43], cooperative navigation and object construction [189], bar-pushing [17], ball tracking and kicking [101], and search and rescue or prosecute [3, 105, 167]. Task allocation

Table 1. Existing MRS

Reference, Year	Agents	Task	Task decomposition	Coalition formation	Task allocation	Decision-making model	Validation
[41], 2016	UGVs	Exploration	N/A	Supervisory control algorithm			Simulations
[15], 2016	UGVs	Navigation	N/A	Hierarchical Clustering	Hierarchical Clustering	Hierarchical Clustering	Simulations
[44], 2015	UGVs	Land mine detection	N/A	N/A	Prediction market	Voronoi partition coverage	Simulations, Experiments
[153], 2015	UAVs	Surveillance	N/A	Dynamic ANT	Multi-Robot Task allocation	Dynamics controllers	Simulations
[97], 2006	UGVs	Treasure hunt	Human Expert	Auctioning	TraderBots	Dynamics controllers	Experiments
[126], 2016	UAVs	Search and prosecute	N/A	Multistage sub-optimal coalition formation	N/A	Distributed finite state machines	Simulations
[3], 2016	UGVs	Search and rescue	Human Expert	N/A	Robot Utility	Swarm Intelligence	Simulations, Experiments
[105], 2015	UAVs	Search and prosecute	N/A	N/A	Welfare-based task allocation	Dynamics controllers	Simulations
[167], 2015	UGVs	Search and rescue	N/A	Human Expert	Binary Max Sum	Dynamics controllers	Simulations
[238], 2015	UAVs	Search and prosecute	N/A	Multistage sub-optimal coalition formation	N/A	Dynamics controllers	Simulations
[132], 2011	UAVs	Search and prosecute	N/A	Particle swarm optimization	N/A	Dynamics controller	Simulations
[57], 2011	UGVs	RoboCup Rescue	N/A	Swarm intelligence	N/A	Dynamics controllers	Simulations
[101], 2010	Wheeled UGV, humanoid	Kick ball into goal	Human Expert	Human Expert	Utility function	Dynamics controllers	Simulations, Experiments
[17], 2015	Modular robots	Bar pushing	N/A	SMART	N/A	SMART	Experiments
[189], 2011	Khepra II	Object transport, Cooperative navigation	N/A	N/A	Auctioning	Dynamics controllers	Simulations, Experiments
[43], 2015	UGVs	Health care facility	N/A	N/A	Auctioning	Dynamics controllers	Simulations

algorithms were mainly auction based where agents bid on tasks based on their capabilities and the highest bidders win the task for execution [189]; utility-based, where tasks were assigned to agents with the highest utility or qualification [3, 101], and binary max sum [167]. Some algorithms allowed task re-allocation [189], were implemented in a distributed framework [189], and allowed communication among robots [17]. MRS systems included swarms of heterogeneous UGVs [3], heterogeneous UAVs [105], and modular robots capable of re-configuring their shapes [17]. References included generally included both simulation and real-world experiments but mainly focused on loosely coordinated tasks such as search and rescue/prosecute, which are simpler than tightly coordinated tasks such as object transportation (bar-pushing).

Some work has automated coalition formation and task execution for various applications, including search and rescue [57], search and prosecute [126, 132, 238], and land mine detection [44]. Coalitions were formed of either UGVs [44, 57] or UAVs [126, 132, 238] but not both, which slightly simplified the coalition formation problem, since robot capabilities are less diverse. In some cases, the robots were homogeneous when considering their hardware capabilities, as in Reference [44]. Various coalition formation algorithms were adopted in these works, including swarm intelligence [57], particle swarm optimization [132], weighted voting games [44], and multi-stage coalition formation algorithms [126, 238]. Except for exploration and land mine detection that were validated in simulations and real-world environments [44], the other referenced works presented simulation results only. Nevertheless, homogeneous robots were deployed to simplify the experiments [44]. This highlights the difficulty of deploying MRS with autonomous dynamic coalition formation and automated task execution in an unrestricted real-world application.

3.3 First Level of Automation

Finally, we present references that only automated the task execution phase in MRS, grouping and ordering them based on task complexity. These references rely on human experts to decompose tasks into sub-tasks, group robots in coalition, and allocate tasks to these coalitions. PLASTIC-Policy was introduced to discover the best policy for cooperation among robots in ad hoc teams allowing robots to adapt on the fly and validated on RoboCup soccer [20]. This problem was modeled as a Markov decision process (MDP) and solved using a fitted Q-iteration algorithm. Three distinct robots, an overhead crane, a mobile manipulator, and a roving eye were used to precisely place a long heavy beam [198]. Combining ground and aerial robots improved the overall system performance. A distributed round robin Q-learning algorithm was developed to transport a hose using a group of UGVs [66].

Swarmanoids is an MRS composed of three types of robot swarms with complementary capabilities: eye-bots (quad-rotors with cameras), hand-bots (grippers), foot-bots (wheeled robots: e-pucks) [55]. The system performed object search and retrieval. A complex task was decomposed into sub-tasks by the human designer. Each sub-task can be performed by one type of robot in the system, i.e., the robots in the system were designed in a way to generate a one to one mapping between sub-tasks and robots, eliminating the need for task allocation. Therefore, this system could accomplish a more complex task even though its workflow is less automated due to the system design, which included robots with complementary skills and individual robots that can form predefined coalitions.

Underwater search and rescue in natural disaster scenarios was performed using unmanned underwater vehicles (UUVs) and UAVs [141]. Autonomous robots performed a rough preliminary search to identify areas of interest that were later more thoroughly investigated by teleoperated vehicles. A navigation and simultaneous localization and mapping (SLAM) algorithm was developed for information gathering in disaster relief [76]. OmniMapper SLAM is based on graph theory and square-root smoothing and mapping. The NavigationManager integrates the search-based planning and ARA*.

Tracking and surveillance applications adopted MRS to improve coverage by leveraging the heterogeneity of robots and the diverse media (land, air, water). HiDDeN, a distributed algorithm, was developed to supervise the completion of UAV and UUV missions in military surveillance and coast securing application [116]. A distributed framework was presented to control an air-ground MRS where a UGV cooperated with UAVs to complete tasks relayed by a ground station [162]. The system relies heavily on communication between the UGVs and UAVs. The overall system follows a behavior-based autonomy framework where sub-tasks are represented as functional components in a state machine. Field experiments on target detection and surveillance validated the proposed framework on a three-robot system. Focusing on UAVs, helicopters and airships with diverse sensing capabilities collaborated to track a target and inspect and monitor an area [150]. Ground systems included a behavioral graph-based collision avoidance and set point tracking navigation algorithm for MRS [185] and MRS patrolling based on Bayesian decision rules that was validated on indoor patrolling with six Pioneer robots [165]. A large number of robots were deployed for intruder detection and tracking by performing exploration and mapping of an unknown indoor environment using a few expensive robots and a large number of low-cost sensory robots [87]. A framework was developed to improve UAVs' self-localization and localization of objects of interest on the group by cooperating with a GPS-enabled UGV [218]. Finally, UAV-UGV MAS have also been used to perform gait monitoring [139].

Navigation, exploration, and mapping tasks have also leveraged UAV-UGV cooperation. For example, a large number UGVs performing motion tasks were aided by a UAV that communicated location related information [181]. Similarly, Stegagno et al. used a single UAV to improve UGVs' SLAM [204]. Mathew et al. developed a path planning algorithm for micro UAVs and UGVs performing package delivery tasks [133]. The problem was modeled as a multiple warehouse delivery problem that was transformed to a generalized traveling salesman problem and solved using the kernel sequence enumeration algorithm in polynomial time. Greenhouse sensor monitoring was achieved using UGVs and UAVs to record temperature, humidity, gas levels, and others [180]. While UAVs were teleoperated, UGVs navigated independently.

Navigation with collision avoidance was achieved using a decentralized Lyapunov synthesis algorithm [25]. A decentralized multi-robot depth-first search algorithm was proposed that allowed UGVs to efficiently explore unknown environments with minimal information exchange and collision avoidance [143]. A cooperative navigation algorithm was developed based on graph and networking theory to ensure complete coverage of the environment explored by mobile robots gathering information [104]. A hierarchical algorithm allowed robots with limited capabilities to perform mapping and exploration by receiving tasks from more powerful robots [35].

Formation control algorithms for MRS have been developed to control UGVs and UAVs using a Lyaponuv framework [172], while a distributed Lyapunov controller was presented for UGVs only in Reference [125]. Wheeled and legged robots were controlled to navigate and avoid obstacles in a given formation [223]. Finally, a planar-based algorithm for cooperative heterogeneous UAVs was developed [85].

4 TASK DECOMPOSITION

Task decomposition, the first step in the MAS workflow for complex task automation, divides a complex task into a set of simpler or more primitive sub-tasks that are either independent or sequentially dependent on each other. For example, mapping a building can be divided into mapping of individual floors and rooms in the building.

Planners for task decomposition problems can be general [28] or domain specific [36, 243]. An example of the latter is soccer task decomposition where covering the playing field is divided among robots based on relative position of the ball and players [36]. The ball, viewed as a gravitational source, creates a gravitational field around it and affects the sub-task assignments. While many systems require the designer to manually decompose complex tasks to a sequence of simpler sub-tasks [55, 101], some work have attempted to automate this process and can be divided into three main categories: decompose-then-allocate, allocate-then-decompose and simultaneous decomposition and allocation [54].

Decompose-then-allocate algorithms first decompose a complex task into a list of sub-tasks in a centralized fashion and then allocate the various sub-tasks to available agents. Task tree decomposition divided tasks based on logistic relationships in the battlefield [120]. Automatic decomposition and abstraction learned to divide complex decision-making tasks into sub-tasks from human demonstrations, using mutual information measures [40]. Tracking people in an environment was dynamically divided among robots based on geographical proximity [88]. These approaches have found limited success due to the high level of domain-specific understanding required to understand the relationship between sub-tasks. Allocate-then-decompose algorithms, such as the M+ algorithm [28], first allocate a list of tasks to agents, and then each agent divides this task into more primitive sub-tasks. This approach allows agents to decompose tasks based on the agents' specific skill set, allowing them to efficiently execute tasks. Finally, simultaneous task decomposition and allocation algorithms opted not to decouple the task decomposition and allocation steps and proposed a solution based on task trees and auctioning [241, 243]. This approach produces more system-specific decompositions by providing feedback to improve task decomposition based on agent capabilities. However, it could be more time-consuming due to sub-task fine-tuning.

5 COALITION FORMATION AND TASK ALLOCATION

After decomposing a complex task into a list of sub-tasks, these sub-tasks should be allocated to a robot or group of robots for execution. Since some of the sub-tasks are multi-robot tasks, they should be assigned to groups of cooperating robots. Next, we discuss research on forming coalitions of robots (coalition formation) and assigning to them sub-tasks (task allocation) before task execution can be performed.

5.1 Coalition Formation

Coalition or team formation divides agents into coalitions or groups. These agents may be non-cooperative [224] or cooperative. In this work, we focus on cooperative coalition formation. Coalition formation can be performed offline to form static coalitions or online to form dynamic teams that can adjust to the environment [78]. Agents are categorized into single-task and multi-task agents [72], i.e., agents that can perform a single task versus those that can perform multiple tasks. As mentioned previously, tasks are either single-robot or multi-robot tasks. Finally, the task to agent mapping or assignment is categorized into instantaneous and time extended [72]. Architectures to represent agent capabilities and task requirements have been developed, including numeric representations [107] and behavior-based representations like schema theory [131]. Many search algorithms have been adopted to find the best robot teams, including ant colony optimization [232], particle swarm optimization [230], and evolutionary algorithms [121]. Search algorithms are simple approaches that do not require significant agent modeling before coalitions can be formed. However, their effectiveness depends on the quality of the metrics adopted to quantify the compatibility of agents within a coalition.

Repeated coalition formation under uncertainty deals with forming time-varying teams when agents have partial information about other agents' capabilities, resulting in uncertainty. Furthermore, information can be heterogeneous, i.e., from different sources [112]. Dynamic, repeated coalition formation with robot type and other uncertainties was performed by incorporating an agent modeling algorithm with game theory [86]. Bayesian reinforcement learning (RL) allowed agents to learn other agents' capabilities through their interactions and transformed the repeated coalition formation problem into a sequential decision-making problem [34]. This approach was validated on a football team formation problem [134]. Dynamic robot coalition formation for area coverage problems was modeled using weighted voting games and Q-learning [46] and extended to formation-based navigation problems [45]. Coalition structures were pruned using the Shapley value and marginal contributions and the transition of agents from one coalition to another was represented by a Markov process [122]. This model searched the coalitions space for the best structure using Markov probability distributions. Self-adapting coalition formation dynamically changed coalition memberships using electric grid-specific heuristics [49]. The benefits of these approaches include increased robustness to environment stochasticity and resilience to agent modeling errors by system designers. Unfortunately, a learning phase must be tolerated to allow the algorithm to learn through trial and error the agent models.

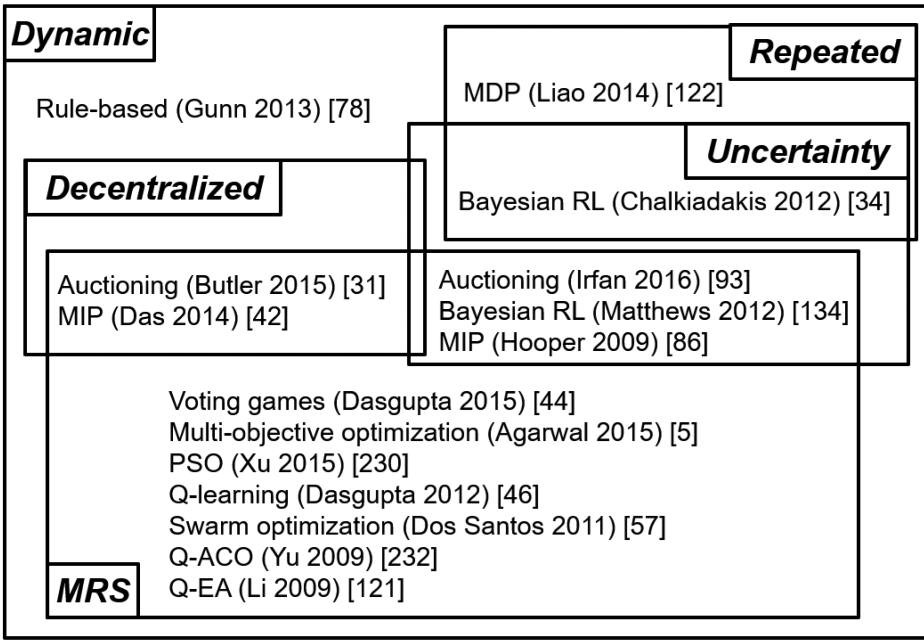

Fig. 4. Classification of coalition formation algorithms in the literature.

Focusing on MRS coalition formation, ASyMTRE facilitated coalition formation for tightly coupled tasks [158], where time and space constraints were added to coalition formation formulations [173]. Other approaches include swarm intelligence in RoboCup Rescue [57], ant colony optimization and genetic algorithm [169], decentralized ant colony optimization for modular robotics [170], and greedy approximate algorithms that converged in polynomial time [4]. Furthermore, the MuRoCo algorithm used market-based optimization to return optimal coalition formation for MRS, validated on a drink serving scenario [179]. Coalition pruning was proposed to form approximate coalitions in real time [126]. Modeling MRS coalition formation as a multi-objective optimization problem allowed the development of CUDA algorithms to speed up processing and was validated on navigation and box pushing tasks [5]. Decentralized optimal and sub-optimal coalition formation algorithms were also developed for UAVs based on particle swarm optimization and validated on UAV search and prosecute mission simulations [132]. Hierarchical optimization solvers searched for sub-optimal, computationally efficient coalitions to improve UAV search and prosecute which improved system responsiveness and scalability but negatively affected performance [238]. Finally, a multi-criteria decision-making algorithm was proposed based on influence diagrams to select the best coalition formation algorithm for a given real-world scenario [192]. Coalition formation algorithms were classified using domain and mission dependent features. Over 100,000 mission scenarios were developed under various conditions, including coalition overlap and communication constraints.

Figure 4 classifies the coalition formation algorithms presented above based on the assumptions they make, from assuming uncertainty in agent capabilities to implementing distributed systems or considering robot agents.

5.2 Task Allocation

Task allocation assigns tasks to an agent or a group of agents, i.e., it aims to find an optimal or near-optimal mapping between agents and tasks. A comprehensive taxonomy for task allocation can be found in Reference [109]. Many approaches have been adopted to produce such a mapping in MRS and have been surveyed in Reference [38, 94]. Auctioning or market-based approaches, common in multi-robot task allocation problems [33, 211], include ASyMTRE-D [158], Murdoch [71], M+ [28], and TraderBots [242].

Since MRS face additional constraints compared to MAS, including spatial, temporal, sensing and actuation constraints, MRS-specific task allocation algorithms have been developed. Examples include swarm intelligence [123], particle swarm optimization with graph theory for UAV military [148], Sandholm algorithm with K-means clustering [62], utility-based task allocation [3], max-sum [167], resource welfare [105], and semantic maps [69]. Many approaches mainly focused on minimizing the distance traveled by robots to complete tasks while ensuring that the capabilities required to successfully complete a task are possessed by the robot assigned to the task. However, some work has aimed to ensure workload balance between robots as well [62].

Auctioning algorithms are another class of algorithm popularly adopted in task allocation problems applied in health care facility scenarios capable of handling task priorities and heterogeneous tasks or robots [43], multi-robot traveling, UUV cooperation and object construction [189], and patrolling tasks [161]. They can be implemented in centralized, decentralized, or distributed topologies where the computations by the auctioneer and bidders can be either performed on a single node (centralized), multiple nodes (decentralized), or without an auctioneer at all (distributed). Since environments are dynamic and partially observable, task allocation algorithms should preempt and re-allocate tasks. Dynamical re-planning in MAS was achieved using an auction-based decentralized method [163]. Single-robot task preemption while minimizing unnecessary reallocation was developed using simultaneous descending auctioning [193]. However, repeated preempting could cause the agents to never complete their tasks causing the whole system to fail. Questions like when a task should be preempted, how often should preemption be considered a suitable choice, and others are important directions to investigate for effective task allocation in real-world environments.

Table 2 summarizes some of the work on task allocation (sorted in chronological order and published within the past 10 years). It compares the various assumptions they make such as assuming heterogeneous agents and virtual or robotic agents.

5.3 Simultaneous Coalition Formation and Task Allocation

Performing coalition formation and task allocation simultaneously has also been proposed in the literature to improve performance in MRS. IQ-ASyMTRE interleaved both problems to solve a broader range of complex tasks using MRS [233]. Combining task priority ranking and resource constraints improved coalition formation for UAVs [210]. Multi-robot task assignments in UAV search tasks were performed by combining dynamic ANT coalition formation and memetic local search task allocation algorithms based on robot, task, and environment information [153].

Auctioning algorithms were also adopted in simultaneous decentralized coalition formation and task allocation. Applications included two-robot box pushing and transportation, obstacle avoidance, and surveillance [195]. Auctioning allowed heterogeneous coalition formation with coalition re-adjustment before task completion to improve performance [93]. Constrained coalition formation and task allocation problems could also be solved using auction algorithms to minimize the time to complete tasks and the distance traveled by robot [42]. Such algorithms have been applied to reconfigurable robot coalition formation by splitting or merging robots while distributing tasks related to navigation, exploration, and surveillance sensors' placement [31].

While these approaches converge to better coalition and task allocations, they are computationally more expensive and require more iterations to converge to an assignment. However, their assignments improve with experience and lead to more efficient MAS and MRS task execution.

Table 2. Summary of Task Allocation Literature

Approach	Heterogeneous Robots	Topology	Preemption	Validation	Application
Consensus-based bundle algorithm [163]	Y	Decentralized	Y	Simulations, Experiments	UAV-UGV indoor navigation coordination
Auction [189]	Y	Distributed	N	Simulations, Experiments	MRS Navigation
Sandholm algorithm, K-means clustering [62]	Y	Centralized	N	Simulations	Navigation
Auction [161]	Y	Centralized	Y	Simulations, Experiments	Patrolling tasks
Auction [193]	Y	Decentralized	Y	Simulations	Abstracted tasks
Max-sum [167]	Y	Distributed	N	Simulations	Robocup rescue
Auction [43]	Y	Distributed	N	Simulations	MRS in healthcare facilities
Resource welfare [105]	Y	Distributed	N	Simulations	UAV target detection
Utility-based [3]	Y	Decentralized	N	Simulations, Experiments	UGV navigation for rescue operations
Particle swarm optimization, graph theory [148]	Y	Centralized	N	Simulations	UAV military tasks

6 MAS PLANNING AND CONTROL

MAS planning and control, also known as decision making, is a main module in MAS. It determines the sequence of actions, or policy, that agents should perform to complete their assigned task once the complex tasks have been decomposed to sub-tasks and allocated to cooperating groups of agents. Decision-making models have been applied to a wide spectrum of fields, including robotics [154], wireless sensor networks [113], cognitive radio networks [130], intelligent transportation systems [166], and electric grids [160]. Decision-making algorithms are generally evaluated based on policy optimality and their time and space complexity [184]. Multiple frameworks have been proposed to model and solve decision-making problems, including RL, game theory, swarm intelligence, and graph-theoretic models.

While more detailed surveys on multi-agent decision-making models can be found in References [177, 215], we will briefly cover some of the frameworks mentioned above. We will briefly cover some of the decision-making models below. More detailed surveys on multi-agent decision making can be found in References [177, 215]. Rizk et al. [177] surveys models swarm intelligence to game theory, discussing their advantages, disadvantages, applications, remaining challenges, and insights into possible future research directions. Torreno et al. surveyed MAS planning by formalizing the definition of cooperative tasks and surveying the main aspects of MAS planning solvers, including planning synthesis schemes, communication mechanisms, and privacy-preserving methodology [215].

Swarm intelligence, inspired by social animals, models the behavior of many decentralized cooperative autonomous agents [24]. Such models are mainly characterized by self-organized and distributed behavior of locally aware and locally interacting agents [226]. Particle swarm optimization [100] is inspired by flocks of birds and schools of fish. Pigeon-inspired optimization algorithm rely on the magnetic field, sun, and landmarks to achieve path planning [58]. Bee colony

Table 3. Comparison of Decision-making Models

Model	Degree of Scalability	Degree of Heterogeneity	Degree of Communication
Swarm Intelligence	High	Low	Low
Multi-agent MDP	Medium	Medium	Medium
Decentralized MDP	Medium	Medium	Medium
Multi-agent POMDP	Medium	Medium	High
Decentralized POMDP	Medium	Medium	High
Interactive POMDP	Low	Medium	High
Partially Observable Stochastic Games	Low	High	Medium

optimization is based on the behavior of bees and relies on direct communication between agents [214]. While swarm-based systems are robust, flexible, scalable, computationally inexpensive, and fault tolerant [29, 55], robots are generally homogeneous or can be divided into a small number of clusters of homogeneous robots, which greatly restricts MRS applications [55]. More details on swarm robotics can be found in these recent surveys [19, 21, 29, 154].

Game-theoretic models include partially observable stochastic games, which are sequential probabilistic games where payoffs are unknown to players and depend on their actions, and the state of the game depends on the previous state and the players' actions [194]. Their sub-classes include MDPs and partially observable MDPs (POMDPs). MDPs assume that decisions satisfy the Markov property, i.e., decisions at the current time step only depend on decisions of the previous time step. MDPs are described by a set of environment states that are fully observable: actions, transition probabilities, rewards, and discount factors. A policy maps states to actions by maximizing the total reward, measured by a value function. Multi-agent MDP and decentralized MDP extended MDP to MAS, which assumes jointly fully observable environments [168]. POMDPs extend MDPs to partially observable environments by adding a set of observations and an observation function to the model representation. MAS extensions include decentralized POMDP [9], multi-agent POMDP [10], and interactive POMDP [73]. Recent work on decentralized POMDP focused on the sub-classes of MDP models and their computational complexity briefly mentioned existing solutions [9].

Table 3 compares the decision-making models based on the degree of scalability of the models, the degree of agent heterogeneity within a MAS, and the degree of communication required among agents. We can see that swarm intelligence has the highest scalability capabilities due to its low communication requirements but does not allow for significant heterogeneity in agent capabilities [177]. The tradeoff between scalability and heterogeneity is evident in each model. Furthermore, increased heterogeneity leads to an increase in the need for communication that reduces the system's ability to scale well.

RL allows agents to learn a policy by rewarding "good" behavior and punishing "bad" behavior through a reward signal. RL is one approach to find optimal or sub-optimal policies for game-theoretic models. Incorporating RL to robotic systems requires additional considerations due to the physical constraints imposed by real-world environments and has been studied in Reference [108]. Multi-agent RL allows cooperative MAS to complete tasks with minimal communication overhead by using the global immediate reward instead of the individual agent immediate reward in the Q-learning algorithm to solve repeated games [234]. Validation on box pushing and sensor distribution demonstrated the superior performance of this algorithm compared to other approached. Sparse interaction to negotiate equilibrium sets and transfer knowledge in multi-agent RL reduced computational complexity and led to better coordination and scalability, as shown by simulations on grid world games and robots shelving items in a warehouse [240]. Information sharing has been modeled as a MDP to reduce communication overhead without affecting performance [12].

Sub-optimal policies for decentralized POMDP were computed using a factored forward-sweep policy computation algorithm that reduced computational complexity and improved scalability [149]. Simulations on hundreds of agents showed the improved scalability with minimal loss in accuracy. An RL inference model that learned MRS configurations and allowed robot systems to complete a task knowing the intermediary robot states and transitions was also developed [206]. Other approaches include dynamic programming [168], expectation maximization [202], heuristic search algorithms [203], temporal difference learning [63], policy search [26], evolutionary computing [227], genetic algorithms [61], neural networks [82], optimization algorithms [59], Monte Carlo methods [197] and deep RL approaches [138].

7 PERCEPTION

Perception is a crucial component of successful MRS deployment that allows robots to model their environment from sensory information and obtain knowledge of how their actions are affecting the environment and whether they are successfully completing their tasks; it is a sub-block of task execution in Figure 2. Without this capability, task execution would be nearly impossible in real-world environments. Sensors measure variables in the environment, allowing robots to observe how their actions have affected the environment, which leads to more effective task execution. From the low-level sensory information, robots need to learn higher-level information, such as the location obstacles, their locations within a map, and the types of objects in an environment, among others.

SLAM allows robots to simultaneously generate a map of the environment and localize themselves within this map, a vital aspect of any task that involves navigation [18, 60]. Many algorithms have been proposed with varying degrees of computational complexity [51], using a wide range of sensors including cameras [68], acoustic sensors [225], structured light [144], and electromagnetic signals [137]. SLAM has also adopted sensor fusion [188] to benefit from diverse signals such sonars with laser range finders [187], WiFi, Bluetooth, LTE, and magnetic signals [137]. Distributed, decentralized, cooperative, or multi-robot SLAM has been developed to leverage multi-robot cooperation and tackle complex environments [75], in communication constrained environments [159], or where direct communication is not possible [106]. Distributed SLAM with sparse robot networks [117] and decentralized active SLAM that forced robot teams to efficiently traverse and map the environment [16] were also investigated.

Scene understanding allows robots to extract general principles from visual cues [79]. It includes computer vision problems of image segmentation, object recognition, event recognition, human activity and behavior recognition [219], semantic annotation [239], and others. Scene understanding has been applied to pedestrian [127], traffic [201], urban [70], video surveillance [103], and underwater scenes [140]. Multi-agent or distributed computer vision algorithms have been developed to improve scene understanding in MAS applications [171]. Deep neural networks performed multi-object classification and scene understanding by extracting features from raw images and incorporating context through conditional latent tree probability models [146]. Markov random fields performed modeling, inference, and learning tasks on visual inputs by representing the inputs' conditional probabilistic dependence with undirected graphs [221]. Time-dependent correlation rules were adopted for dynamic scene understanding in traffic surveillance problems where motion patterns were detected using object tracking, spectral clustering, and Allen's interval-based temporal logic [208]. Traffic patterns were learned using hierarchical pattern mining based on latent Dirichlet allocation; traffic states were learned from activities that were modeled from spatial location and velocity [201].

Object motion tracking is another important aspect of scene understanding that helps robots achieve their goals by tracking objects of interest. Developed systems were based on received signal strength variations on wireless links [228] and Kalman filter-based SLAM with laser-based occupancy grids [98] to name a few. Multi-robot or cooperative tracking [37] has been developed to track pedestrians [217] and other objects, using particle filters [6], RL [124], and least-squares minimization [7]. More information can be found in the recent survey on MRS object detection and tracking [178].

Automatic speech recognition is important in human–robot interactions [84]. Approaches include Hidden Markov models [95], deep neural networks [2], and support vector machines [200]. While deep neural networks have had the best performance to date, they are computationally expensive and require many data points to achieve good performance, making integration with robotic systems for real-time applications a challenge. Some work has attempted to address the computational complexity of automatic speech recognition [209], but there is still a lot of room for improvement. Algorithms that are robust to noise have been proposed to handle noisy environments [119]. In addition, distant talking [74] and the noise from a robot's hardware [91] add to the difficulty of deploying speech recognition algorithms in MRS.

Many systems have combined two or more of the above functionalities into robotic systems. SLAM and scene understanding have been combined within a unified system; for example, Li et al. [118] performed visual dense SLAM with probabilistic semantic inference and object pose estimation. SLAM and object motion tracking were combined in the robotic surveillance system in References [7, 52]. Multi-modal human–machine interaction through audio and visual processing has also been investigated in the literature to improve human–machine communication [135]. Combining all these capabilities into one end-to-end system has generally been investigated in the field of cognitive architecture design [110]. Embodied cognitive architectures, i.e., those that have been embedded in robots, include ACT-R/E [216] and CRAM [22]. These systems have been mainly tested in restricted or simplified environments. End-to-end systems combining all these functionalities have been successfully integrated into autonomous vehicles deployed in the real world but have not seen widespread success in MRS. In summary, sophisticated multi-modal perception modules have seen significant improvements to the state of the art in the past few decades that will help MRS make better decisions in real-world environments.

8 CHALLENGES AND INSIGHTS

Allowing heterogeneous agents to cooperate increases the scope of solvable tasks. It introduces parallelism and robustness, leading to better performance with simpler agents compared to having a single powerful but complex agent performing the same task [157]. However, it also increases the complexity of the design process. Many challenges still face the research community before effective deployment of MRS performing complex tasks can be achieved. In this section, we discuss some challenges faced in designing MRS and some insights on future research directions based on the research areas identified in Figure 2.

8.1 Big Data

Perception problems, including object recognition, speech recognition, and natural language processing, have greatly benefited from hardware advancements that allowed machine-learning algorithms to develop models from big data. However, cloud accessibility may be an issue in some robotics applications, and perception algorithms deployed on robot platform do not have the computational resources to leverage these advancements and improve the robots' models of their environments. In addition, decision-making algorithms would greatly benefit from the developed models to improve decisions. However, these models are computationally expensive for robotics applications even if they are trained offline; the most successful models are derived from deep neural networks, which require gigabytes of memory to store the models and millions of operations to propagate the input through the network and obtain an output. Therefore, future research should investigate methods to incorporate big data models into computationally constrained and communication restricted MRS applications to improve task planning and execution.

Deep learning compression [81], deep learning software [114] and hardware accelerators [220], and neurosynaptic processors like IBM's TrueNorth chip that can efficiently train neural networks [8, 191] are research directions that can help MRS leverage big data. Edge computing, which allows network edge devices to perform some computations instead of sending them to the cloud [190], is another research area that could provide some solutions when cloud accessibility due to communication constraints is an issue for MRS.

8.2 Internet of Things

While state-of-the-art algorithms in perception and scene understanding have seen significant improvements, out-performing humans in some scenarios, integration with robotics applications is still in its early stages. Furthermore, there is still plenty of information to extract from the environment, especially in this era of IoT. Sensor fusion and distributed sensing from heterogeneous sources are two areas that can help improve perception for robotics applications.

Distributed deep neural networks are one possible solution to sensor fusion and influencing robots' decision making [48, 96]. The challenges of IoT-aided robotics have been discussed in Reference [77], which included security, communication, consensus, and information flow. Ray et al. proposed combining IoT and robotics into a unified architecture, termed Internet of robotic things [174], to address some of these challenges by allowing seamless cooperation between the various "things." Applications in health care, wearable robotics design such as prosthetics and exoskeletons [147], assistive social robots [199], rescue operations, and others would greatly benefit from the concept of Internet of robotic things.

8.3 Task Complexity

As tasks become more complex, decision-making algorithms struggle to recognize their complexity and decompose them to simpler tasks that can be solved efficiently. To aid MRS in completing complex tasks in uncertain environments, the task decomposition step should be automated to allow re-planning as conditions change. Furthermore, automated task decomposition could make use of existing ontology and domain-specific dictionaries in natural language processing to decompose tasks to sub-tasks. The decomposition could consider the available agents' capabilities and the model of the environment, beyond the workflow depicted in Figure 2. Crowdsourcing and contacting human operators at call centers could also be leveraged to assist robots in decomposing complex tasks [99]. Reinforcement learning frameworks incorporating this human assistance can be developed to improve the accuracy of fully autonomous decomposition. Furthermore, CrowdPhysics, which allows tasks that must be completed in the physical world to be distributed among humans that need to "collaborate and synchronize in both time and physical space" to successfully complete tasks [186] and similar platforms can be used as training data for MRS to learn how complex tasks are distributed among multiple agents.

8.4 Autonomous Machine Learning

Many machine-learning algorithms still rely on human intervention to manually tune algorithm parameters. Autonomous machine learning (AutoML) is a machine-learning sub-field striving to develop learning algorithms that do not require a human expert to select a learning algorithm, manually tune parameters, or select data for training [80, 182, 183, 207]. Incorporating AutoML into MRS would result in general agents that can better handle dynamic environments. Bayesian optimization has been commonly adopted to eliminate human intervention and further improved by incorporating a meta-learning phase that looks to similar problems that had been previously learned to guide its learning into the AutoML workflow [67]. Randomized neural networks that have fewer tunable hyper-parameters while preserving the algorithm's responsiveness are one possible framework for AutoML. Future research should also consider applying or extending AutoML algorithms developed for mobile devices [83] to MRS applications. Randomized non-iterative neural networks are one possible solution for AutoML, because they do not contain as many tunable hyper-parameters as iterative neural networks [176]. Furthermore, they have a low computational profile, making them suitable for mobile devices, and they have found success in control and time-sensitive applications encountered by MRS.

8.5 Scalability and Heterogeneity Tradeoff

To effectively operate in smart cities, MRS need to be scalable, adaptable, and generalizable to successfully cope with the dynamic environment and complexity of their tasks. Having multiple robots act on the environment simultaneously further increases the uncertainty in the system. Many decentralized MRS planning and control algorithms have been proposed in the literature but still face challenges when dealing with the tradeoff between scalability and robot heterogeneity in highly dynamic environments. Therefore, developing efficient planning algorithms that strikes a task-appropriate balance between scalability and heterogeneity will take MRS a step closer to more ubiquitous existence in smart cities. Hierarchical approaches where local interactions are dense and global interactions are sparse could be adopted to improve scalability while allowing agent heterogeneity.

8.6 Coalition Formation and Task Allocation

Simultaneous coalition formation and task allocation could lead to more optimal mappings and should be investigated further, since only a few works have considered this approach but obtained promising results [233]. Furthermore, the coalition and task assignments should be dynamic and time variant to better cope with task complexity and environment variability. Therefore, coalitions might have to be dynamically altered and assigned new tasks before the completion of their assigned tasks to achieve successful task execution. Coalition formation and task allocation algorithms that allow repeated, dynamic coalition formation and task exemption have been developed but still face limitations, especially in highly dynamic environments. They should also consider the tradeoff between agent capabilities' redundancy within a coalition and fault tolerance or robustness to agent failures. While many approaches for MAS have been developed, less work has considered the physical constraints imposed by MRS on coalitions. Future work should consider real-world experimental results to guide their efforts in realizing a coalition formation and task allocation algorithm capable of forming effective and efficient robot teams in real-world environments that successfully complete their assigned tasks.

8.7 Human-in-the-Loop

The term *human-in-the-loop* refers to a system architecture that requires robots or agents to interact with humans. Humans' involvement can range from giving agents instructions to executing actions alongside other robots and tele-operating vehicles. The benefits of such systems include expanding the scope of tasks MRS can execute without achieving full autonomy, complementing the skills of MRS with those of humans to efficiently execute certain tasks, giving humans more control over the system and improving system adaptability and resistance to environmental stochasticity. However, this system is also faced with many challenges from communication to responsiveness and fault tolerance. Communicating with agents via text requires the agents to perform natural language processing to parse messages and text generation to send messages. Communicating via speech imposes additional complexity by requiring the integrations of speech recognition and synthesis modules. While these fields have seen significant strides in recent years with the emergence of deep learning and big data, embodied by systems like Siri, Amazon Alexa, and others [129], accuracy, computational complexity, and responsiveness are still issues that need improvements before incorporating them into more complex systems with coordination, synchronization, and time constraints [89]. The responsiveness of the system will suffer when incorporating humans in the loop due to the additional communication overhead and human performance variability. Finally, such systems may be more error prone and could be affected by human fatigue and distractions. Therefore, determining whether human-in-the-loop is an asset or liability in a given scenario is key to choosing the right system architecture that would lead to the successful completion of tasks.

8.8 Transfer Learning

The highly stochastic nature of real-world environments and the variability in agent capabilities within heterogeneous MRS imply that a given MRS will rarely encounter two identical scenarios. However, MRS will encounter many similar scenarios. Therefore, leveraging previous experiences to improve current decision making would significantly improve the performance, adaptability, and robustness of MRS. Transfer learning is a learning paradigm that allows agents to jump-start their learning by transferring knowledge from previous experiences to current reinforcement learning problems instead of learning from scratch during every new scenario [213]. Transfer learning has been applied to MAS [151, 212] and MRS [50] but has yet to be tested in end-to-end systems deployed in the real world on complex tasks. Furthermore, transfer learning for RL has been mainly tested on benchmark problems and gaming environments but not in real-world environments. Therefore, this field of research, which started gaining traction less than two decades ago [213], still has many open problems that must be addressed before successful integration into MRS in the real world can be realized.

8.9 Unified Framework

Significant contributions have been made to the various modules in Figure 2. However, one reason that has hindered the successful deployment of fully automated MRS is the fact that most of the work has mainly dealt with these modules independently. Taking a more holistic approach by viewing these research fields as connected within a larger field is necessary to take steps toward successful MRS deployments. Furthermore, feedback connections between the different modules should be incorporated into the workflow to further improve its efficacy. For example, task decomposition should be influenced by the capabilities of the agents and coalitions, as well as the environment's state estimated by the perception module and vice versa. End-to-end simulations and real-world experiments will also help in improving the formulated models by identifying major weaknesses preventing successful deployment.

8.10 Other Challenges

Communication constraints and connectivity uncertainty further complicate things for cooperative MRS, especially for tightly coordinated problems. While connecting MRS to the cloud also allows us to reduce the computational load on these mobile devices and improve their performance [222], the existence and stability of this connection is uncertain and may sometimes cripple the system instead of improving its performance. The time sensitivity of certain tasks and limited hardware resources of robots requires the development of efficient algorithms for decision making, perception, coalition formation, and task decomposition and allocation. Finally, evaluation standards are needed to effectively compare the performance of MRS, as they are still underdeveloped.

9 CONCLUSION

In this survey, we presented an overview of MRS that consist of robots with diversity in sensing, actuation, and computational capabilities. We defined the terminology commonly used in the field and identified the main components of a workflow that attempts to minimize human intervention with MRS team formation, task decomposition, task allocation, robot perception, and planning and control. Then, existing MRS were surveyed and classified based on the degree of human intervention necessary. While many MRS have been proposed in the literature, most of these systems still require the system designer to decompose complex tasks to simpler sub-tasks and assign tasks to sub-groups or build robot coalitions with complementary skills. Finally, we surveyed existing work in each sub-field and discussed some of the challenges faced in each area as well as the overall system design.

REFERENCES

[1] Sameera Abar, Georgios K. Theodoropoulos, Pierre Lemarinier, and Gregory M. P. O'Hare. 2017. Agent based modelling and simulation tools: A review of the state-of-art software. *Comput. Sci. Rev.* 24 (May 2017), 13–33.

[2] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. 2014. Convolutional neural networks for speech recognition. *IEEE/ACM Trans. Aud. Speech Lang. Process.* 22, 10 (2014), 1533–1545.

[3] Tamer Abukhalil, Madhav Patil, Sarosh Patel, and Tarek Sobh. 2016. Coordinating a heterogeneous robot swarm using robot utility-based task assignment (RUTA). In *Proceedings of the IEEE 14th International Workshop on Advanced Motion Control.* 57–62.

[4] Julie A. Adams et al. 2011. Coalition formation for task allocation: Theory and algorithms. *Auton. Agents Multi-Agent Syst.* 22, 2 (2011), 225–248.

[5] Manoj Agarwal, Nitin Agrawal, Shikhar Sharma, Lovekesh Vig, and Naveen Kumar. 2015. Parallel multi-objective multi-robot coalition formation. *Expert Syst. Applicat.* 42, 21 (2015), 7797–7811.

[6] Aamir Ahmad and Pedro Lima. 2013. Multi-robot cooperative spherical-object tracking in 3D space based on particle filters. *Robot. Auton. Syst.* 61, 10 (2013), 1084–1093.

[7] Aamir Ahmad, Gian Diego Tipaldi, Pedro Lima, and Wolfram Burgard. 2013. Cooperative robot localization and target tracking based on least squares minimization. In *Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA'13).* IEEE, 5696–5701.

[8] Filipp Akopyan, Jun Sawada, Andrew Cassidy, Rodrigo Alvarez-Icaza, John Arthur, Paul Merolla, Nabil Imam, Yutaka Nakamura, Pallab Datta, Gi-Joon Nam, et al. 2015. Truenorth: Design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip. *IEEE Trans. Comput.-Aid. Des. Integr. Circ. Syst.* 34, 10 (2015), 1537–1557.

[9] Christopher Amato, Girish Chowdhary, Alborz Geramifard, N. Kemal Ure, and Mykel J. Kochenderfer. 2013. Decentralized control of partially observable Markov decision processes. In *Proceedings of the IEEE 52nd Annual Conference on Decision and Control.* 2398–2405.

[10] Christopher Amato, Frans A. Oliehoek, and Eric Shyu. 2013. Scalable bayesian reinforcement learning for multiagent POMDPs. In *Proceedings of the 1st Multidisciplinary Conference on Reinforcement Learning and Decision Making.*

[11] Ofra Amir, Barbara J. Grosz, Edith Law, and Roni Stern. 2013. Collaborative health care plan support. In *Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems.* International Foundation for Autonomous Agents & Multiagent Systems, 793–796.

[12] Ofra Amir, Barbara J. Grosz, and Roni Stern. 2014. To share or not to share? The single agent in a team decision problem. In *Proceedings of the 28th AAAI Conf. Artificial Intelligence* (2014), 3092–3093.

[13] Tamio Arai, Enrico Pagello, and Lynne E. Parker. 2002. Editorial: Advances in multi-robot systems. *IEEE Trans. Robot. Autom.* 18, 5 (2002), 655–661.

[14] Nordmann Arne, Hochgeschwender Nico, Wigand Dennis, and Wrede Sebastian. 2016. A survey on domain-specific modeling and languages in robotics. *J. Softw. Eng. Robot.* 7, 1 (2016), 75–99.

[15] Omur Arslan, Dan P. Guralnik, and Daniel E. Koditschek. 2016. Coordinated robot navigation via hierarchical clustering. *IEEE Trans. Robot.* 32, 2 (2016), 352–371.

[16] Nikolay Atanasov, Jerome Le Ny, Kostas Daniilidis, and George J. Pappas. 2015. Decentralized active information acquisition: Theory and application to multi-robot SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation.* 4775–4782.

[17] José Baca, Prithvi Pagala, Claudio Rossi, and Manuel Ferre. 2015. Modular robot systems towards the execution of cooperative tasks in large facilities. *Robot. Auton. Syst.* 66 (2015), 159–174.

[18] Tim Bailey and Hugh Durrant-Whyte. 2006. Simultaneous localization and mapping (SLAM): Part II. *IEEE Robot. Autom. Mag.* 13, 3 (2006), 108–117.

[19] Jan Carlo Barca and Y. Ahmet Sekercioglu. 2013. Swarm robotics reviewed. *Robotica* 31, 3 (2013), 345–359.

[20] Samuel Barrett and Peter Stone. 2015. Cooperating with unknown teammates in complex domains: A robot soccer case study of ad hoc teamwork. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'15).* 2010–2016.

[21] Levent Bayındır. 2016. A review of swarm robotics tasks. *Neurocomputing* 172 (2016), 292–321.

[22] Michael Beetz, Lorenz Mösenlechner, and Moritz Tenorth. 2010. CRAM: A cognitive robot abstract machine for everyday manipulation in human environments. In *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'10).* IEEE, 1012–1017.

[23] Patrick Benavidez, Mohan Kumar, Sos Agaian, and Mo Jamshidi. 2015. Design of a home multi-robot system for the elderly and disabled. In *Proceedings of the 10th Annual Systems of Engineering Conference.* IEEE, 392–397.

[24] Gerardo Beni and Jing Wang. 1993. Swarm intelligence in cellular robotic systems. In *Robots and Biological Systems: Towards a New Bionics?* Springer, 703–712.

[25] Ahmed Benzerrouk, Lounis Adouane, and Philippe Martinet. 2014. Stable navigation in formation for a multi-robot system based on a constrained virtual structure. *Robot. Auton. Syst.* 62, 12 (2014), 1806–1815.

[26] Daniel S. Bernstein, Christopher Amato, Eric A. Hansen, and Shlomo Zilberstein. 2009. Policy iteration for decentralized control of Markov decision processes. *J. Artif. Intell. Res.* 34, 1 (2009), 89.

[27] A. H. Bond and L. Gasser. 1988. Readings in Distributed Artificial Intelligence. Morgan Kaufmann.

[28] Sylvia C. Botelho and Rachid Alami. 1999. M+: A scheme for multi-robot cooperation through negotiated task allocation and achievement. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 2. 1234–1239.

[29] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. 2013. Swarm robotics: A review from the swarm engineering perspective. *Swarm Intell.* 7, 1 (2013), 1–41.

[30] Luis Búrdalo, Andrés Terrasa, Vicente Julián, and Ana García-Fornes. 2018. The information flow problem in multi-agent systems. *Eng. Appl. Artif. Intell.* 70 (April 2018), 130–141.

[31] Zack Butler and Jacob Hays. 2015. Task allocation for reconfigurable teams. *Robot. Auton. Syst.* 68 (June 2015), 59–71.

[32] Philippe Caloud, Wonyun Choi, Jean-Claude Latombe, Claude Le Pape, and Mark Yim. 1990. Indoor automation with many mobile robots. In *Proceedings of the IEEE International Workshop on Intelligent Robots and Systems: Towards a New Frontier of Applications.* 67–72.

[33] Adam Campbell and Annie S. Wu. 2011. Multi-agent role allocation: Issues, approaches, and multiple perspectives. *Auton. Agents Multi-agent Syst.* 22, 2 (2011), 317–355.

[34] Georgios Chalkiadakis and Craig Boutilier. 2012. Sequentially optimal repeated coalition formation under uncertainty. *Auton. Agents Multi-Agent Syst.* 24, 3 (2012), 441–484.

[35] Praneel Chand and Dale A. Carnegie. 2013. Mapping and exploration in a hierarchical heterogeneous multi-robot system using limited capability robots. *Robot. Auton. Syst.* 61, 6 (2013), 565–579.

[36] Jianping Chen, Yimin Yang, and Liang Wei. 2010. Research on the approach of task decomposition in soccer robot system. In *Proceedings of the IEEE International Conference on Digital Manufacturing and Automation*, Vol. 2. 284–289.

[37] Chen Tun Chou, Jiun-Yi Li, Ming-Fang Chang, and Li Chen Fu. 2011. Multi-robot cooperation based human tracking system using laser range finder. In *Proceedings of the International Conference on Robotics and Automation.* IEEE, 532–537.

[38] B. B. Choudhury and B. B. Biswal. 2008. Task allocation methodologies for multi-robot systems. *IEEE Sponsored Conf. Computational Intelligence, Control & Computer Vision in Robotics & Automation* (2008), 99–106.

[39] CNBC. 2016. *An Internet of Things That Will number Ten Billions.* Retrieved from http://www.cnbc.com/2016/02/01/an-internet-of-things-that-will-number-ten-billions.html.

[40] Luis C. Cobo, Charles L. Isbell Jr, and Andrea L. Thomaz. 2012. Automatic task decomposition and state abstraction from demonstration. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems,* Vol. 1. 483–490.

[41] Xuefeng Dai, Laihao Jiang, and Yan Zhao. 2016. Cooperative exploration based on supervisory control of multi-robot systems. *Appl. Intell.* 45, 1 (2016), 1–12.

[42] Gautham P. Das, Thomas Martin McGinnity, and Sonya A. Coleman. 2014. Simultaneous allocations of multiple tightly-coupled multi-robot tasks to coalitions of heterogeneous robots. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics.* 1198–1204.

[43] Gautham P. Das, Thomas M. McGinnity, Sonya A. Coleman, and Laxmidhar Behera. 2015. A distributed task allocation algorithm for a multi-robot system in healthcare facilities. *J. Intell. Robot. Syst.* 80, 1 (2015), 33–58.
[44] Prithviraj Dasgupta, José Baca, K. R. Guruprasad, Angélica Muñoz-Meléndez, and Janyl Jumadinova. 2015. The COMRADE system for multirobot autonomous landmine detection in postconflict regions. *J. Robot.* 2015 (2015), 17 pages.
[45] Prithviraj Dasgupta and Ke Cheng. 2015. Dynamic multi-robot team reconfiguration using weighted voting games. *J. Exp. Theoret. Artif. Intell.* 28, 4 (2015), 1–22.
[46] Prithviraj Dasgupta, Ke Cheng, and Bikramjit Banerjee. 2012. Adaptive multi-robot team reconfiguration using a policy-reuse reinforcement learning approach. In *Advanced Agent Technology*. Springer, 330–345.
[47] Paul Davidsson, Stefan Johansson, and Mikael Svahnberg. 2005. Characterization and evaluation of multi-agent system architectural styles. In *Software Engineering for Multi-Agent Systems IV*. Springer, 179–188.
[48] Elias De Coninck, Tim Verbelen, Bert Vankeirsbilck, Steven Bohez, Sam Leroux, and Pieter Simoens. 2015. Dianne: Distributed artificial neural networks for the internet of things. In *Proceedings of the 2nd Workshop on Middleware for Context-Aware Applications in the IoT*. ACM, 19–24.
[49] Gabriel de O Ramos, Juan C. Burguillo Rial, and Ana L. C. Bazzan. 2013. Self-adapting coalition formation among electric vehicles in smart grids. In *Proceedings of the IEEE International Conference on Self-Adaptive and Self-Organizing Systems*. 11–20.
[50] Coline Devin, Abhishek Gupta, Trevor Darrell, Pieter Abbeel, and Sergey Levine. 2017. Learning modular neural network policies for multi-task and multi-robot transfer. In *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA'17)*. IEEE, 2169–2176.
[51] Nitin Kumar Dhiman, Dipti Deodhare, and Deepak Khemani. 2015. Where am I? creating spatial awareness in unmanned ground robots using SLAM: A survey. *Sadhana* 40, 5 (2015), 1385–1433.
[52] Donato Di Paola, Annalisa Milella, Grazia Cicirelli, and Arcangelo Distante. 2010. An autonomous mobile robotic system for surveillance of indoor environments. *Int. J. Adv. Robot. Syst.* 7, 1 (2010), 8.
[53] M. Bernardine Dias. 2004. *Traderbots: A New Paradigm for Robust and Efficient Multirobot Coordination in Dynamic Environments*. Ph.D. Dissertation. Carnegie Mellon University, Pittsburgh.
[54] M. Bernardine Dias, Robert Zlot, Nidhi Kalra, and Anthony Stentz. 2006. Market-based multirobot coordination: A survey and analysis. *Proc. IEEE* 94, 7 (2006), 1257–1270.
[55] Marco Dorigo, Dario Floreano, Luca M. Gambardella, Francesco Mondada, Stefano Nolfi, Tarek Baaboura, Mauro Birattari, Michael Bonani, Manuele Brambilla, Arne Brutschy, et al. 2013. Swarmanoid: A novel concept for the study of heterogeneous robotic swarms. *IEEE Robot. Autom. Mag.* 20, 4 (2013), 60–71.
[56] Rajesh Doriya, Siddharth Mishra, and Swati Gupta. 2015. A brief survey and analysis of multi-robot communication and coordination. In *Proceedings of the 2015 International Conference on Computing, Communication & Automation (ICCCA'15)*. IEEE, 1014–1021.
[57] Fernando Dos Santos and Ana L. C. Bazzan. 2011. Towards efficient multiagent task allocation in the robocup rescue: A biologically-inspired approach. *Auton. Agents Multi-Agent Syst.* 22, 3 (2011), 465–486.
[58] Haibin Duan and Peixin Qiao. 2014. Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning. *Int. J. Intell. Comput. Cybernet.* 7, 1 (2014), 24–37.
[59] Francois Dufour and Tomas Prieto-Rumeau. 2013. Finite linear programming approximations of constrained discounted Markov decision processes. *SIAM J. Contr. Optim.* 51, 2 (2013), 1298–1324.
[60] Hugh Durrant-Whyte and Tim Bailey. 2006. Simultaneous localization and mapping: Part I. *IEEE Robot. Autom. Mag.* 13, 2 (2006), 99–110.
[61] Barış Eker and H. Levent Akın. 2013. Solving decentralized POMDP problems using genetic algorithms. *Auton. Agents Multi-agent Syst.* 27, 1 (2013), 161–196.
[62] Murugappan Elango, Ganesan Kanagaraj, and S. G. Ponnambalam. 2013. Sandholm algorithm with K-means clustering approach for multi-robot task allocation. In *Swarm, Evolutionary, and Memetic Computing*. Springer, 14–22.
[63] Pablo Escandell-Montero, José M. Martínez-Martínez, José D. Martín-Guerrero, Emilio Soria-Olivas, and Juan Gómez-Sanchis. 2014. Least-squares temporal difference learning based on an extreme learning machine. *Neurocomputing* 141 (Oct. 2014), 37–45.
[64] Alessandro Farinelli, Giorgio Grisetti, Luca Iocchi, S. Lo Cascio, and Daniele Nardi. 2003. Design and evaluation of multi agent systems for rescue operations. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, Vol. 4. 3138–3143.
[65] Jacques Ferber. 1999. *Multi-agent Systems: An Introduction to Distributed Artificial Intelligence*. Vol. 1. Addison-Wesley, Reading, MA.
[66] Borja Fernandez-Gauna, Ismael Etxeberria-Agiriano, and Manuel Graña. 2015. Learning multirobot hose transportation and deployment by distributed round-robin q-learning. *PLoS ONE* 10, 7 (2015), e0127129.
[67] Matthias Feurer, Aaron Klein, Katharina Eggensperger, Jost Springenberg, Manuel Blum, and Frank Hutter. 2015. Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems*. 2962–2970.

[68] Jorge Fuentes-Pacheco, José Ruiz-Ascencio, and Juan Manuel Rendón-Mancha. 2015. Visual simultaneous localization and mapping: A survey. *Artif. Intell. Rev.* 43, 1 (2015), 55–81.

[69] Cipriano Galindo, Juan-Antonio Fernández-Madrigal, Javier González, and Alessandro Saffiotti. 2008. Robot task planning using semantic maps. *Robot. Auton. Syst.* 56, 11 (2008), 955–966.

[70] Andreas Geiger, Martin Lauer, and Raquel Urtasun. 2011. A generative model for 3d urban scene understanding from movable platforms. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1945–1952.

[71] Brian P. Gerkey and Maja J. Matari. 2002. Sold!: Auction methods for multirobot coordination. *IEEE Trans. Robot. Autom.* 18, 5 (2002), 758–768.

[72] Brian P. Gerkey and Maja J. Matarić. 2004. A formal analysis and taxonomy of task allocation in multi-robot systems. *Int. J. Robot. Res.* 23, 9 (2004), 939–954.

[73] Piotr J. Gmytrasiewicz and Prashant Doshi. 2004. Interactive POMDPs: Properties and preliminary results. In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems,* Vol. 3. IEEE, 1374–1375.

[74] Randy Gomez, Tatsuya Kawahara, Keisuke Nakamura, and Kazuhiro Nakadai. 2012. Multi-party human-robot interaction with distant-talking speech recognition. In *Proceedings of the 7th Annual ACM/IEEE International Conference on Human–Robot Interaction.* 439–446.

[75] Bruno Duarte Gouveia, David Portugal, Daniel C. Silva, and Lino Marques. 2015. Computation sharing in distributed robotic systems: A case study on SLAM. *IEEE Trans. Autom. Sci. Eng.* 12, 2 (2015), 410–422.

[76] Jason Gregory, Jonathan Fink, Ethan Stump, Jeffrey Twigg, John Rogers, David Baran, Nicholas Fung, and Stuart Young. 2016. Application of multi-robot systems to disaster-relief scenarios with limited communication. In *Field and Service Robotics.* Springer, 639–653.

[77] Luigi Alfredo Grieco, Alessandro Rizzo, Simona Colucci, Sabrina Sicari, Giuseppe Piro, Donato Di Paola, and Gennaro Boggia. 2014. IoT-aided robotics applications: Technological implications, target domains and open issues. *Comput. Commun.* 54 (2014), 32–47.

[78] Tyler Gunn and John Anderson. 2013. Dynamic heterogeneous team formation for robotic urban search and rescue. *Proc. Comput. Sci.* 19 (2013), 22–31.

[79] Ruiqi Guo. 2014. *Scene Understanding with Complete Scenes and Structured Representations.* Ph.D. Dissertation. University of Illinois at Urbana–Champaign.

[80] Isabelle Guyon, Imad Chaabane, Hugo Jair Escalante, Sergio Escalera, Damir Jajetic, James Robert Lloyd, Núria Macià, Bisakha Ray, Lukasz Romaszko, Michèle Sebag, et al. 2016. A brief review of the chalearn automl challenge: Any-time any-dataset learning without human intervention. In *Proceedings of the Workshop on Automatic Machine Learning.* 21–30.

[81] Song Han, Huizi Mao, and William J. Dally. 2015. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *Int. Conf. Learning Representations (ICLR'16).* 1–14.

[82] Alexander Hans and Steffen Udluft. 2010. Ensembles of neural networks for robust reinforcement learning. In *Proceedings of the 9th International Conference on Machine Learning and Applications.* IEEE, 401–406.

[83] Yihui He, Ji Lin, Zhijian Liu, Hanrui Wang, Li-Jia Li, and Song Han. 2018. Amc: Automl for model compression and acceleration on mobile devices. In *Proceedings of the European Conference on Computer Vision (ECCV'18).* 784–800.

[84] Stefan Heinrich and Stefan Wermter. 2011. Towards robust speech recognition for human-robot interaction. In *Proceedings of the IROS Workshop on Cognitive Neuroscience Robotics.* 29–34.

[85] E. G. Hernandez-Martinez, E. D. Ferreira-Vazquez, A. Lopez-Gonzalez, J. J. Flores-Godoy, G. Fernandez-Anaya, and P. Paniagua-Contro. 2016. Formation control of heterogeneous robots using distance and orientation. In *Proceedings of the IEEE International Conference on Control Applications.* 507–512.

[86] Daylond J. Hooper, Gilbert L. Peterson, and Brett J. Borghetti. 2009. Dynamic coalition formation under uncertainty. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems.* 4799–4804.

[87] Andrew Howard, Lynne E. Parker, and Gaurav S. Sukhatme. 2006. Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection. *Int. J. Robot. Res.* 25, 5–6 (2006), 431–447.

[88] Tao Hu, Stefano Messelodi, and Oswald Lanz. 2014. Dynamic task decomposition for probabilistic tracking in complex scenes. In *Proceedings of the 22nd IEEE International Conference on Pattern Recognition.* 4134–4139.

[89] Ting-Hao Kenneth Huang, Walter S. Lasecki, Amos Azaria, and Jeffrey P. Bigham. 2016. "Is there anything else I can help you with?": Challenges in deploying an on-demand crowd-powered conversational agent. In *Proceedings of the 4th AAAI Conference on Human Computation and Crowdsourcing.*

[90] IBM. 2015. *1st Workshop on Cognitive Architectures.* Retrieved from http://researcher.watson.ibm.com/researcher/view_group.php?id=5848.

[91] Gökhan Ince, Kazuhiro Nakadai, Tobias Rodemann, Hiroshi Tsujino, and Jun-Ichi Imura. 2011. Whole body motion noise cancellation of a robot for improved automatic speech recognition. *Adv. Robot.* 25, 11–12 (2011), 1405–1426.

[92] Pablo Iñigo-Blasco, Fernando Diaz-del Rio, Ma Carmen Romero-Ternero, Daniel Cagigas-Muñiz, and Saturnino Vicente-Diaz. 2012. Robotics software frameworks for multi-agent robotic systems development. *Robot. Auton. Syst.* 60, 6 (2012), 803–821.
[93] Muhammad Irfan and Adil Farooq. 2016. Auction-based task allocation scheme for dynamic coalition formations in limited robotic swarms with heterogeneous capabilities. In *Proceedings of the IEEE International Conference on Intelligent Systems Engineering*. 210–215.
[94] Xiao Jia and Max Q.-H. Meng. 2013. A survey and analysis of task allocation algorithms in multi-robot systems. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics*. 2280–2285.
[95] Hui Jiang. 2010. Discriminative training of HMMs for automatic speech recognition: A survey. *Comput. Speech Lang.* 24, 4 (2010), 589–608.
[96] Jiong Jin, Jayavardhana Gubbi, Slaven Marusic, and Marimuthu Palaniswami. 2014. An information framework for creating a smart city through internet of things. *IEEE IoT J.* 1, 2 (2014), 112–121.
[97] E. Gil Jones, Brett Browning, M. Bernardine Dias, Brenna Argall, Manuela Veloso, and Anthony Stentz. 2006. Dynamically formed heterogeneous robot teams performing tightly-coordinated tasks. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 570–575.
[98] K. Kakinuma, Makoto Ozaki, Masafumi Hashimoto, T. Yokoyama, and Kazuhiko Takahashi. 2011. Laser-based pedestrian tracking with multiple mobile robots using outdoor SLAM. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics*. 998–1003.
[99] Ben Kehoe, Sachin Patil, Pieter Abbeel, and Ken Goldberg. 2015. A survey of research on cloud robotics and automation. *IEEE Trans. Autom. Sci. Eng.* 12, 2 (2015), 398–409.
[100] James Kennedy and Eberhart Russell C. 1995. Particle swarm optimization. In *Proceedings of the International Conference on Neural Networks*. IEEE Press, 1942–1948.
[101] Jutta Kiener and Oskar Von Stryk. 2010. Towards cooperation of heterogeneous, autonomous robots: A case study of humanoid and wheeled robots. *Robot. Auton. Syst.* 58, 7 (2010), 921–929.
[102] Hongkeun Kim, Hyungbo Shim, and Jin Heon Seo. 2011. Output consensus of heterogeneous uncertain linear multi-agent systems. *IEEE Trans. Autom. Contr.* 56, 1 (2011), 200–206.
[103] In Su Kim, Hong Seok Choi, Kwang Moo Yi, Jin Young Choi, and Seong G. Kong. 2010. Intelligent visual surveillance survey. *Int. J. Contr. Autom. Syst.* 8, 5 (2010), 926–939.
[104] Jonghoek Kim. 2016. Cooperative exploration and networking while preserving collision avoidance. *IEEE Trans. Cybern.* 47, 12 (2016), 4038–4048.
[105] Min-Hyuk Kim, Hyeoncheol Baik, and Seokcheon Lee. 2015. Resource welfare based task allocation for UAV team with resource constraints. *J. Intell. Robot. Syst.* 77, 3–4 (2015), 611–627.
[106] Alexander Kleiner and Dali Sun. 2007. Decentralized SLAM for pedestrians without direct communication. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1461–1466.
[107] Matthias Klusch and Andreas Gerber. 2002. Dynamic coalition formation among rational agents. *IEEE Intell. Syst.* 17, 3 (2002), 42–47.
[108] Jens Kober and Jan Peters. 2012. Reinforcement learning in robotics: A survey. In *Reinforcement Learning*. Springer, 579–610.
[109] G. Ayorkor Korsah, Anthony Stentz, and M. Bernardine Dias. 2013. A comprehensive taxonomy for multi-robot task allocation. *Int. J. Robot. Res.* 32, 12 (2013), 1495–1512.
[110] Iuliia Kotseruba, Oscar J. Avella Gonzalez, and John K. Tsotsos. 2016. A review of 40 years of cognitive architecture research: Focus on perception, attention, learning and applications. *arXiv preprint arXiv:1610.08602*, 1–74.
[111] Sarit Kraus. 2015. Intelligent agents for rehabilitation and care of disabled and chronic patients. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'15)*. 4032–4036.
[112] Sarit Kraus, Onn Shehory, and Gilad Taase. 2003. Coalition formation with uncertain heterogeneous information. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems*. ACM, 1–8.
[113] Pradeep Kumar Krishnappa and B. R. Prasad Babu. 2015. Investigating open issues in swarm intelligence for mitigating security threats in MANET. *Int. J. Elect. Comput. Eng.* 5, 5 (2015).
[114] Nicholas D. Lane, Sourav Bhattacharya, Petko Georgiev, Claudio Forlivesi, Lei Jiao, Lorena Qendro, and Fahim Kawsar. 2016. Deepx: A software accelerator for low-power deep learning inference on mobile devices. In *Proceedings of the 15th International Conference on Information Processing in Sensor Networks*. IEEE Press, 23.
[115] Pat Langley, John E. Laird, and Seth Rogers. 2009. Cognitive architectures: Research issues and challenges. *Cogn. Syst. Res.* 10, 2 (2009), 141–160.
[116] Charles Lesire, Guillaume Infantes, Thibault Gateau, and Magali Barbier. 2016. A distributed architecture for supervision of autonomous multi-robot missions. *Auton. Robots* 40, 7 (2016), 1343–1362.
[117] Keith Y. K. Leung, Timothy D. Barfoot, and Hugh H. T. Liu. 2010. Decentralized cooperative simultaneous localization and mapping for dynamic and sparse robot networks. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 3554–3561.

[118] Chi Li, Han Xiao, Keisuke Tateno, Federico Tombari, Nassir Navab, and Gregory D. Hager. 2016. Incremental scene understanding on dense SLAM. In *Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'16)*. IEEE, 574–581.

[119] Jinyu Li, Li Deng, Yifan Gong, and Reinhold Haeb-Umbach. 2014. An overview of noise-robust automatic speech recognition. *IEEE/ACM Trans. Aud. Speech Lang. Process.* 22, 4 (2014), 745–777.

[120] Xiong Li, Sheng Dang, Kunju Li, and Qiansheng Liu. 2010. Multi-agent-based battlefield reconnaissance simulation by novel task decomposition and allocation. In *Proceedings of the 5th IEEE International Conference on Computer Science and Education*. 1410–1414.

[121] Zhiyong Li, Bo Xu, Lei Yang, Jun Chen, and Kenli Li. 2009. Quantum evolutionary algorithm for multi-robot coalition formation. In *Proceedings of the 1st ACM/SIGEVO Summit on Genetic and Evolutionary Computation*. 295–302.

[122] Stephen Shaoyi Liao, Jia-Dong Zhang, Raymond Lau, and Tianying Wu. 2014. Coalition formation based on marginal contributions and the markov process. *Decis. Supp. Syst.* 57 (Jan. 2014), 355–363.

[123] S. H. Liu, Yu Zhang, H. Y. Wu, and Jie Liu. 2010. Multi-robot task allocation based on swarm intelligence. *J. Jilin Univ.* 40, 1 (2010), 123–129.

[124] Zheng Liu, Marcelo H. Ang Jr, and Winston Khoon Guan Seah. 2005. Reinforcement learning of cooperative behaviors for multi-robot tracking of multiple moving targets. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1289–1294.

[125] Z. Liu, W. Chen, J. Lu, H. Wang, and J. Wang. 2016. Formation control of mobile robots using distributed controller with sampled-data and communication delays. *IEEE Trans. Contr. Syst. Technol.* 24, 6 (2016), 2125–2132.

[126] Zhong Liu, Xiao-guang Gao, and Xiao-wei Fu. 2016. Coalition formation for multiple heterogeneous UAVs cooperative search and prosecute with communication constraints. In *Proceedings of the Chinese Control & Decision Conference*. IEEE, 1727–1734.

[127] D. F. Llorca, M. A. Sotelo, A. M. Hellín, A. Orellana, M. Gavilán, I. G. Daza, and A. G. Lorente. 2012. Stereo regions-of-interest selection for pedestrian protection: A survey. *Transport. Res. C: Emerg. Technol.* 25 (2012), 226–237.

[128] Matt Luckcuck, Marie Farrell, Louise Dennis, Clare Dixon, and Michael Fisher. 2018. Formal specification and verification of autonomous robotic systems: A survey. *Arxiv Preprint Arxiv:1807.00048* (2018).

[129] Ewa Luger and Abigail Sellen. 2016. Like having a really bad PA: The gulf between user expectation and experience of conversational agents. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 5286–5297.

[130] Jarmo Lunden, Sanjeev R. Kulkarni, Visa Koivunen, and H. Vincent Poor. 2013. Multiagent reinforcement learning based spectrum sensing policies for cognitive radio networks. *IEEE J. Sel. Topics Signal Process.* 7, 5 (2013), 858–868.

[131] Damian M. Lyons, Michael Arbib, et al. 1989. A formal model of computation for sensory-based robotics. *IEEE Trans. Robot. Autom.* 5, 3 (1989), 280–293.

[132] Joel G. Manathara, P. B. Sujit, and Randal W. Beard. 2011. Multiple UAV coalitions for a search and prosecute mission. *J. Intell. Robot. Syst.* 62, 1 (2011), 125–158.

[133] Neil Mathew, Stephen L. Smith, and Steven L. Waslander. 2015. Planning paths for package delivery in heterogeneous multirobot teams. *IEEE Trans. Autom. Sci. Eng.* 12, 4 (2015), 1298–1308.

[134] Tim Matthews, Sarvapali Ramchurn, and Georgios Chalkiadakis. 2012. Competing with humans at fantasy football: Team formation in large partially-observable domains. In *Proceedings of the 26th AAAI Conf. Artificial Intelligence* (2012), 1394–1400.

[135] Nikolaos Mavridis. 2015. A review of verbal and non-verbal human–robot interactive communication. *Robot. Auton. Syst.* 63 (2015), 22–35.

[136] Kathryn Merrick. 2017. Value systems for developmental cognitive robotics: A survey. *Cogn. Syst. Res.* 41 (2017), 38–55.

[137] Piotr Mirowski, Tin Kam Ho, Saehoon Yi, and Michael MacDonald. 2013. SignalSLAM: Simultaneous localization and mapping with mixed wifi, bluetooth, LTE and magnetic signals. In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation*. IEEE, 1–10.

[138] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. *Int. Conf. Machine Learning* (2016), 1928–1937.

[139] Fabio Morbidi, Christopher Ray, and Gian Luca Mariottini. 2011. Cooperative active target tracking for heterogeneous robots with application to gait monitoring. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3608–3613.

[140] Davide Moroni, Maria Antonietta Pascali, Marco Reggiannini, and Ovidio Salvetti. 2013. Underwater scene understanding by optical and acoustic data integration. In *Proceedings of the Meetings on Acoustics*, Vol. 17. Acoustical Society of America.

[141] Robin R. Murphy, Karen L. Dreger, Sean Newsome, Jesse Rodocker, Brian Slaughter, Richard Smith, Eric Steimle, Tetsuya Kimura, Kenichi Makabe, Kazuyuki Kon, et al. 2012. Marine heterogeneous multirobot systems at the great eastern japan tsunami recovery. *J. Field Robot.* 29, 5 (2012), 819–831.

[142] Richard M. Murray. 2007. Recent research in cooperative control of multivehicle systems. *J. Dynam. Syst. Meas. Contr.* 129, 5 (2007), 571–583.

[143] Sarat Chandra Nagavarapu, Leena Vachhani, and Arpita Sinha. 2015. Multi-robot graph exploration and map building with collision avoidance: A decentralized approach. *J. Intell. Robot. Syst.* (2015), 1–21.

[144] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. 2011. KinectFusion: Real-time dense surface mapping and tracking. In *Proceedings of the International Symposium on Mixed and Augmented Reality*. IEEE, 127–136.

[145] Martin Ngobye, Wouter T. De Groot, and Theo P. Van Der Weide. 2010. Types and priorities of multi-agent system interactions. *Interdisc. Descr. Complex Syst.* 8, 1 (2010), 49–58.

[146] Tejaswi Nimmagadda and Anima Anandkumar. 2015. Multi-object classification and unsupervised scene understanding using deep learning features and latent tree probabilistic models. *Arxiv Preprint:1505.00308* (2015).

[147] Domen Novak and Robert Riener. 2015. A survey of sensor fusion methods in wearable robotics. *Robot. Auton. Syst.* 73 (2015), 155–170.

[148] Gyeongtaek Oh, Youdan Kim, Jaemyung Ahn, and Han-Lim Choi. 2016. PSO-based optimal task allocation for cooperative timing missions. *IFAC-PapersOnLine* 49, 17 (2016), 314–319.

[149] Frans A. Oliehoek, Shimon Whiteson, and Matthijs T. J. Spaan. 2013. Approximate solutions for factored Dec-POMDPs with many agents. In *Proceedings of the International Conference on Autonomous Agents & Multi-agent Syst.* 563–570.

[150] Anibal Ollero, Simon Lacroix, Luis Merino, Jeremi Gancet, Johan Wiklund, Volker Remuß, Iker Veiga Perez, Luis G. Gutiérrez, Domingos Xavier Viegas, Miguel Angel González Benitez, et al. 2005. Multiple eyes in the skies: Architecture and perception issues in the COMETS unmanned air vehicles project. *IEEE Robot. Autom. Mag.* 12, 2 (2005), 46–57.

[151] Shayegan Omidshafiei, Jason Pazis, Christopher Amato, Jonathan P. How, and John Vian. 2017. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. In *Proceedings of the International Conference on Machine Learning*. 2681–2690.

[152] Jun Ota. 2006. Multi-agent robot systems as distributed autonomous systems. *Adv. Eng. Inf.* 20, 1 (2006), 59–70.

[153] M. Padmanabhan and G. R. Suresh. 2015. Coalition formation and task allocation of multiple autonomous robots. In *Proceedings of the IEEE 3rd International Conference on Signal Processing, Communication and Networking*. 1–5.

[154] Narendra Singh Pal and Sanjeev Sharma. 2013. Robot path planning using swarm intelligence: A survey. *Int. J. Comput. Applicat.* 83, 12 (2013).

[155] Liviu Panait and Sean Luke. 2005. Cooperative multi-agent learning: The state of the art. *Auton. Agents Multi-Agent Syst.* 11, 3 (2005), 387–434.

[156] Lynne E. Parker. 2008. Distributed intelligence: Overview of the field and its application in multi-robot systems. *J. Phys. Agents* 2, 1 (2008), 5–14.

[157] Lynne E. Parker. 2008. Multiple mobile robot systems. In *Springer Handbook of Robotics*. Springer, 921–941.

[158] Lynne E. Parker and Fang Tang. 2006. Building multirobot coalitions through automated task solution synthesis. *Proc. IEEE* 94, 7 (2006), 1289–1305.

[159] Liam Paull, Guoquan Huang, Mae Seto, and John J. Leonard. 2015. Communication-constrained multi-AUV cooperative SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 509–516.

[160] Markus Peters, Wolfgang Ketter, Maytal Saar-Tsechansky, and John Collins. 2013. A reinforcement learning approach to autonomous decision-making in smart electricity markets. *Mach. Learn.* 92, 1 (2013), 5–39.

[161] Charles Pippin, Henrik Christensen, and Lora Weiss. 2013. Performance based task assignment in multi-robot patrolling. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*. ACM, 70–76.

[162] Charles Pippin, Gary Gray, Michael Matthews, Dave Price, Ai-Ping Hu, Warren Lee, Michael Novitzky, and Paul Varnell. 2010. *The Design of an Air-ground Research Platform for Cooperative Surveillance*. Technical Report. Tech. Rep. 112010, Georgia Tech Research Institute.

[163] Sameera Ponda, Josh Redding, Han-Lim Choi, Jonathan P. How, Matt Vavrina, and John Vian. 2010. Decentralized planning for complex missions with dynamic communication constraints. In *Proceedings of the IEEE American Control Conference*. 3998–4003.

[164] David Portugal and Rui Rocha. 2011. A survey on multi-robot patrolling algorithms. In *Proceedings of the Doctoral Conference on Computing, Elect. & Ind. Syst.* Springer, 139–146.

[165] David Portugal and Rui P. Rocha. 2016. Cooperative multi-robot patrol with Bayesian learning. *Auton. Robots* 40, 5 (2016), 929–953.

[166] K. J. Prabuchandran, A. N. Hemanth Kumar, and Shalabh Bhatnagar. 2014. Multi-agent reinforcement learning for traffic signal control. In *Proceedings of the 17th International Conference on Intelligent Transportation Systems.* IEEE, 2529–2534.

[167] Marc Pujol-Gonzalez, Jesus Cerquides, Alessandro Farinelli, Pedro Meseguer, and Juan Antonio Rodriguez-Aguilar. 2015. Efficient inter-team task allocation in robocup rescue. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems.* International Foundation for Autonomous Agents & Multiagent Syst., 413–421.

[168] Martin L. Puterman. 2014. *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* John Wiley & Sons, New York, NY.

[169] Binsen Qian and Harry H. Cheng. 2014. A bio-inspired mobile agent-based coalition formation system for multiple modular-robot systems. In *Proceedings of the IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications.* 1–6.

[170] Binsen Qian and Harry H. Cheng. 2016. A mobile agent-based coalition formation system for multi-robot systems. In *Proceedings of the 12th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications.* 1–6.

[171] Richard J. Radke. 2010. A survey of distributed computer vision algorithms. In *Handbook of Ambient Intelligence and Smart Environments.* Springer, 35–55.

[172] Reihane Rahimi, Farzaneh Abdollahi, and Karo Naqshi. 2014. Time-varying formation control of a collaborative heterogeneous multi agent system. *Robot. Auton. Syst.* 62, 12 (2014), 1799–1805.

[173] Sarvapali D. Ramchurn, Maria Polukarov, Alessandro Farinelli, Cuong Truong, and Nicholas R. Jennings. 2010. Coalition formation with spatial and temporal constraints. In *Proceedings of the 9th International Conference on Autonomous Agents & Multiagent Syst.*, Vol. 3. 1181–1188.

[174] Partha Pratim Ray. 2016. Internet of robotic things: Concept, technologies, and challenges. *IEEE Access* 4 (Dec 2016), 9489–9500.

[175] Luis Riazuelo, Moritz Tenorth, Daniel Di Marco, Marta Salas, Dorian Gálvez-López, Mösenlechner, et al. 2015. Roboearth semantic mapping: A cloud enabled knowledge-based approach. *IEEE Trans. Autom. Sci. Eng.* 12, 2 (2015), 432–443.

[176] Yara Rizk and Mariette Awad. 2019. On extreme learning machines in sequential and time series prediction: A non-iterative and approximate training algorithm for recurrent neural networks. *Neurocomputing* 325 (Jan. 2019), 1–19.

[177] Yara Rizk, Mariette Awad, and Edward Tunstel. 2018. Decision making in multi agent systems: A survey. *Trans. Cogn. Dev. Syst.* 10, 3 (2018), 514–529.

[178] Cyril Robin and Simon Lacroix. 2016. Multi-robot target detection and tracking: Taxonomy and survey. *Auton. Robots* 40, 4 (2016), 729–760.

[179] Florian Rohrmüller, Dirk Wollherr, and Martin Buss. 2012. Muroco: A framework for capability-and situation-aware coalition formation in cooperative multi-robot systems. *J. Intell. Robot. Syst.* 67, 3–4 (2012), 339–370.

[180] Juan Jesús Roldán, Pablo Garcia-Aunon, Mario Garzón, Jorge de León, Jaime del Cerro, and Antonio Barrientos. 2016. Heterogeneous multi-robot system for mapping environmental variables of greenhouses. *Sensors* 16, 7 (2016), 1018.

[181] Lorenzo Rosa, Marco Cognetti, Andrea Nicastro, Pol Alvarez, and Giuseppe Oriolo. 2015. Multi-task cooperative control in a heterogeneous ground-air robot team. *IFAC J. Syst. Contr.* 48, 5 (2015), 53–58.

[182] Asim Roy. 2008. Connectionism, controllers, and a brain theory. *IEEE Trans. Syst. Man Cybern. A., Syst. Hum.* 38, 6 (2008), 1434–1441.

[183] Asim Roy. 2010. On NSF "open questions," some external properties of the brain as a learning system and an architecture for autonomous learning. In *Proceedings of the International Joint Conference on Neural Networks.* IEEE, 1–8.

[184] Stuart Russell and Peter Norvig. 2009. *Artificial Intelligence: A Modern Approach.* Pearson.

[185] Lorenzo Sabattini, Cristian Secchi, and Cesare Fantuzzi. 2016. Coordinated dynamic behaviors for multirobot systems with collision avoidance. *IEEE Trans. Cybern.* 47, 12 (2016), 4062–4073.

[186] Adam Sadilek, John Krumm, and Eric Horvitz. 2013. Crowdphysics: Planned and opportunistic crowdsourcing for physical tasks. *J. Sea Res.* 21, 10,424 (2013), 125–620.

[187] Joao Machado Santos, Micael S. Couceiro, David Portugal, and Rui P. Rocha. 2014. Fusing sonars and LRF data to perform SLAM in reduced visibility scenarios. In *Proceedings of the IEEE International Conference on Autonomous Robot Systems and Competitions.* 116–121.

[188] João Machado Santos, Micael S. Couceiro, David Portugal, and Rui P. Rocha. 2015. A sensor fusion layer to cope with reduced visibility in SLAM. *J. Intell. Robot. Syst.* 80, 3–4 (2015), 401–422.

[189] Sanem Sariel-Talay, Tucker R. Balch, and Nadia Erdogan. 2011. A generic framework for distributed multirobot cooperation. *J. Intell. Robot. Syst.* 63, 2 (2011), 323–358.

[190] Mahadev Satyanarayanan. 2017. The emergence of edge computing. *Computer* 50, 1 (2017), 30–39.
[191] Jun Sawada, Filipp Akopyan, Andrew S. Cassidy, Brian Taba, Michael V. Debole, Pallab Datta, Rodrigo Alvarez-Icaza, Arnon Amir, John V. Arthur, Alexander Andreopoulos, et al. 2016. Truenorth ecosystem for brain-inspired computing: Scalable systems, software, and applications. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis.* IEEE Press, 12.
[192] Sayan D. Sen and Julie A. Adams. 2015. An influence diagram based multi-criteria decision making framework for multirobot coalition formation. *Auton. Agents Multi-Agent Syst.* 29, 6 (2015), 1061–1090.
[193] Travis C. Service, Sayan D. Sen, and Julie A. Adams. 2014. A simultaneous descending auction for task allocation. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics.* 379–384.
[194] Lloyd S. Shapley. 1953. Stochastic games. *Proc. Natl. Acad. Sci. U.S.A.* 39, 10 (1953), 1095.
[195] Pedro M. Shiroma and Mario F. M. Campos. 2009. CoMutaR: A framework for multi-robot coordination and task allocation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems.* 4817–4824.
[196] Yoav Shoham and Kevin Leyton-Brown. 2008. *Multiagent Systems: Algorithmic, Game-theoretic, and Logical Foundations.* Cambridge University Press.
[197] David Silver and Joel Veness. 2010. Monte-Carlo planning in large POMDPs. In *Advances in Neural Information Processing Systems.* 2164–2172.
[198] Reid Simmons, Sanjiv Singh, David Hershberger, Josue Ramos, and Trey Smith. 2001. First results in the coordination of heterogeneous robots for large-scale assembly. In *Experimental Robotics VII.* Springer, 323–332.
[199] Pieter Simoens, Christof Mahieu, Femke Ongenae, Femke De Backere, Stijn De Pestel, Jelle Nelis, Filip De Turck, Shirley A. Elprama, Katriina Kilpi, Charlotte Jewell, et al. 2016. Internet of robotic things: Context-aware and personalized interventions of assistive social robots (short paper). In *Proceedings of the 2016 5th IEEE International Conference on Cloud Networking (Cloudnet'16).* IEEE, 204–207.
[200] Rubén Solera-Urena, Ana Isabel Garcia-Moral, Carmen Peláez-Moreno, Manel Martinez-Ramon, and Fernando Diaz-de Maria. 2012. Real-time robust automatic speech recognition using compact support vector machines. *IEEE Trans. Aud. Speech Lang. Process.* 20, 4 (2012), 1347–1361.
[201] Lei Song, Fan Jiang, Zhongke Shi, Rafael Molina, and Aggelos K. Katsaggelos. 2014. Toward dynamic scene understanding by hierarchical motion pattern mining. *IEEE Trans. Intell. Transp. Syst.* 15, 3 (2014), 1273–1285.
[202] Zhao Song, Xuejun Liao, and Lawrence Carin. 2015. Solving DEC-POMDPs by expectation maximization of value functions. *AAAI Spring Symposium Series* (2015), 68–76.
[203] Matthijs T. J. Spaan, Frans A. Oliehoek, and Christopher Amato. 2011. Scaling Up optimal heuristic search in Dec-POMDPs via incremental expansion. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence.*
[204] Paolo Stegagno, Marco Cognetti, L. S. Rosa, Pietro Peliti, and Giuseppe Oriolo. 2013. Relative localization and identification in a heterogeneous multi-robot system. In *Proceedings of the IEEE International Conference on Robotics and Automation.* 1857–1864.
[205] Peter Stone and Manuela Veloso. 2000. Multiagent systems: A survey from a machine learning perspective. *Auton. Robots* 8, 3 (2000), 345–383.
[206] Xueqing Sun, Tao Mao, and Laura E. Ray. 2009. Inference model for heterogeneous robot team configuration based on Reinforcement Learning. In *Proceedings of the IEEE International Conference on Technologies for Practical Robot Applications.* 55–60.
[207] Toshihisa Tabuchi, Seiichi Ozawa, and Asim Roy. 2009. An autonomous learning algorithm of resource allocating network. In *Proceedings of the International Conference on Intelligence Data Engineering and Automated Learning.* Springer, 134–141.
[208] Ayesha M. Talha and Imran N. Junejo. 2014. Dynamic scene understanding using temporal association rules. *Image Vis. Comput.* 32, 12 (2014), 1102–1116.
[209] Zheng-Hua Tan and Børge Lindberg. 2010. Low-complexity variable frame rate analysis for speech recognition and voice activity detection. *IEEE J. Sel. Top. Sign. Process.* 4, 5 (2010), 798–807.
[210] Biwei Tang, Zhanxia Zhu, Hyo-Sang Shin, and Antonios Tsourdos. 2015. Task-priority based task allocation of multiple UAVs with resource constraint. In *Proceedings of the 23th Mediterranean Conference on Control and Automation.* IEEE, 8–13.
[211] Su-Yan Tang, Yi-Fan Zhu, Qun Li, and Yong-Lin Lei. 2010. Survey of task allocation in multi agent systems. *Syst. Eng. Electron.* 32, 10 (2010), 2155–2161.
[212] Adam Taylor, Ivana Dusparic, Edgar Galván-López, Siobhán Clarke, and Vinny Cahill. 2013. Transfer learning in multi-agent systems through parallel transfer. In *Proceedings of the 30th International Conference on Machine Learning.* 1–9.
[213] Matthew E. Taylor and Peter Stone. 2009. Transfer learning for reinforcement learning domains: A survey. *J. Mach. Learn. Res.* 10 (Jul. 2009), 1633–1685.

[214] Dušan Teodorović and Mauro Dell'Orco. 2005. Bee colony optimization–a cooperative learning approach to complex transportation problems. In *Advanced OR and AI Methods in Transportation.* 51–60.

[215] Alejandro Torreño, Eva Onaindia, Antonín Komenda, and Michal Štolba. 2018. Cooperative multi-agent planning: A survey. *ACM Comput. Surv.* 50, 6 (2018), 84.

[216] J. Gregory Trafton, Laura M. Hiatt, Anthony M. Harrison, Franklin P. Tamborello II, Sangeet S. Khemlani, and Alan C. Schultz. 2013. Act-r/e: An embodied cognitive architecture for human-robot interaction. *J. Hum.-Robot Interact.* 2, 1 (2013), 30–55.

[217] Nicolas A. Tsokas and Kostas J. Kyriakopoulos. 2012. Multi-robot multiple hypothesis tracking for pedestrian tracking. *Auton. Robots* 32, 1 (2012), 63–79.

[218] Richard T. Vaughan, Gaurav S. Sukhatme, Francisco J. Mesa-Martinez, and James F. Montgomery. 2000. Fly spy: Lightweight localization and target tracking for cooperating air and ground robots. In *Distributed Autonomous Robotic Systems 4.* Springer, Berlin, 315–324.

[219] Sarvesh Vishwakarma and Anupam Agrawal. 2013. A survey on activity recognition and behavior understanding in video surveillance. *Vis. Comput.* 29, 10 (2013), 983–1009.

[220] Chao Wang, Lei Gong, Qi Yu, Xi Li, Yuan Xie, and Xuehai Zhou. 2017. DLAU: A scalable deep learning accelerator unit on FPGA. *IEEE Trans. Comput.-Aid. Des. Integr. Circ. Syst.* 36, 3 (2017), 513–517.

[221] Chaohui Wang, Nikos Komodakis, and Nikos Paragios. 2013. Markov random field modeling, inference & learning in computer vision & image understanding: A survey. *Comput. Vis. Image Understand.* 117, 11 (2013), 1610–1627.

[222] L. Wang, M. Liu, and M. Q. Meng. 2017. A hierarchical auction-based mechanism for real-time resource allocation in cloud robotic systems. *IEEE Trans. Cybern.* 47, 2 (2017), 473–484.

[223] Qining Wang, Ming Wu, Yan Huang, and Long Wang. 2008. Formation control of heterogeneous multi-robot systems. *IFAC Proc. Vol.* 41, 2 (2008), 6596–6601.

[224] Wanyuan Wang, Jiuchuan Jiang, Bo An, Yichuan Jiang, and Bing Chen. 2017. Toward efficient team formation for crowdsourcing in noncooperative social networks. *IEEE Trans. Cybern.* 47, 12 (2017), 4208–4222.

[225] Sarah E. Webster, Jeffrey M. Walls, Louis L. Whitcomb, and Ryan M. Eustice. 2013. Decentralized extended information filter for single-beacon cooperative acoustic navigation: Theory and experiments. *IEEE Trans. Robot.* 29, 4 (2013), 957–974.

[226] Gerhard Weiss (Ed.). 2013. *Multiagent Systems.* MIT Press.

[227] Shimon Whiteson. 2012. Evolutionary computation for reinforcement learning. In *Reinforcement Learning.* Springer, Berlin, 325–355.

[228] Joey Wilson and Neal Patwari. 2011. See-through walls: Motion tracking using variance-based radio tomography networks. *IEEE Trans. Mobile Comput.* 10, 5 (2011), 612–621.

[229] Jinseok Woo, Kazuyoshi Wada, and Naoyuki Kubota. 2012. Robot partner system for elderly people care by using sensor network. In *Proceedings of the 4th IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics.* 1329–1334.

[230] Bo Xu, Zhaofeng Yang, Yu Ge, and Zhiping Peng. 2015. Coalition formation in multi-agent systems based on improved particle swarm optimization algorithm. *Int. J. Hybrid Inf. Technol.* 8, 3 (2015).

[231] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. 2013. A survey and analysis of multi-robot coordination. *Int. J. Adv. Robot. Syst.* 10 (2013).

[232] Zhang Yu, Liu Shuhua, Fu Shuai, and Wu Di. 2009. A quantum-inspired ant colony optimization for robot coalition formation. In *Proceedings of the Chinese Control and Decision Conference.* IEEE, 626–631.

[233] Yu Zhang and Lynne E. Parker. 2013. IQ-ASyMTRe: Forming executable coalitions for tightly coupled multirobot tasks. *IEEE Trans. Robot.* 29, 2 (2013), 400–416.

[234] Z. Zhang, D. Zhao, J. Gao, D. Wang, and Y. Dai. 2017. FMRQ-A multiagent reinforcement learning algorithm for fully cooperative tasks. *IEEE Trans. Cybern.* 47, 6 (2017), 1367–1379.

[235] Yuanshi Zheng and Long Wang. 2012. Distributed consensus of heterogeneous multi-agent systems with fixed and switching topologies. *Int. J. Contr.* 85, 12 (2012), 1967–1976.

[236] Yuanshi Zheng and Long Wang. 2014. Containment control of heterogeneous multi-agent systems. *Int. J. Contr.* 87, 1 (2014), 1–8.

[237] Y. Zheng, Y. Zhu, and L. Wang. 2011. Consensus of heterogeneous multi-agent systems. *IET Contr. Theory Applicat.* 5, 16 (2011), 1881–1888.

[238] Liu Zhong, Gao Xiao-Guang, and Fu Xiao-Wei. 2015. Coalition formation for multiple heterogeneous UAVs in unknown environment. In *Proceedings of the IEEE 5th International Conference on Instrumentation and Measurement, Computer, Communication and Control.* 1222–1227.

[239] Kai Zhou, Karthik Mahesh Varadarajan, Michael Zillich, and Markus Vincze. 2011. Web mining driven semantic scene understanding and object localization. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics.* 2824–2829.

[240] L. Zhou, P. Yang, C. Chen, and Y. Gao. 2017. Multiagent reinforcement learning with sparse interactions by negotiation and knowledge transfer. *IEEE Trans. Cybern.* 47, 5 (2017), 1238–1250.

[241] Robert Zlot and Anthony Stentz. 2003. Multirobot control using task abstraction in a market framework. In *Proceedings of the Collaborative Technology Alliances Conference.*

[242] Robert Zlot and Anthony Stentz. 2005. Complex task allocation for multiple robots. In *Proceedings of the IEEE International Conference on Robotics and Automation.* 1515–1522.

[243] Robert Zlot and Anthony Stentz. 2006. Market-based multirobot coordination for complex tasks. *Int. J. Robot. Research* 25, 1 (2006), 73–101.

Received January 2018; revised January 2019; accepted January 2019