IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ADVANCED COMMUNICATIONS AND NETWORKING TECHNIQUES FOR WIRELESS CONNECTED INTELLIGENT ROBOT SWARMS

Received January 31, 2020, accepted March 9, 2020, date of publication March 17, 2020, date of current version March 26, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.2981408*

# A Survey on Multi-Robot Coordination in Electromagnetic Adversarial Environment: Challenges and Techniques

**YUNLONG WU[1,2], XIAOGUANG REN[1,2], HAO ZHOU[1,2], YANZHEN WANG[1,2], AND XIAODONG YI[1]**

[1]Artificial Intelligence Research Center (AIRC), National Innovation Institute of Defense Technology (NIIDT), Beijing 100071, China
[2]Tianjin Artificial Intelligence Innovation Center (TAIIC), Tianjin 300457, China

Corresponding authors: Xiaoguang Ren (rxg_nudt@126.com) and Yanzhen Wang (yanzhenwang@hotmail.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB1001900, and in part by the National Natural Science Foundation of China under Grant 61906212, Grant 91648204, Grant 61601486, and Grant 61802426.

**ABSTRACT** Wireless communications and networking are playing an important role in coordination and cooperation of multi-robot systems (MRS). However, it is challenging to keep a reliable and stable wireless connection in practical applications. Especially, robots acting in electromagnetic adversarial (EA) environments may encounter more serious situations including scarce spectrum, active interference, adversarial competition, etc. In this survey, we firstly analyze the challenges faced by MRS in EA environments, and provide a categorization according to the "sense-decide-act" robot control procedure. Secondly, enabling techniques for each challenge are introduced. Finally, typical robotics software architectures are introduced, as frameworks for efficient arrangement of the above mentioned enabling techniques.

**INDEX TERMS** Electromagnetic adversarial environment, multi-robot coordination, communication connectivity, survey.

## I. INTRODUCTION

Bill Gates published an article in *Scientific American* in 2007, named *a robot in every home*. He envisions a future where robots will become a nearly ubiquitous part of our daily lives [1]. More than 10 years have past, we indeed see that millions of robots have stepped into our houses and help us to do many things. Nowadays, various kinds of robots are also playing important roles in our social lives, e.g., surveillance [2], data collection [3], field exploration [4], etc. Compared with single-robot systems, a system consisting of multiple robots can largely improve the task performance by coordination, which is more suitable for challenging tasks and complex environment. In recent years, considerable research contributions have been made to the evolution of multi-robot system (MRS), and this trend will continue in the coming decades [5].

In most multi-robot application scenarios, coordination among robots largely relies on the success of information exchange [6], such as situational awareness information, control commands, state data, etc. As mobile robots are less likely to be connected via wires, thus maintaining a reliable and stable wireless communication connectivity becomes vital for multi-robot coordination. However, in realistic communication environment, the wireless channel often experiences path loss, shadowing and multipath fading [7], which largely decreases the channel reliability and makes the communication link unstable. Especially, in complex electromagnetic environment, there will be more influencing factors affecting the wireless channel. A typical example is the unknown jammers [8], [9], which actively disrupt the communication connectivity of other entities that rely on wireless communications.

In the era of robotics, the jammers will become more autonomous and intelligent. With the development of software defined radio (SDR) [10] and artificial intelligence technologies, the robots equipped with the programmable radio frequency (RF) devices can quickly change waveforms and create constant new signals based on the perception of the electromagnetic environment. In order to distinguish from the traditional jammers, we call the jammers that incorporate robotic technologies the *jamming robots*. The higher the

The associate editor coordinating the review of this manuscript and approving it for publication was Luciano Bononi.

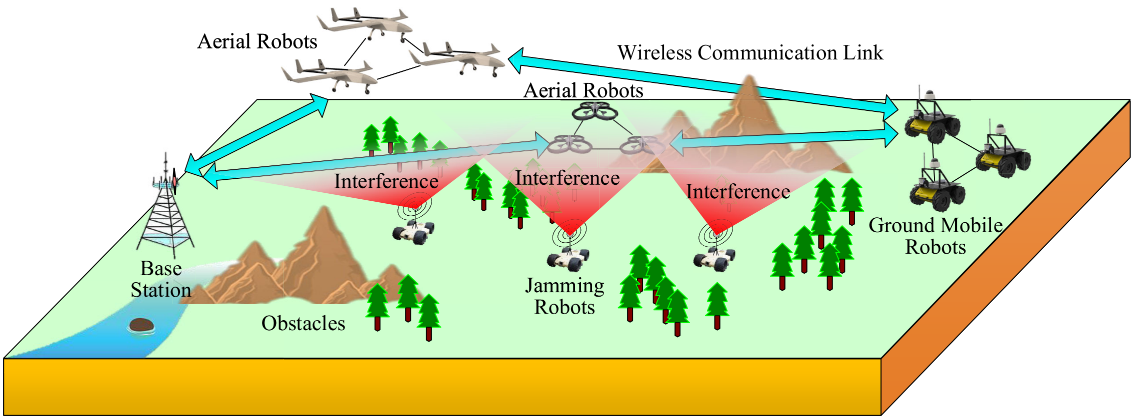



**FIGURE 1.** A representative EA environment where task robots (e.g., ground mobile robots and aerial robots) aim at finishing specific tasks, e.g., exploration, surveillance, target tracking, etc. Meanwhile, a team of intelligent jamming robots seamlessly change their positions to disrupt the communication connectivity among task robots.

intelligence of the jamming robots, the more difficult it is to be located and identified. As a result, while the robots are performing tasks, they also need to compete with the jamming robots for available communication resources. As a result, in this environment, the task robots may face more adversarial hindrances from the jamming robots. This survey may define such kind of environment as an *electromagnetic adversarial (EA) environment* and explore the effective ways to solve the multi-robot coordination problem in EA environment, as shown in Fig. 1.

Cynefin framework classifies the environment into five categories: simple, complicated, complex, chaotic and disorder [11]. The EA environment can be seen as a kind of complex environment. The framework proposes to solve the problem in complex environment with the loop of ‘‘sense-analyse-respond’’, which emphasizes the continuous interaction with the environment to obtain the solutions. The Cynefin framework is similar to the classic robotic ‘‘sense-plan-act’’ operating paradigm [12]. In order to highlight the intelligent characteristics of robots, ‘‘analyse’’ and ‘‘plan’’ can be collectively marked as ‘‘decide’’, and ‘‘act’’ can incorporate the meaning of ‘‘response’’. Therefore, the above models can be uniformly represented as the ‘‘sense-decide-act’’ behavior chain, and the multi-robot coordination in EA environment can also be carried out according to this model.

Following the ‘‘sense-decide-act’’ behavior chain, we may firstly summarize the challenges of multi-robot coordination in EA environment in Section II. Then, for each challenge, we will try to put forward the relevant techniques and analyze their applicable conditions in Section III. Finally, Section IV concludes this survey. We hope the research results of this survey may provide a theoretical reference for the research of multi-robot coordination in EA environment.

## II. MAIN CHALLENGES

In this section, we will summarize the main challenges of multi-robot coordination in EA environment according to the ‘‘sense-decide-act’’ behavior chain. The first three challenges introduce the corresponding challenges from sense, decide and act, respectively. The final challenge pushes out the simulation environment requirements for training multi-robot coordination in EA environment.

### A. COMMUNICATION CONNECTIVITY MAINTENANCE

In multi-robot applications, the performance of task execution largely relies on the information sharing (e.g. sensing data) between robots. Therefore, keeping a wireless communication connectivity in the sense stage is one of the fundamental requirements for multi-robot coordination. However, in EA environment, the wireless channel is highly dynamic and the active interference from the intelligent jamming robots will further decrease the average received signal-to-noise ratio (SNR) on the receiver robot, which makes the communication connectivity even more serious. Therefore, keeping a reliable and stable wireless communication link among robots in EA environment is particularly important and challenging, and designing appropriate solutions is urgently needed.

### B. COOPERATIVE-COMPETITIVE DECISION-MAKING

Multi-robot coordination in EA environment is essentially a mixed cooperative-competitive problem. The relationship between task robots is cooperative, and they have consistent mission objectives. However, there is a competitive relationship between the task robots and the jamming robots. They are both competing for the spectrum resources, which makes the objective of task robots and the jamming robots coupled. For example, when the jamming robots occupy a certain frequency band, it will result in the scarcity of the spectrum resources of the task robots in this band. Therefore, in the decide stage, the relationship and objectives of the task robots and the jamming robots need to be considered at the same time. Based on the above considerations, constructing a ‘‘cooperative-competitive’’ unified solving framework is necessary to achieve the multi-robot coordination in EA environment.

### C. AUTONOMOUS ACTION CONTROL

In the act stage, the control command of a robot is usually calculated according to the outputs of the decide stage. However, the higher-level decision cannot guarantee stability and robustness for the lower-level control commands. Especially, in EA environment, the surroundings around robots are highly dynamic and accident-prone. For solving this problem, it is necessary to allow each robot to play their own ''personality'' and fully exploit its autonomy. In order to achieve this goal, a hierarchical control engine is needed. When the communication link continues to be poor or the objective function can not be further optimized, the engine will take over the right of control, and select actions according to predefined rules. We may see that autonomy and coordination are never opposites, but a unified whole. Therefore, how to adopt autonomy to compensate for coordination is one of the main challenges to improve the swarm intelligence in the EA environment.

### D. ENVIRONMENT ADAPTATION

Training the ''sense-decide-act'' behavior chain needs to continuously interact with the environment. Considering the intelligence of the jamming robots is continuously evolving, the task robots will face an unknown and constant new environment. Training algorithms in real electromagnetic environment may encounter problems such as high deploying cost, inflexible scene update, and low training efficiency. In order to quickly obtain a adaptive algorithm for unknown environment, it is necessary to build a multi-domain simulation environment which should have the ability to rapidly generate vivid electromagnetic-geographical environment.

## III. ENABLING TECHNIQUES

In order to meet the challenges proposed in Section II, this section will review the existing enabling techniques which may give useful suggestions for multi-robot coordination in EA environment. Fig. 2 demonstrates the relationship between the challenges above and the enabling techniques

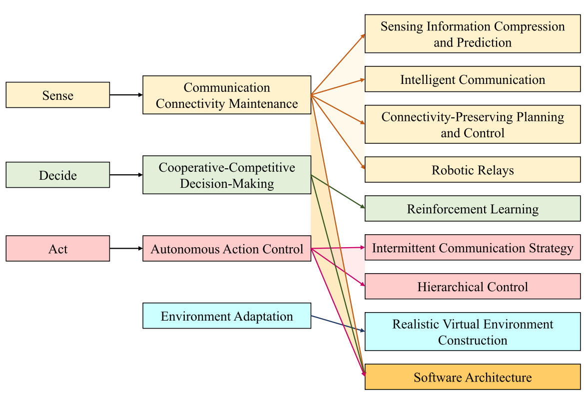



**FIGURE 2. The relationship between the challenges and the enabling techniques.**

reviewed in this section. For maintaining the communication connectivity, we have summarized four kinds of associated techniques: sensing information compression and prediction, intelligent communication, connectivity-preserving planning and control, and robotic relays. In the challenge of cooperative-competitive decision-making, we recommend using reinforcement learning, especially the multi-agent deep reinforcement learning to solve the problem. For the autonomous action control, we may relax the requirements for maintaining all-time communication connectivity and give the robot greater autonomy, which involves two kinds of techniques: intermittent communication strategy and hierarchical control. The technique of realistic virtual environment construction supports the evolving of the whole ''sense-decide-act'' behavior chain in a large number of different scenarios, which may help to strengthen the environment adaptation of the robots. Finally, in order to enable the above enabling techniques to be adapted to specific robots, the task should be flexibly adjusted by a well-designed software architecture according to the environment.

### A. SENSING INFORMATION COMPRESSION AND PREDICTION

As discussed in Subsection II-A, multi-robot coordination relies on information sharing, e.g. cooperative localization and mapping [13]. However, in EA environment, the communication bandwidth is limited which makes high-dimensional data sharing between task robots challenging. In order to solve this problem, we may consider the techniques from both the source and sink sides.

In the source side, for saving bandwidth, we may reduce the dimensions of the shared data by extracting features whose dimensions are much lower than the raw data. The main dimension reduction methods can be divided into two categories: manual and non-manual methods. In the manual methods, each feature of the data is carefully designed by hand. The kind of methods have a long operating cycle, low flexibility, and highly depends on domain experts' knowledge [14], [15]. A typical example is incremental feature dependency discovery (iFDD), which has been successfully used to denote the state features of an air combat scenario [16]. In the scenario, the pilot's preference is fully considered when designing the base features, e.g., aspect angle, antenna train angle, aspect angle rate, etc. For the non-manual method, it may refer to the approaches that generate features without explicitly specifying their attributes, e.g., principal component analysis (PCA) [17], multiple dimensional scaling (MDS) [18], isometric mapping (Isomap) [18], locally linear embedding (LLE) [19], etc. When combined with artificial neural network and deep learning, the feature extraction will own the ability to learn nonlinear features [20]. autoencoder (AE), as a typical deep learning method, plays an important role in unsupervised learning and nonlinear feature extraction, which has been widely used in the field of image and speech [21], [22]. Fig. 3 demonstrates an example of variational autoencoder (VAE) used in image feature extraction,

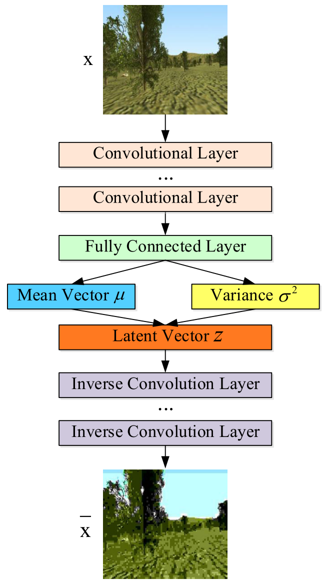



**FIGURE 3.** An example of variational autoencoder (VAE) used in image encoding and decoding.

where $z$ denotes the encoded vector (latent variables) whose dimension is smaller than that of the raw image $x$.

As soon as the shared data is compressed, it will be transmitted through the wireless communication link, and sent to the sink robot. In EA environment, the transmitted data is easy to be interrupted. Therefore, on the sink side, it is also necessary to consider how to recover the shared data when it is incomplete or lost. For the high-dimension data, an alternative way is to predict the lost data by correlating historical memory. Typical methods are recurrent neural network (RNN) [23], long short-term memory (LSTM) [24], etc. On the other hand, in addition to high-dimensional data, there are some low-dimensional data need to be shared for multi-robot collaborative control, e.g. positions. For example, collision avoidance requires the current robot to know the positions of the neighboring robots. When the data is not received, the robot needs to predict the other robots' behaviors. The usual methods for predicting the other robots' behaviors are based on prior models. For example, a simple model is assuming others robots move at a constant speed [25]. However, this assumption may not be accurate in many scenarios. In order to improve the accuracy of prediction, effective methods may embed the dynamics models into the behavior predictions of the robots, a typical method is model predictive control (MPC) and its extensions [26].

### B. INTELLIGENT COMMUNICATION

In above subsection, we have summarized the techniques of compression and prediction, trying to make the shared data can be successfully received by the sink side. These techniques are analyzed from the application level of the network protocol. However, the transmission efficiency is largely determined by the physical layer. Therefore, we may use the anti-jamming communication techniques to make the data transmission more reliable [27]. Traditional methods include frequency-hopping spread spectrum (FHSS), direct-sequence spread spectrum (DSSS), hybrid spread spectrum [28], smart antenna [29], diversity scheme [30], etc.

**FIGURE 4.** A TurtleBot robot equipped with USRP N210 and GNU Radio.

When entering the era of artificial intelligence, in order to overcome the challenge of the intelligent jamming robots, the task robots need to have the ability to exploit the spectrum intelligently. Cognitive radio (CR) is the key enabling technique that supports dynamic spectrum access networks which are seen as the next generation communication networks [31]. Moreover, many researchers regard CR as the key technology for 5G deployment [32], [33]. In CR, the users (e.g. the task robots) should be equipped with two main capabilities: cognitive capability and reconfigurability [34]. Cognitive capability denotes the ability to sense the electromagnetic environment, including spectrum sensing, autonomous learning, modeling, and reasoning. Reconfigurability refers to the ability to dynamically change the waveform configurations and parameters according to the sensed data for achieving a better communication performance [31]. In order to achieve reconfigurability, there are some software architectures have given the paradigms, e.g., GNU Radio [35], Software Communication Architecture (SCA) [36], Iris [37], etc. For each software architecture, it just provides the signal processing blocks. In realistic communication applications, the software architecture also needs to use external radio frequency (RF) hardware to create software-defined radios (SDR). The most popular hardware platform is the Universal Software Radio Peripheral (USRP) which is designed by Ettus Research [38]. Currently, designing and prototyping radio communication systems with GNU Radio and USRP has been the paradigm in CR and SDR [39]. Fig. 4 gives an example that a radio-mapping robot mounting a USRP device with GNU Radio on a TurtleBot robot [40].

Moreover, in addition to using electromagnetic signals for communication, other communication media can also be used. For example, line-of-sight (LOS) based communication (by infrared or visible light) is a typical representative which is difficult to be interfered by the jammers [41]. This kind of new communication paradigm is very suitable for the communication between military units in the battlefield, e.g. establishing communication links [41], collaborative patrol [42], [43], etc.

### C. CONNECTIVITY-PRESERVING PLANNING AND CONTROL

Considering the above techniques are all implemented on the onboard computer carried on the robot, they can be seen as a kind of passive methods which may not change the channel conditions. In order to actively improve the channel condition, besides the onboard computer, a good choice is adopting the mobility of the robots, which has been analyzed in our previous work [44]. By utilizing the knowledge of connectivity quality, we may plan and control the motion of the robots to improve specified task-oriented performance, while satisfying certain communication constraints. This kind of technique is often called *communication-aware motion planning* [44], [45], *connectivity preserving* [46], [47], or *connectivity maintaining* [48], [49]. In this subsection, we may use *connectivity-preserving planning and control* (CPPC) to represent the above concepts.

In fact, connectivity is not a new topic in recent years. As early as the beginning of this century, with the rapid development of wireless sensor network (WSN), how to maintain the sensing coverage and communication connectivity between sensors has become the focus of researchers [50], [51]. In WSN, the communication range of each sensor is modeled as a sphere, where the sensor can only communicate with the sensors in the sphere. The aim of deploying sensors in WSN is achieving the desired coverage with the least number of sensors [52]. Later, in order to meet the requirements of different tasks and cope with rapid topology changes [53], mobile wireless sensor network (MWSN) has been proposed, and the researches on connectivity MWSN have attracted a lot of research attentions [54]–[56]. The mobile sensors can be seen as the simplest robots, which are only used to implement the sensing function.

Compared with the mobile sensors, the robots have more powerful capabilities and longer battery life, which makes CPPC attract more and more attentions. Based on different ways to model the connectivity, the current works about CPPC can be divided into two categories: one is the graph theory based (GT-based) method and the other is realistic channel based (RC-based) method.

In GT-based method, the robots are abstracted into nodes in the graph, while the edges represent the communication links between nodes [57]. The communication range of a robot is modeled as a spherical region with a radius $r$. When the distance between two robots is smaller than $r$, we think the robots can communicate with each other with little bit error. On the contrary, if the distance exceeds $r$, the connectivity is regarded broken [58]. Within this framework, numerous research contributions about CPPC for multi-robot systems have been proposed [48], [49], [59]–[61]. For a comprehensive overview and tutorial of adopting graph-theoretic definition of connectivity, the readers are referred to [57], [62], in which the authors also provided various approaches about CPPC ranging from convex optimization to potential fields based control methods.

However, the spherical model can not reflect the real communication environments, which motivates the research on RC-based method fully considering the effects of path loss, shadowing and multipath fading. The RC-based method often assesses the communication connectivity with a probabilistic channel model based on realistic channel measurements. For a more detailed discussion of the realistic channel modeling, you may refer to [7]. The RC-based method is to continuously evaluate the wireless channel of the multi-robot system during the execution of the task, and schedule the mobility of the robot to improve the communication connectivity. In [45], the authors proposed to exploit the mobility of the robots for improving the performance of wireless channel assessment and target tracking, as well as minimizing the probability of target detection error for surveillance, while guaranteeing connectivity constraints in [63]. In [2], the authors consider a multi-robot surveillance scenario and try to exploit the mobility of the robots to improve the channel capacity. However, considering the the complex interactions between the environments and the electro-magnetic waves in EA environment, it is hard to predict the exact channel quality of a given location. Therefore, in our previous work of [40], we assume the wireless fading channel is quasi-static and directly exploiting the realistic channel measurements as the basis to solve the CPPC problem.

Based on the above discussion, CPPC focuses on exploiting the mobility of the robots to ensure the communication connectivity between robots. In CPPC, the communication schemes are fixed or having a limited adaptive capability so that the communication quality has to be guaranteed by the motion planning or controlling. Against this problem, an inevitable way is to combine the intelligent communication techniques in Subsection III-B with the methods in this subsection, then form a joint communication-motion planning and controlling framework.

### D. ROBOTIC RELAYS

Subsection III-C has reviewed the works about keeping a permanent communication connectivity in multi-robot systems. In these works, each robot is of the same type and is capable of performing tasks. However, when communication quality gets worse, in order to enhance the channel quality, some robots may act as relays specializing in establishing communication links for other robots [64]. In EA environment, we may deployed several task robots acting as relays to ensure the communication connectivity between the source and sink, as shown in Fig. 5.

In recent years, many research works focus on combining relay technologies and robotics to strengthen the communication performance in multi-robot systems [64], [82]. According to the main topic of this survey, we extract seven key words to summarize the existing works, including *scenario, channel model, metric, mobility of RX/TX, components,* and *method*. The term of scenario refers to the specific tasks, e.g. link building [67], task allocation [71], exploration [72], etc. Secondly, the channel model depicts how many influencing

**TABLE 1.** The related works of robotic relays are classified according to the key words: scenario, channel model, metric, mobility of RX/TX, components, and method.

| Related Work | Scenario | Channel Model | Metric | Mobility of RX | Mobility of TX | Component | Method |
|---|---|---|---|---|---|---|---|
| Y. Wu, et al. [65] | Occasional task | Path loss, Shadowing, Multipath fading | PER | Fixed | Movable | Homogenous | Optimization |
| Y. Marchukov, et al. [66] | Information gathering | Path loss | Distance | Fixed | Fixed | Homogenous | Search, Fast marching |
| Y. Yan, et al. [67] | Link building | Path loss, Shadowing, Multipath fading | BER | Fixed | Fixed | Homogenous | Optimization |
| Y. Zeng, et al. [68] | Link building | Path loss | SNR | Fixed | Fixed | Single Relay | Optimization |
| J. Fink, et al. [69], [70] | Link building | Path loss, Shadowing, Multipath fading | PER | Fixed | Movable | Homogenous | Gradient-based control, Probabilistic search |
| S. Ponda, et al. [71] | Task allocation | Path loss model | Distance | Fixed | Movable | Heterogeneous | Auction protocol |
| N. Goddemeier, et al. [72] | Exploration | Path loss | RSSI | Fixed | Movable | Homogenous | Potential field |
| F. El-Moukaddem, et al. [73] | Wireless sensor networks | Path loss | PRR | Fixed | Fixed | Homogenous | Search |
| B. Min, et al. [74] | Link building | Path loss | Distance | Fixed | Fixed | Homogenous | Genetic algorithm, Genetic swarm algorithm |
| D. Choi, et al. [75] | Link building | Path loss, Multipath fading | Capacity | Fixed | Fixed | Single | Optimization |
| O. Faqir, et al. [76] | Data transmission | Path loss, Multipath fading | Capacity | Fixed | Fixed | Single | Model predictive control |
| S. Zeng, et al. [77] | Link building | Path loss, Multipath fading | SNR | Fixed | Movable | Single | Optimization |
| Y. Pei et al. [78] | Exploration | Path loss | Distance | Fixed | Movable | Heterogeneous | Search |
| P. Ladosz, et al. [79], [80] | Link building | Path loss, Shadowing | Global message connec-tivity, SNR | Movable | Movable | Homogenous | Optimization, Model predictive control |
| K. Kim, et al. [81] | Link building | Measurement | Packet-delivery ratio (PDR) | Fixed | Movable | Homogenous | Spatial probing |

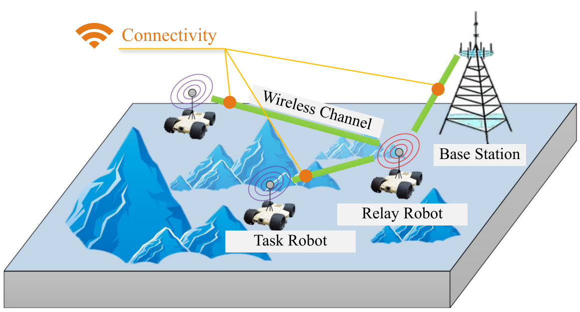



**FIGURE 5.** A scenario where a relay robot helps the task robots to keep a communication connectivity with the base station.

factors considered in wireless channel modeling. According to the theory of wireless communication, the channel is often modeled as a system affected by three factors: path loss, shadowing and multipath fading [7]. As the main influencing factor, path loss is considered in most of the existing works. In most cases, the effect of path loss is modeled as a decreasing function proportional to distance between the source and the sink. While in some works, the communication range of a task robot is reduced to a disk model, and the area beyond the disk is regarded disconnected. Shadowing is the effect that the signal is blocked by the obstacles between the transmitter and the receiver. It is often modeled by a non-smooth step function of the number of obstacles between two task robots [70]. For the multipath fading, it is due to the reflections and refractions, which is often modeled as a zero-mean random variable [7]. The third term, metric, denotes the quantitative standard to evaluate the communication connectivity. The candidate metrics can be the channel capacity [75], bit error rate (BER) [67], packet error rate (PER) [65], packet reception ratio (PRR) [73], etc. Some more direct metrics are also adopted in existing works, such as signal-to-noise ratio (SNR) [68], received signal strength indication (RSSI) [72], etc. Moreover, according to whether the data source/sink can be movable, we may use the metric of mobility of RX/TX to classify the existing works. The next term, component, is used to denote whether the robot team is homogenous or heterogeneous. Most of existing works focus on the research of homogenous robot teams. However, compared with the homogenous team, heterogeneous robots may be more suit to handle certain tasks which will involve different roles and responsibilities [71], [78]. For example, in a disaster rescue task, the aerial robots may be equipped with cameras to perform search mission, and the ground robots can be deployed to perform rescue operations [71]. The last term, method, is used to select the specific solving methods, e.g., probabilistic search [69], [70], potential field [72], etc.

### E. REINFORCEMENT LEARNING

The multi-robot coordination in EA environment is a form of complex problem, which is hard to model every conditions of circumstances. Moreover, the actions of robots will make the environment constantly changing, which results in highly dynamic problem space. Therefore, we may not adopt the supervised learning and unsupervised learning which highly rely on data. Another popular learning type, reinforcement learning (RL) [83], not requiring pre-existing knowledge or data, is an effective technique for solving large-scale complex problems. Furthermore, in EA environment, the intelligence level of jamming robots is continuously improving along with the task robots, which makes the task robots always face a new circumstance. RL trains models by receiving rewards or punishments on the actions taken by robots, so it is able to learn policies to respond to unforeseen environments. In recent years, the researches on RL related algorithms have been very active. Fig. 6 lists some of the main algorithms for current reinforcement learning. According to whether the transition model of each robot is known, RL methods can be divied into model-free and model-based methods. In model-free methods, robots need to keep interacting with environment by trial and error to learn about the consequences of actions. Moreover, model-free methods can be further divided into value-based and policy-based methods, where value-based methods emphasize using a value function to evaluate the action taking in a given state, while policy-based methods may focus on directly searching for the optimal policy [84]. Model-based methods also play an important role in RL, especially in the scenario where the cost of trial-and-error experiments is unaffordable. However, classical RL, including both model-free and model-based methods, may lack scalability and just fit for fairly low-dimensional problems. Nowadays, with the continuous improvement of computer performance and data processing capability, RL combined with deep learning is able to be used to solve large-scale problems.

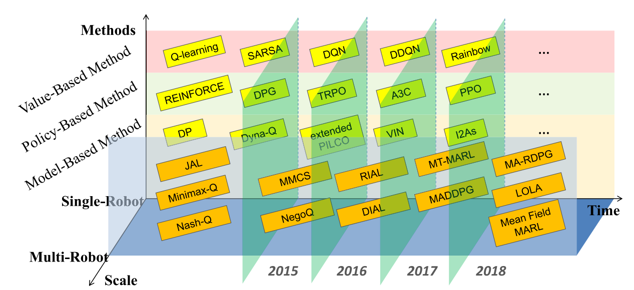



**FIGURE 6.** Typical algorithms for reinforcement learning and multi-agent reinforcement learning.

The cooperative-competitive decision-making in EA environment may involve multi-agent reinforcement learning (MARL). At present, many researchers have done a lot of work in multi-robot decision-making based on deep RL theory, and applied to a number of scenarios, e.g., multi-scenario ranking [85], multi-target capturing [86], battle games [87], switch riddle [88], etc. In MARL, the core issue is how to solve the "curse of dimensionality" problem caused by the growth of the number of robots [89]. A simple way is to adopt the centralized method of classical RL. Although the algorithm framework is simple, the state space and action space of the problem will increase exponentially with the number of robots [90]. On the other hand, adopting decentralized method will affect the convergence of RL algorithms. Therefore, in order to solve this problem, the alternative solution is based on the framework of centralized training with decentralized execution [85], [89]. That is, although the decentralized mode is adopted in the training stage, the global signal is introduced to guide the training process. In the execution stage, the decentralized strategy is completely adopted. Fig. 7 demonstrates a typical decentralized MARL learning framework based on actor-critic setup [89]. In this learning framework, each robot obtains the sensed data from the sense stage and selects an action based on the action policy which will be evaluated by a global evaluation function.

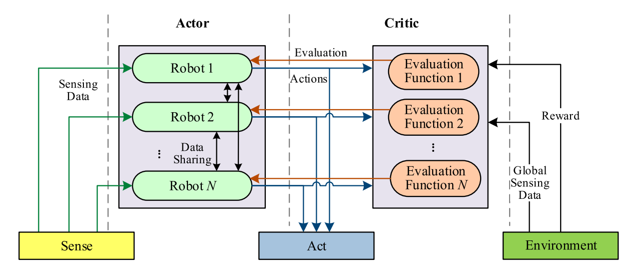



**FIGURE 7.** A typical decentralized framework for multi-agent reinforcement learning.

### F. INTERMITTENT COMMUNICATION STRATEGY

From Subsection III-A to Subsection III-D, we have summarized the techniques trying to make the multi-robot system connected. However, considering the uncertainty of the realistic wireless channel, keeping an all-time communication connectivity is not possible in practice [91]. Moreover, in EA environment, the active interference from the jamming robots may make the communication links more fragile. In this condition, insisting on maintaining connectivity may restrict the motion of the robots, which will affect the performance of task execution. Therefore, for the multi-robot system, an alternative way is making full use of the autonomy of the robots and allowing the temporary connectivity disconnection. However, the robots need to be able to communicate at least once within a limited time, which is known in the existing literatures as *intermittent communication* [91], [92].

In order to achieve the intermittent communication, a common strategy is to adopt the rendezvous-based communication method [93]. In this method, the task robots will communicate in a rendezvous fashion. The rendezvous can occur either in the spatial dimension or in the time dimension, or both.

In the spatial dimension, the robot will communicate at the specified locations [91]–[93]. In [91] and [92], the robots are used for performing data gathering tasks and periodically communicate at common locations. In these works, the possible communication locations are predefined and the robots can only communicate at the same location. In [93], the authors considered a exploration scenario where a team of robots survey an unknown environment independently and exchange map information in a scheduled rendezvous location. In this scenario, the rendezvous locations are not predefined but are determined based on the information of the previous rendezvous. In EA environment, if the communication link is seriously interfered, this kind of data sharing method can be adopted, which can be regarded as a *hand-off* communication method. In addition to the spatial dimension, the robots can communicate at specified time intervals, which is known as *periodic connectivity*. For example, in [94], a multi-robot search problem is considered where the robots search for a target and regain connectivity in a prearranged time interval and location. This method does not require communication at the same location, but needs the line-of-sight communication condition, which is suitable for the scenario where the communication environment is relatively simple and the obstacles are not dense.

The above methods can be regarded as a kind of active communication, i.e., each robot follows predefined rules to reach a rendezvous. However, if there are no predefined rules, or if the communication is only temporarily decided to be established, then the robot is required to search for the robot to be communicated. Multi-agent target searching is a hot research topic and receives a lot of attentions [95]. For the multi-agent searching problem, its computational complexity is exponential in the number of searchers, but can be optimized with coordination [96], [97]. In [97], the authors propose a reconnection method that combines with target tracking. Each robot may maintain beliefs of other robots' positions which can be used to plan the optimal path to the targets. By using the beliefs, the other robots can quickly find the disconnected target robot with a low computational complexity.

### G. HIERARCHICAL CONTROL

According to the "sense-decide-act" behavior chain, the action policy of each task robot comes from the output of the decide stage. However, in adversarial environment, the intelligence of both task robots and jamming robots continue increasing. When the intelligence level of the task robots is lower than the jamming robots, the task robots may not obtain good action policies, or even worse. In such cases, the task robots should make full use of their autonomy, and adjust the action policy according to how "bad" the condition is. Therefore, in the act stage, when the task robots are not clever than the jamming robots, they may select suboptimal action policies to get rid of the jamming robots. In order to support this function, it needs to build a hierarchical control engine which can select the appropriate action according to environmental metrics, such as the quality of communication links. Currently, the implementation for robotic hierarchical control mainly includes finite state machine, decision tree [98], subsumption architecture [99], behavior tree [100].

The finite state machine is a basic method for implementing action switching. However, when adding a new action state, the finite state machine may require a lot of changes and have poor scalability [100]. The decision tree realizes the action selection through the nesting of if-then clauses, which has the advantages of modularization and hierarchy. However, since the information flow is one-way, no feedback information flows out from the node, which makes the fault handling very difficult [98], [100]. The subsumption architecture is based on hierarchical control theory, which decomposes complex tasks into specific actions. High-priority actions can accommodate (or suppress) low-priority actions, but the method also faces the difficulties in adding or removing actions [99]. The behavior tree (BT) model has a universal representation ability and can accommodate the above three models, which are currently widely used in robot motion control [100].

Following the BT model, we may firstly classify the action policies into several action levels. Each level is assigned a priority value and a capacity value. The higher the capability value, the lower the corresponding priority value. Assume there are $L$ levels $H = \{H_1, H_2, \cdots, H_L\}$, each level $H_i$ may contain one or more actions $H_i = \left\{h_{i,1}, h_{i,2}, \cdots, h_{i,B}\right\}$. If the priority of the priority is $i$, the corresponding capacity is $L - i$. Each level may correspond to a triggering condition $C_i$ which is used to select action policies. A BT model contains five kinds of nodes: selector, sequence, parallel, action, and condition. The leaf nodes of the BT can be action or condition nodes. Condition nodes are used to judge if some triggering condition is satisfied, e.g., battery level, etc. An action node usually corresponds to a specific action, e.g., trajectory tracing. In general, selector, sequence and parallel nodes do not act as the leaf nodes. The selector node is used to judge if the condition node is satisfied. Selector node may firstly try its leftmost children, i.e. the condition node, and return success if it is satisfied or try its right node if not satisfied. The sequence node is used to execute an action when its corresponding condition is satisfied. For the parallel node, it corresponds to the condition that more than one action should be taken. Fig. 8 demonstrates a BT model example where we may see that an action level at least contains a selector node, a condition node, a sequence node, and a action node.

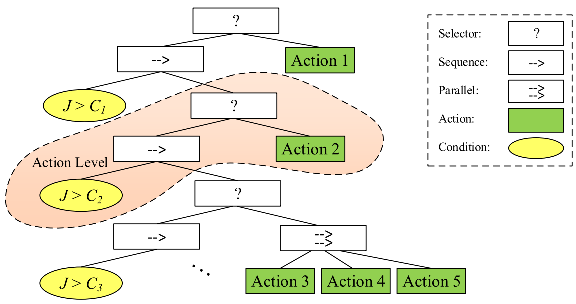



**FIGURE 8.** **A BT model example.**

### *H. REALISTIC VIRTUAL ENVIRONMENT CONSTRUCTION*

According to the above discussion, the EA environment is highly dynamic and hard to precisely predict. Therefore, in order to make the task robots adapt to a constant new environment, an effective way is training the ''sense-decide-act'' behavior chain of each task robot in a variety of scenarios. However, the cost of building many realistic scenarios is unaffordable and the training process is inefficient. Therefore, it may be a good alternative to make robots trained in a virtual environment.

For each robot, the optimality of its trained results largely relies on the fidelity of the virtual environment. For example, in the decide stage, RL needs to adjust the robot's action policy by gaining feedback through interaction with the environment. However, building a realistic electromagnetic-geographical scenario may pose a major challenge. In battle games, the research on geographical scenario generation is relatively mature. In recent years, some artificial intelligence companies cooperated with game companies to develop the RL simulation platforms, e.g., Universe [101], SC2LE [102], etc. Therefore, the game geographical scenario generation can provide an important reference for the construction of the virtual geographical environment. Furthermore, computer graphics combined with deep learning has done a lot of work on realistic geographical scenario generation, including large-scale outdoor scenarios [103] and small-scale indoor scenarios [104].

Based on [103], we try to propose a large-scale geographical scenario generation example with conditional generative adversarial network (cGAN), as shown in Fig. 9. Firstly, based on the public digital elevation model (DEM) dataset, we need to label the main geographical elements, e.g., rivers, crests, etc, which may build a terrain feature map (a kind of point-line-surface mesh maps). In addition, in order to enhance the generalized representation ability of the terrain feature map, it is necessary to be blurred and down-sampled. Then, the labeled feature terrain map and the original terrain environment are put into cGAN for training. The generator is implemented with a convolutional autoencoder (CAE), while the discriminator is using a convolutional neural network (CNN). Then, according to a specific task, we may construct random terrain feature maps and put them into the generator to obtain a generated terrain environment. Finally, ground objects (such as buildings) can be randomly generated according to the available ground areas provided by the terrain feature map, and merged with the terrain environment in the form of layers to construct a virtual geographical scenario.

The geographical scenario is static in each training process of RL, but the electromagnetic environment may constantly change with the movements of robots. Therefore, for the electromagnetic environment simulation, on one hand it is necessary to keep the simulation accuracy, on the other it needs to meet the real-time requirement of the training process. Current researches about electromagnetic environment construction mainly focus on two kinds of models: stochastic model and deterministic model. The stochastic model simulates the wireless channel based on the statistical characteristics of signals. This type of method has a small amount of computation, but cannot accurately predict the channel quality at a given location [7]. The deterministic models mainly include finite-difference time-domain (FDTD) method [105], ray tracing method [106], etc. This type of method can accurately describe the characteristics of the channel, but requires complete environmental information and is computationally intensive. Considering compromising on the accuracy and the real-time requirement, an alternative method is ray tracing. However, when the number of robots increases, the computation time of ray tracing will directly affect the efficiency of the evolution of the whole system. Therefore, it is necessary to adopt parallel programming technology to accelerate the construction of electromagnetic environment based on multi-core and many-core technologies, e.g., [107]–[110].

### *I. SOFTWARE ARCHITECTURE*

In above subsections, we have summarized the enabling techniques for solving multi-robot coordination in EA environment. While in practical robotic applications, each technique above is implemented by specific algorithms which are embedded in software. Robotic software architecture design can be seen as software engineering, which has the characteristics of needing to interact with an uncertain and dynamic environment [111]. In order to adapt to such situations, the architecture design should follow the principles of modularity and hierarchy, which motivates the robot system to be designed as distributed component-based systems [111]–[113]. In the component-based architecture, each component has the independent functionality which corresponds to a class of algorithms. The typical implementations include Yet Another Robot Platform (YARP) [114], Open Robot Control Software (OROCOS) [115], Robot Operating System (ROS) [116], etc. A detailed discussion about traditional robotic software architecture can be seen in [111].

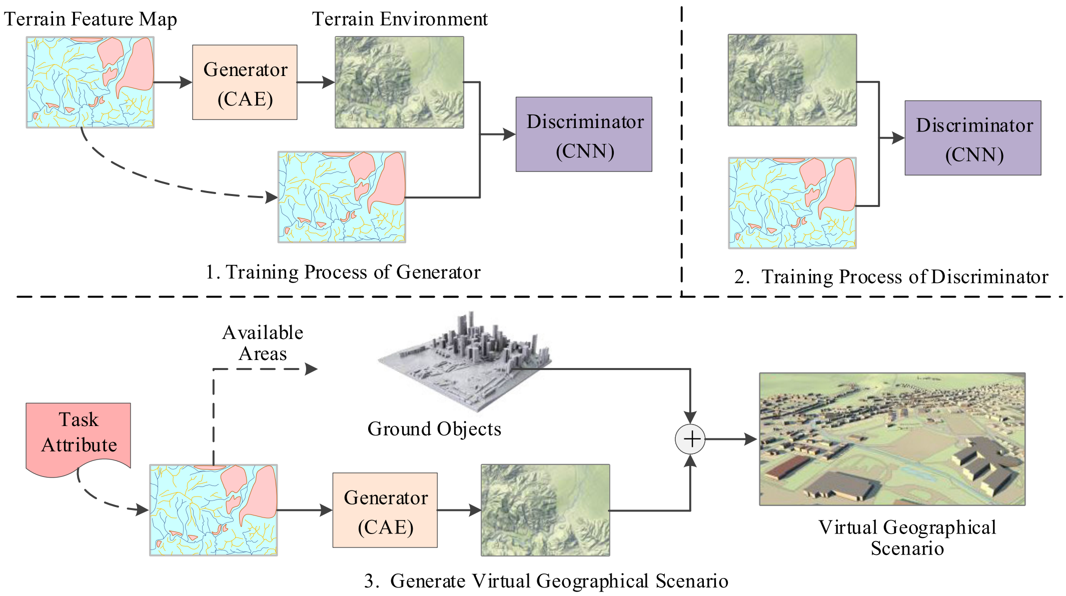



**FIGURE 9.** A large-scale geographical scenario generation example implemented with cGAN and its training process.

Nowadays, the robot system tends to be more complex, which makes the system need more components to complete a specific task [117]. For a certain scenario in EA environment, there are two main issues that need to be addressed. The first is how to modularize various kinds of components according to their functions. An alternative way is to classify the components according to the "sense-decide-act" behavior chain. In terms of technical implementation, plugin is an effective way to realize components and can be developed independently from specific applications [118]. Currently, plugin techniques are widely used in the development of robot functions [119], [120], and some plugin-based architectures are produced, e.g. OpenRAVE [121], OpenMRH [122], micROS [118], etc.

The second issue is how to design efficient component management and scheduling mechanisms to cope with the situation changes in EA environment. The hierarchical decomposition of robotic systems can be decomposed from the dimensions of time, space, task, etc., but which dimension is better does not form a consistent view [111]. In [117], the authors point out that the maintenance burdens and resource limitations are two main restrictions for the computing platforms on robots. In order to solve this problem, they propose Rorg, a tool to manage the components and resource by adopting Linux containers. Moreover, the works of [123] fully consider the performance of the low-power CPU equipped on robots, and propose MPT, a template-based framework that can generate robot-specific motion planning code. Compared with the well-known OMPL [124], MPT may need less wall-clock time and memory for specific robots, e.g. Nao. In [125], a hierarchical robotics framework MaestROS is proposed where different components can be orchestrated to perform complex behaviors. MaestROS can support high-level instruction inputs, such as natural language and demonstrations, and can train the robots to perform a task with these instructions. As the latest version of ROS, ROS2 has integrated the behavior tree model into its navigation system [126], [127].

## IV. CONCLUSION

In this survey, we first proposed the definition of the EA environment, and then suggested adopting the model of "sense-decide-act" behavior chain to solve the multi-robot coordination problem in EA environment. Following this model, we summarized four challenges from each stage of the behavior chain and environment adaptation, respectively. Afterwards, for each challenge, we detailedly reviewed the related enabling techniques and their application conditions. Finally, the typical software architecture designs were introduced to promote the integration of the enabling techniques.

## REFERENCES

[1] B. Gates, "A robot in every home," *Sci. Amer.*, vol. 296, no. 1, pp. 58–65, Jan. 2007.
[2] M. Lindhe and K. Johansson, "Using robot mobility to exploit multipath fading," *IEEE Wireless Commun.*, vol. 16, no. 1, pp. 30–37, Feb. 2009.
[3] Y. Yan and Y. Mostofi, "To go or not to go: On energy-aware and communication-aware robotic operation," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 3, pp. 218–231, Sep. 2014.
[4] K. Cesare, R. Skeele, S.-H. Yoo, Y. Zhang, and G. Hollinger, "Multi-UAV exploration with limited communication and battery," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Seattle, WA, USA, May 2015, pp. 2230–2235.
[5] G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, R. Merrifield, B. J. Nelson, B. Scassellati, M. Taddeo, R. Taylor, M. Veloso, Z. L. Wang, and R. Wood, "The grand challenges of science robotics," *Sci. Robot.*, vol. 3, no. 14, Jan. 2018.
[6] J. Stephan, J. Fink, V. Kumar, and A. Ribeiro, "Concurrent control of mobility and communication in multirobot systems," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1248–1254, Oct. 2017.

[7] M. Malmirchegini and Y. Mostofi, "On the spatial predictability of communication channels," *IEEE Trans. Wireless Commun.*, vol. 11, no. 3, pp. 964–978, Mar. 2012.
[8] C. Dixon and E. W. Frew, "Optimizing cascaded chains of unmanned aircraft acting as communication relays," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 5, pp. 883–898, Jun. 2012.
[9] X. Liu, Y. Xu, L. Jia, Q. Wu, and A. Anpalagan, "Anti-jamming communications using spectrum waterfall: A deep reinforcement learning approach," *IEEE Commun. Lett.*, vol. 22, no. 5, pp. 998–1001, May 2018.
[10] T. Ulversoy, "Software defined radio: Challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 4, pp. 531–550, 4th Quart., 2010.
[11] C. F. Kurtz and D. J. Snowden, "The new dynamics of strategy: Sense-making in a complex and complicated world," *IBM Syst. J.*, vol. 42, no. 3, pp. 462–483, 2003.
[12] J. Rojas, S. Luo, D. Zhu, Y. Du, H. Lin, Z. Huang, W. Kuang, and K. Harada, "Online robot introspection via wrench-based action grammars," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Vancouver, BC, Canada, Sep. 2017, pp. 5429–5436.
[13] J. Dong, E. Nelson, V. Indelman, N. Michael, and F. Dellaert, "Distributed real-time cooperative localization and mapping using an uncertainty-aware expectation maximization approach," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Seattle, WA, USA, May 2015, pp. 5807–5814.
[14] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal, and G. S. Sukhatme, "Interactive perception: Leveraging action in perception and perception in action," *IEEE Trans. Robot.*, vol. 33, no. 6, pp. 1273–1291, Dec. 2017.
[15] A. Byravan and D. Fox, "SE3-Nets: Learning rigid body motion using deep neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Singapore, May 2017, pp. 173–180.
[16] J. S. McGrew, J. P. How, B. Williams, and N. Roy, "Air-combat strategy using approximate dynamic programming," *J. Guid., Control, Dyn.*, vol. 33, no. 5, pp. 1641–1654, Sep. 2010.
[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.
[18] T. F. Cox and M. A. Cox, *Multidimensional Scaling*. London, U.K.: Chapman & Hall, 2000.
[19] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.
[20] Z. Chen, A. Jacobson, N. Sünderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford, "Deep learning features at scale for visual place recognition," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Singapore, May 2017, pp. 3223–3230.
[21] M. Freitag, S. Amiriparian, S. Pugachevskiy, N. Cummins, and B. Schuller, "audeep: Unsupervised learning of representations from audio with deep recurrent neural networks," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6340–6344, Apr. 2017.
[22] D. Cockbum, J.-P. Roberge, T.-H.-L. Le, A. Maslyczyk, and V. Duchaine, "Grasp stability assessment through unsupervised feature learning of tactile images," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Singapore, May 2017, pp. 2238–2244.
[23] D. Ha and J. Schmidhuber, "World models," 2018, *arXiv:1803.10122*. [Online]. Available: http://arxiv.org/abs/1803.10122
[24] L. Sun, Z. Yan, S. M. Mellado, M. Hanheide, and T. Duckett, "3DOF pedestrian trajectory prediction learned from long-term autonomous mobile robot deployment data," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 1–7.
[25] S. Ammoun and F. Nashashibi, "Real time trajectory prediction for collision risk estimation between vehicles," in *Proc. IEEE 5th Int. Conf. Intell. Comput. Commun. Process.*, Cluj-Napoca, Romania, Aug. 2009, pp. 417–422.
[26] G. Williams, B. Goldfain, P. Drews, J. M. Rehg, and E. A. Theodorou, "Best response model predictive control for agile interactions between autonomous ground vehicles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 2403–2410.
[27] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, 2nd Quart., 2016.
[28] R. C. Dixon, *Spread Spectrum Systems With Commercial Applications*. Hoboken, NJ, USA: Wiley, 1994.
[29] A. Alexiou and M. Haardt, "Smart antenna technologies for future wireless systems: Trends and challenges," *IEEE Commun. Mag.*, vol. 42, no. 9, pp. 90–97, Sep. 2004.
[30] L. Zheng and D. N. C. Tse, "Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1073–1096, May 2003.
[31] Y.-C. Liang, K.-C. Chen, G. Ye Li, and P. Mahonen, "Cognitive radio networking and communications: An overview," *IEEE Trans. Veh. Technol.*, vol. 60, no. 7, pp. 3386–3407, Sep. 2011.
[32] I. Kakalou, K. E. Psannis, P. Krawiec, and R. Badea, "Cognitive radio network and network service chaining toward 5G: Challenges and requirements," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 145–151, Nov. 2017.
[33] F. Hu, B. Chen, and K. Zhu, "Full spectrum sharing in cognitive radio networks toward 5G: A survey," *IEEE Access*, vol. 6, pp. 15754–15776, Feb. 2018.
[34] B. Wang and K. J. Ray Liu, "Advances in cognitive radio networks: A survey," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 5–23, Feb. 2010.
[35] *GNURadio*. [Online]. Available: http://www.gnu.org/software/gnuradio/
[36] J. Bard and V. J. Kovarik, Jr., *Software Defined Radio: The Software Communications Architecture*. Hoboken, NJ, USA: Wiley, 2007.
[37] P. D. Sutton, J. Lotze, H. Lahlou, S. A. Fahmy, K. E. Nolan, B. Ozgul, T. W. Rondeau, J. Noguera, and L. E. Doyle, "Iris: An architecture for cognitive radio networking testbeds," *IEEE Commun. Mag.*, vol. 48, no. 9, pp. 114–122, Sep. 2010.
[38] *USRP*. [Online]. Available: https://www.ettus.com/
[39] Y. Zhao, J. Pradhan, J. Huang, Y. Luo, and L. Pu, "Joint energy-and-bandwidth spectrum sensing with GNU radio and USRP," *ACM SIGAPP Appl. Comput. Rev.*, vol. 14, no. 4, pp. 40–49, Jan. 2015.
[40] Y. Wu, B. Zhang, X. Yi, and Y. Tang, "Collaborative communication in multi-robot surveillance based on indoor radio mapping," in *Proc. Int. Conf. Collaborative Comput., Netw., Appl. Worksharing*, Beijing, China, Nov. 2016, pp. 211–220.
[41] M. M. Rahman, L. Bobadilla, and B. Rapp, "Establishing line-of-sight communication via autonomous relay vehicles," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Baltimore, MD, USA, Nov. 2016, pp. 642–647.
[42] T. Alam, M. M. Rahman, L. Bobadilla, and B. Rapp, "Multi-vehicle patrolling with limited visibility and communication constraints," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Baltimore, MD, USA, Oct. 2017, pp. 465–470.
[43] T. Alam, M. M. Rahman, P. Carrillo, L. Bobadilla, and B. Rapp, "Stochastic multi-robot patrolling with limited visibility," *J. Intell. Robot. Syst.*, vol. 97, no. 2, pp. 411–429, Feb. 2020.
[44] B. Zhang, Y. Wu, X. Yi, and X. Yang, "Joint communication-motion planning in wireless-connected robotic networks: Overview and design guidelines," 2015, *arXiv:1511.02299*. [Online]. Available: http://arxiv.org/abs/1511.02299
[45] A. Ghaffarkhah and Y. Mostofi, "Communication-aware motion planning in mobile networks," *IEEE Trans. Autom. Control*, vol. 56, no. 10, pp. 2478–2485, Oct. 2011.
[46] A. Li, W. Luo, S. Nagavalli, and K. Sycara, "Decentralized coordinated motion for a large team of robots preserving connectivity and avoiding collisions," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Singapore, May 2017, pp. 1505–1511.
[47] M. Lu, "Rendezvous with connectivity preservation of mobile agents subject to uniform time-delays," *Automatica*, vol. 88, pp. 31–37, Feb. 2018.
[48] M. M. Zavlanos and G. J. Pappas, "Potential fields for maintaining connectivity of mobile networks," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 812–816, Aug. 2007.
[49] L. Siligardi, J. Panerati, M. Kaufmann, M. Minelli, C. Ghedini, G. Beltrame, and L. Sabattini, "Robust area coverage with connectivity maintenance," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Montreal, QC, Canada, May 2019, pp. 2202–2208.
[50] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in *Proc. IEEE Conf. Comput. Commun.*, vol. 3, Anchorage, AK, USA, Apr. 2001, pp. 1380–1387.
[51] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Ad Hoc Sensor Wireless Netw.*, vol. 1, nos. 1–2, pp. 89–124, Mar. 2005.
[52] Y.-C. Wang, C.-C. Hu, and Y.-C. Tseng, "Efficient placement and dispatch of sensors in a wireless sensor network," *IEEE Trans. Mobile Comput.*, vol. 7, no. 2, pp. 262–274, Feb. 2007.
[53] V. Ramasamy, "Mobile wireless sensor networks: An overview," *Wireless Sensor Networks: Insights and Innovations*. Croatia, Balkans: InTech, Oct. 2017, pp. 3–19.

[54] Z. Liao, J. Wang, S. Zhang, J. Cao, and G. Min, "Minimizing movement for target coverage and network connectivity in mobile sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 7, pp. 1971–1983, Jul. 2014.

[55] N. Tamboli and M. Younis, "Coverage-aware connectivity restoration in mobile sensor networks," *J. Netw. Comput. Appl.*, vol. 33, no. 4, pp. 363–374, Jul. 2010.

[56] G. Tan, S. A. Jarvis, and A.-M. Kermarrec, "Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks," *IEEE Trans. Mobile Comput.*, vol. 8, no. 6, pp. 836–848, Jun. 2009.

[57] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, "Graph-theoretic connectivity control of mobile robot networks," *Proc. IEEE*, vol. 99, no. 9, pp. 1525–1540, Sep. 2011.

[58] M. N. Rooker and A. Birk, "Multi-robot exploration under the constraints of wireless networking," *Control Eng. Pract.*, vol. 15, no. 4, pp. 435–445, Apr. 2007.

[59] K. Khateri, M. Pourgholi, M. Montazeri, and L. Sabattini, "A comparison between decentralized local and global methods for connectivity maintenance of multi-robot networks," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 633–640, Apr. 2019.

[60] J. Panerati, L. Gianoli, C. Pinciroli, A. Shabah, G. Nicolescu, and G. Beltrame, "From swarms to stars: Task coverage in robot swarms with connectivity constraints," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 7674–7681.

[61] M. Ji and M. Egerstedt, "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 693–703, Aug. 2007.

[62] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[63] A. Ghaffarkhah and Y. Mostofi, "Path planning for networked robotic surveillance," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3560–3575, Jul. 2012.

[64] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

[65] Y. Wu, B. Zhang, S. Yang, X. Yi, and X. Yang, "Energy-efficient joint communication-motion planning for relay-assisted wireless robot surveillance," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Atlanta, GA, USA, May 2017, pp. 1–9.

[66] Y. Marchukov and L. Montano, "Fast and scalable multi-robot deployment planning under connectivity constraints," in *Proc. IEEE Int. Conf. Auto. Robot Syst. Competitions (ICARSC)*, Porto, Portugal, Apr. 2019, pp. 1–7.

[67] Y. Yan and Y. Mostofi, "Robotic router formation in realistic communication environments," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 810–827, Aug. 2012.

[68] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for mobile relaying systems," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Washington, DC, USA, Dec. 2016, pp. 1–6.

[69] J. Fink, A. Ribeiro, and V. Kumar, "Robust control for mobility and wireless communication in cyber–physical systems with application to robot teams," *Proc. IEEE*, vol. 100, no. 1, pp. 164–178, Jan. 2012.

[70] J. Fink, A. Ribeiro, and V. Kumar, "Robust control of mobility and communications in autonomous robot teams," *IEEE Access*, vol. 1, pp. 290–309, May 2013.

[71] S. S. Ponda, L. B. Johnson, A. N. Kopeikin, H.-L. Choi, and J. P. How, "Distributed planning strategies to ensure network connectivity for dynamic heterogeneous teams," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 5, pp. 861–869, Jun. 2012.

[72] N. Goddemeier, K. Daniel, and C. Wietfeld, "Role-based connectivity management with realistic Air-to-Ground channels for cooperative UAVs," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 5, pp. 951–963, Jun. 2012.

[73] F. El-Moukaddem, E. Torng, and G. Xing, "Mobile relay configuration in data-intensive wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 2, pp. 261–273, Feb. 2013.

[74] B.-C. Min, Y. Kim, S. Lee, J.-W. Jung, and E. T. Matson, "Finding the optimal location and allocation of relay robots for building a rapid end-to-end wireless communication," *Ad Hoc Netw.*, vol. 39, pp. 23–44, Mar. 2016.

[75] D. H. Choi, S. H. Kim, and D. K. Sung, "Energy-efficient maneuvering and communication of a single UAV-based relay," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 3, pp. 2320–2327, Jul. 2014.

[76] O. J. Faqir, Y. Nie, E. C. Kerrigan, and D. Gündüz, "Energy-efficient communication in mobile aerial relay-assisted networks using predictive control," *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 197–202, Nov. 2018.

[77] S. Zeng, H. Zhang, K. Bian, and L. Song, "UAV relaying: Power allocation and trajectory optimization using Decode-and-Forward protocol," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Kansas City, MO, USA, May 2018, pp. 1–6.

[78] Y. Pei and M. W. Mutka, "Steiner traveler: Relay deployment for remote sensing in heterogeneous multi-robot exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, Saint Paul, MN, USA, May 2012, pp. 1551–1556.

[79] P. Ladosz, H. Oh, G. Zheng, and W.-H. Chen, "A hybrid approach of learning and model-based channel prediction for communication relay uavs in dynamic urban environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, Montreal, Canada, May 2019, pp. 1–8.

[80] P. Ladosz, H. Oh, and W.-H. Chen, "Prediction of air-to-ground communication strength for relay UAV trajectory planner in urban environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Vancouver, BC, Canada, Sep. 2017, pp. 6831–6836.

[81] K.-H. Kim, K. G. Shin, and D. Niculescu, "Mobile autonomous router system for dynamic (Re)formation of wireless relay networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1828–1841, Sep. 2012.

[82] Y. Wu, B. Zhang, X. Yi, and Y. Tang, "Communication-motion planning for wireless relay-assisted multi-robot system," *IEEE Wireless Commun. Lett.*, vol. 5, no. 6, pp. 568–571, Dec. 2016.

[83] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[84] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 26–38, Nov. 2017.

[85] J. Feng, H. Li, M. Huang, S. Liu, W. Ou, Z. Wang, and X. Zhu, "Learning to collaborate: Multi-scenario ranking via multi-agent reinforcement learning," in *Proc. World Wide Web Conf.*, Lyon, France, Apr. 2018, pp. 1939–1948.

[86] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian, "Deep decentralized multi-task multi-agent reinforcement learning under partial observability," in *Proc. 34th Int. Conf. Mach. Learn.*, Sydney, NSW, Australia, vol. 70, Aug. 2017, pp. 2681–2690.

[87] Y. Yang, R. Luo, M. Li, M. Zhou, W. Zhang, and J. Wang, "Mean field multi-agent reinforcement learning," in *Proc. 35th Int. Conf. Mach. Learn.*, Stockholm, Sweden, vol. 80, Jul. 2018, pp. 5571–5580.

[88] J. N. Foerster, Y. M. Assael, N. de Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, Barcelona, Spain, Dec. 2016, pp. 2145–2153.

[89] R. Lowe, Y. Wu, A. Tamar, J. Harb, O. P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Proc. Adv. Neural Inf. Process. Syst.*, Long Beach, CA, USA, Dec. 2017, pp. 6379–6390.

[90] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *Proc. Int. Conf. Auto. Agents Multiagent Syst.*, Sao Paulo, Brazil, Nov. 2017, pp. 66–83.

[91] Y. Kantaros and M. M. Zavlanos, "Distributed intermittent communication control of mobile robot networks under time-critical dynamic tasks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 1–9.

[92] Y. Kantaros, M. Guo, and M. M. Zavlanos, "Temporal logic task planning and intermittent connectivity control of mobile robot networks," *IEEE Trans. Autom. Control*, vol. 64, no. 10, pp. 4105–4120, Oct. 2019.

[93] B. L. Wellman, S. Dawson, J. de Hoog, and M. Anderson, "Using rendezvous to overcome communication limitations in multirobot exploration," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Anchorage, AK, USA, Oct. 2011, pp. 2401–2406.

[94] G. A. Hollinger and S. Singh, "Multirobot coordination with periodic connectivity: Theory and experiments," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 967–973, Aug. 2012.

[95] F. Yan, K. Di, J. Jiang, Y. Jiang, and H. Fan, "Efficient decision-making for multiagent target searching and occupancy in an unknown environment," *Robot. Auto. Syst.*, vol. 114, pp. 41–56, Apr. 2019.

[96] G. Hollinger, S. Singh, J. Djugash, and A. Kehagias, "Efficient multi-robot search for a moving target," *Int. J. Robot. Res.*, vol. 28, no. 2, pp. 201–219, Feb. 2009.

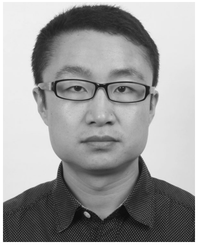
[97] I. Vandermeulen, R. Gross, and A. Kolling, "Re-establishing communication in teams of mobile robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Madrid, Spain, Oct. 2018, pp. 7947–7954.

[98] C. L. Nehaniv, K. Dautenhahn, and K. Dautenhahn, *Imitation in Animals and Artifacts*. Cambridge, MA, USA: MIT Press, 2002.
[99] R. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robot. Autom.*, vol. 2, no. 1, pp. 14–23, Mar. 1986.
[100] M. Colledanchise and P. Ögren, "How behavior trees modularize hybrid control systems and generalize sequential behavior compositions, the subsumption architecture, and decision trees," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 372–389, Apr. 2016.
[101] (2016). *OpenAI*. [Online]. Available: https://blog.openai.com/universe/
[102] O. Vinyals *et al.*, "StarCraft II: A new challenge for reinforcement learning," 2017, *arXiv:1708.04782*. [Online]. Available: https://arxiv.org/abs/1708.04782
[103] É. Guérin, J. Digne, É. Galin, A. Peytavie, C. Wolf, B. Benes, and B. Martinez, "Interactive example-based terrain authoring with conditional generative adversarial networks," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 1–13, Nov. 2017.
[104] K. Wang, M. Savva, A. X. Chang, and D. Ritchie, "Deep convolutional priors for indoor scene synthesis," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 1–14, Jul. 2018.
[105] M. M. Rana and A. S. Mohan, "Segmented-Locally-One-Dimensional-FDTD method for EM propagation inside large complex tunnel environments," *IEEE Trans. Magn.*, vol. 48, no. 2, pp. 223–226, Feb. 2012.
[106] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications," *IEEE Access*, vol. 3, pp. 1089–1100, Jul. 2015.
[107] L. Wang, K. Guan, B. Ai, G. Li, D. He, R. He, L. Tian, J. Dou, and Z. Zhong, "An accelerated algorithm for ray tracing simulation based on high-performance computation," in *Proc. 11th Int. Symp. Antennas, Propag. EM Theory (ISAPE)*, Guilin, China, Oct. 2016, pp. 512–515.
[108] D. He, B. Ai, K. Guan, L. Wang, Z. Zhong, and T. Kürner, "The design and applications of high-performance ray-tracing simulation platform for 5G and beyond wireless communications: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 10–27, 1st Quart., 2018.
[109] B. R. Epstein and D. L. Rhodes, "GPU-accelerated ray tracing for electromagnetic propagation analysis," in *Proc. IEEE Int. Conf. Wireless Inf. Technol. Syst.*, Honolulu, HI, USA, Aug. 2010, pp. 1–4.
[110] S. G. Parker, J. Bigler, A. Dietrich, H. Friedrich, J. Hoberock, D. Luebke, D. McAllister, M. McGuire, K. Morley, and A. Robison, "OptiX: A general purpose ray tracing engine," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–13, Jul. 2010.
[111] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Berlin, Germany: Springer, 2016.
[112] D. Brugali and P. Scandurra, "Component-based robotic engineering (Part I) [Tutorial]," *IEEE Robot. Autom. Mag.*, vol. 16, no. 4, pp. 84–96, Dec. 2009.
[113] D. Brugali and A. Shakhimardanov, "Component-based robotic engineering (Part II)," *IEEE Robot. Autom. Mag.*, vol. 17, no. 1, pp. 100–112, Mar. 2010.
[114] G. Metta, P. Fitzpatrick, and L. Natale, "YARP: Yet another robot platform," *Int. J. Adv. Robotic Syst.*, vol. 3, no. 1, p. 8, Mar. 2006.
[115] H. Bruyninckx, "Open robot control software: The OROCOS project," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Seoul, South Korea, May 2001, pp. 2523–2528.
[116] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, Kobe, Japan, May 2009, pp. 1–6.
[117] S. Wang, X. Liu, J. Zhao, and H. I. Christensen, "Rorg: Service robot software management with Linux containers," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Montreal, QC, Canada, May 2019, pp. 584–590.
[118] X. Yang, H. Dai, X. Yi, Y. Wang, S. Yang, B. Zhang, Z. Wang, Y. Zhou, and X. Peng, "MicROS: A morphable, intelligent and collective robot operating system," *Robot. Biomimetics*, vol. 3, no. 1, pp. 1–9, Dec. 2016.
[119] A. Koubaa, H. Bennaceur, I. Chaari, S. Trigui, A. Ammar, M.-F. Sriti, M. Alajlan, O. Cheikhrouhou, and Y. Javed, "Integration of global path planners in ROS," in *Robot Path Planning and Cooperation*. Cham, Switzerland: Springer, 2018, pp. 83–102.
[120] K. Takaya, T. Asai, V. Kroumov, and F. Smarandache, "Simulation environment for mobile robots testing using ROS and gazebo," in *Proc. 20th Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Sinaia, Romania, Oct. 2016, pp. 96–101.
[121] R. Diankov and J. Kuffner, "Openrave: A planning architecture for autonomous robotics," Robot. Inst., Pittsburgh, PA, USA, Tech. Rep. CMU-RI-TR-08-34, Jul. 2008.
[122] F. Sanfilippo and K. Y. Pettersen, "OpenMRH: A modular robotic hand generator plugin for OpenRAVE," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Zhuhai, China, Dec. 2015, pp. 1–6.
[123] J. Ichnowski and R. Alterovitz, "Motion planning templates: A motion planning framework for robots with low-power CPUs," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Montreal, QC, Canada, May 2019, pp. 612–618.
[124] I. A. Sucan, M. Moll, and L. E. Kavraki, "The open motion planning library," *IEEE Robot. Autom. Mag.*, vol. 19, no. 4, pp. 72–82, Dec. 2012.
[125] A. Munawar, G. De Magistris, T.-H. Pham, D. Kimura, M. Tatsubori, T. Moriyama, R. Tachibana, and G. Booch, "MaestROB: A robotics framework for integrated orchestration of low-level control and high-level reasoning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 527–534.
[126] *BehaviorTree.CPP*. [Online]. Available: https://www.behaviortree.dev/
[127] *Navigation2*. [Online]. Available: https://index.ros.org/r/navigation2/

**YUNLONG WU** received the B.S. degree in information security from Wuhan University, Wuhan, China, in 2011, and the M.S. degree in computer science and technology, and the Ph.D. degree in microelectronics and solid electronics from the National University of Defense Technology, Changsha, China, in 2013 and 2017, respectively. Since 2018, he has been an Assistant Professor with the National Innovation Institute of Defense Technology, Beijing, China. His main research interests include robotic communications, multiagent systems, and swarm intelligence.

**XIAOGUANG REN** received the B.S., M.S., and Ph.D. degrees in computer science and technology from the National University of Defense Technology, Changsha, China, in 2008, 2011, and 2014, respectively. Since 2019, he has been an Associate Professor with the National Innovation Institute of Defense Technology, Beijing, China. His main research interests include computer architecture, parallel computing, and robot operating systems.

**HAO ZHOU** received the B.S. and M.S. degrees in computer science and technology from the National University of Defense Technology, Changsha, Hunan, China, in 2009 and 2011, respectively, and the Ph.D. degree in computer science and engineering from the University of New South Wales, Sydney, Australia, in 2016. Since 2018, he has been an Assistant Professor with the National Innovation Institute of Defense Technology, Beijing, China. His main research interests include robot operating systems, swarm intelligence, computer architecture, programming language, and compiler.



**YANZHEN WANG** received the Ph.D. degree in computer science from the National University of Defense Technology, China, in 2011. He is currently an Associate Researcher with the National Innovation Institute of Defense Technology (NIIDT), China. His research interests include robotics software, artificial intelligence, and virtual reality.

**XIAODONG YI** received the Ph.D. degree from the National University of Defense Technology, Changsha, China, in 2006. He is currently a Full Professor with the National Innovation Institute of Defense Technology. He has devoted to Kylin operating systems, which was used in Milkway supercomputers, for over ten years. His research interests include operating systems, high performance computing, robotics software, and so on.