Subteaming and Adaptive Formation Control for Coordinated Multi-Robot Navigation

Author Names Omitted for Anonymous Review. Paper-ID 188

***Abstract*—Coordinated multi-robot navigation is a fundamental capability for robots to operate as a team in diverse environments. During navigation, robot teams usually need to maintain specific formations, such as circular formations in order to protect human teammates positioned at the center. However, in complex scenarios such as narrow corridors, rigidly preserving the predefined team formations can become infeasible. Therefore, robot teams must have the capability for dynamically splitting into smaller subteams and adaptively controlling the subteams to navigate through these complex scenarios while preserving suitable formations. In this paper, we introduce a novel method for *SubTeaming and Adaptive Formation* (STAF) to enable coordinated multi-robot navigation in complex and challenging scenarios. STAF is built upon a unified hierarchical learning framework that incorporates three levels of robot learning: (1) high-level deep graph cut for team splitting, (2) intermediate-level graph learning for facilitating coordinated navigation among subteams, and (3) low-level policy learning for controlling individual mobile robots to reach their goal positions while avoiding collisions. In order to evaluate and validate STAF, we conducted extensive experiments in both indoor and outdoor environments using robotics simulations and physical robot teams. Experimental results have demonstrated that STAF enables the novel capability for subteaming and adaptive formation control, and achieves promising performance in coordinated multi-robot navigation through complex and challenging scenarios.**

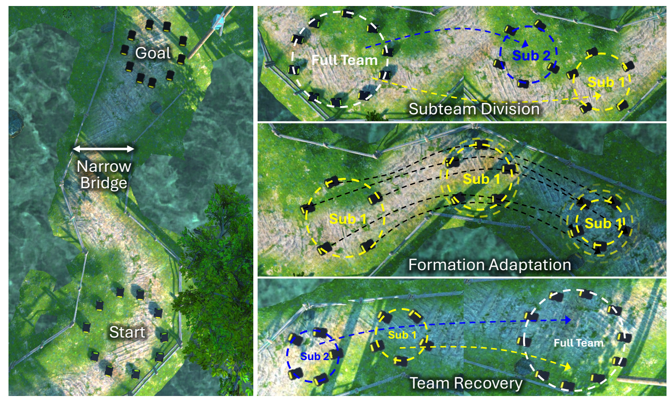


Fig. 1. In this coordinated multi-robot navigation scenario, a team of ten Warthog robots, moving in a circular formation, encounters a narrow bridge over a creek in outdoor field environments. The bridge is too narrow for the entire robot team to cross at once. The robots must be capable of (1) dividing into smaller subteams and (2) adapting their formations for the subteams to pass through the bridge while maintaining a specific formation. Our novel STAF approach enables subteam division, adaptive formation control, and team recovery within a unified hierarchical learning framework, which significantly enhances coordinated navigation capabilities.

I. Introduction

Multi-robot systems have attracted significant attention in recent years due to their unique advantages, such as redundancy [1], parallelism [2], and scalability [3]. These characteristics make them indispensable for addressing complex, large-scale tasks that are usually infeasible for single robots. Coordinated multi-robot navigation is a fundamental capability that allows teams of robots to traverse environments in a synchronized manner and reach goal positions collectively [4]. This coordination is particularly critical in dynamic and complex environments with uncertainty, where robots must adapt their actions based on the behavior of teammates and external conditions. Effective coordinated multi-robot navigation is necessary not only for achieving operational efficiency but also for ensuring the safety and reliability of robot teams when deployed in a variety of real-world applications, such as search and rescue [5–7], space exploration [8, 9], and intelligent transportation [10, 11].

During coordinated navigation, robots are often required to maintain specific formations tailored to their missions, such as circular formations to protect an important agent at the center or line formations to maximize coverage during search operations. However, rigid adherence to predefined formations can hinder effective navigation in certain scenarios. For instance, Figure 1 depicts a team of ten robots in a circular formation encountering a corridor too narrow for the entire team to pass through. In such challenging scenarios, the team must be capable of dynamically dividing into smaller units that operate both independently and cohesively (i.e., *subteaming*) and controlling the subteams to pass through the narrow corridor while adaptively maintaining a specific formation (i.e., *adaptive formation control*).

Due to the importance of coordinated multi-robot navigation, a wide range of techniques have been developed in recent years. Traditional approaches, including classical planning methods [12], game-theoretic methods [13, 14], and optimization-based methods [15], often face challenges such as high computational costs and limitations in handling complex, dynamic environments, including narrow corridors and other constrained spaces. Recently, learning-based approaches have shown considerable promise. For example, deep neural networks [16, 17] have been used to model multi-robot systems; multi-agent reinforcement learning [18, 19] has been applied to improve coordination and enable navigation of multi-agent systems. However, previous learning-based methods [16–19] have not addressed adaptive formation control within coordinated multi-robot navigation, particularly in complex scenarios like traversing narrow corridors. Several methods have been designed to address robot subteaming, such as using graph cuts for team division [20, 21] and mixed-integer programming for subteam task allocation [1, 22, 23]. However, these methods generally focus on team division alone, without the capability to control the subteams or individual robots, which makes them unsuitable for addressing coordinated navigation.

To address the challenges above and enable effective coordinated multi-robot navigation in complex scenarios where the entire robot team cannot pass through, we introduce a novel approach called *SubTeaming and Adaptive Formation* (**STAF**), which offers new capabilities for subteam division, formation adaptation, and team recovery. Specifically, we design a graph representation to encode a team of robots, where each node represents a robot along with its associated attributes, such as its position, velocity, goal, and distance to obstacles, and each edge represents the spatial relationships between pairs of robots. Our STAF approach integrates three levels of robot learning into a hierarchical framework. At the high level, given the graph representation of a robot team, STAF performs deep graph cuts to divide the entire robot team into subteams. The intermediate level of STAF focuses on learning the coordination of these robot subteams for navigation, which develops a graph neural network with learnable message sharing to coordinate robots within a subteam, while generating graph embeddings to encode the subteam context. Finally, at the low level, given these embeddings, STAF employs reinforcement learning to learn a navigation policy that controls each individual robot to adaptively maintain subteam formation, reach the goal position, and avoid collisions.

Our primary contribution is the introduction of the novel STAF method to enable a new multi-robot navigation capability of subteaming and formation adaptation. The specific novelties of this paper include:

- This work introduces one of the first problem formulations and learning-based solutions for subteaming and formation adaptation in multi-robot coordinated navigation. It enables new multi-robot capabilities, including subteam division, formation adaptation, and team recovery, allowing a team of robots to navigate complex environments in a coordinated manner, particularly narrow corridors where maintaining original formation is infeasible.
- We introduce a novel hierarchical robot learning method that simultaneously integrates high-level deep graph cut for subteaming, intermediate-level graph learning for subteam coordination and adaptive formation control, and low-level individual robot control for collision-free navigation in complex environments.

The remainder of the paper is organized as follows. Section II provides a literature review of the state-of-the-art related work. In Section III, we introduce our novel STAF approach to enable subteaming and formation adaptation. Section IV discusses our experimental results using both robotics simulations and physical robot teams. In Section V, we discuss the limitations of our work and point out future directions. Finally, we conclude the paper in Section VI.

II. Related Work

In this section, we will first review existing techniques for learning-free coordinated multi-robot navigation. Then, we will review previous methods for subteaming in the areas of multi-robot navigation and task allocation. Finally, we present the state-of-the-art works on hierarchical learning for robotics.

A. Learning-Free Coordinated Multi-Robot Navigation

We review existing methods from two main perspectives: the multi-robot team formation and the theoretical perspective to enhance coordinated efficiency. From the multi-robot team formation angle, prevalent configurations include the leader-follower structure, where follower agents are programmed to maintain group behavior by following a leader agent [4, 15, 24, 25]. Additionally, virtual region methods that allow robot teams to adjust their formation within specified virtual areas are thoroughly explored [26–29]. However, these formations are often rigid and lack the flexibility needed to adapt to complex environments that require dynamic formation changes.

From the theoretical perspective, classic methods of coordinated multi-robot navigation are categorized into three groups: traditional planning methods, game-theoretical approaches, and optimization-based techniques. Traditional planning methods include algorithms, such as A^* and its variants [30], rapidly exploring random trees (RRT) [12], and probabilistic roadmap (PRM) [31]. Game-theoretical approaches model multi-robot navigation as cooperative games for path planning [13, 14]. Optimization-based methods aim to optimize various objectives in order to coordinate multiple robots during navigation, such as identifying traversable areas to prevent collisions [32, 33], maintaining communication [15], maximizing area coverage [34], and addressing hierarchical quadratic programming (HQP) problems for cooperative tasks [11, 35]. Traditional methods in coordinated navigation are primarily based upon heuristic searching and typically incur substantial computational costs. Additionally, none of these previous classic methods effectively address subteaming and formation adaptation in the context of coordinated navigation, particularly in complex scenarios such as traversing narrow corridors.

B. Subteaming in Multi-Robot Navigation and Task Allocation

Integrating subteaming with coordinated multi-robot navigation introduces additional complexity beyond the standard multi-robot coordination, which requires team splitting, merging, and reformation in response to environmental and task constraints. Existing methods can be broadly categorized into four groups, including graph-based, leader-follower-based, optimization-based, and heuristic-based methods.

Graph-based methods [1, 20, 21, 36] use graph partitioning and graph cut techniques to determine how to divide and merge teams, typically relying on centralized computation or explicit connectivity constraints. Leader-follower methods [15, 37, 38] employ predefined hierarchy-based motion strategies, where a subset of agents leads and others follow, limiting flexibility in dynamic environments. Optimization-based methods typically use mixed-integer programming [22, 23, 39, 40] to compute optimal assignments and motion plans. Heuristic-based methods [41, 42] offer computationally efficient alternatives by leveraging problem-specific heuristics to determine team formation and coordination strategies. However, these methods generally focus on team division alone, without the capability of controlling the subteams or individual robots, which makes them unsuitable for addressing coordinated navigation.

C. Hierarchical Learning for Robotics

Recently, learning-based methods have gained significant attention for improving coordinated navigation in multi-robot systems. Reinforcement learning (RL) approaches have shown promising results in enabling robots to adapt to environmental changes [43, 44]. However, single-level RL methods often struggle with convergence in complex scenarios. Graph neural networks (GNNs) have been used to enhance team coordination and communication [17, 45], supporting decentralized decision-making [16, 46]. Multi-agent reinforcement learning (MARL) further improves coordinated navigation by training robots to cooperate effectively [18, 19, 44, 47]. These learning-based approaches have been successfully applied in areas such as connected autonomous driving [8, 48], area coverage [49], and search-and-rescue missions [5].

Hierarchical learning attracts increasing attention to address this issue for complex multi-robot tasks, such as solving combinatorial optimization for multi-robot task allocation [50], maintaining communication that ensures connectivity among robots [51], multi-robot path planning [52, 53] and consensus reaching [54]. Specifically, the lower level policy aims to optimize individual robot control, such as enabling obstacle avoidance [55, 56]. The upper level focuses on multi-robot planning and coordination, such as selecting sub-goals through goal-based planning [57], dividing exploration areas using dynamic Voronoi partitions [58], facilitating obstacle avoidance [59] and communication between robots [18].

These methods leverage hierarchical policy to optimize both high-level task planning and low-level motion control simultaneously, which achieves promising performance compared to traditional methods that rely on predefined rules and explicit environment representations. However, applying hierarchical RL to formation adaptation and subteaming remains an open challenge due to the need for scalable representations of team structures, dynamic adaptation to changing environments, and efficient integration of formation control with flexible team reconfiguration.

III. Approach

In this section, we discuss our STAF method that enables new multi-robot capabilities of subteaming and formation adaptation for coordinated multi-robot navigation. An overview of STAF is illustrated in Figure 2.

Notation. Matrices are represented as boldface capital letters, e.g., $\mathbf{M} = (m_{i,j}) \in \mathbb{R}^{n\times m}$, where $m_{i,j}$ denotes the element in the i-th row and j-th column of $\mathbf{M}$. Vectors are represented as boldface lowercase letters $\mathbf{v} = \{v_1, v_2, \ldots, v_n\} \in \mathbb{R}^n$, where v_i is the i-th element in the vector. Scalars are represented by non-bold lowercase letters. Additionally, we represent a set of n variables using calligraphic symbols, e.g., $\mathcal{V} = \{\mathbf{v}_i\}^n$.

A. Problem Definition

We represent a team of n robots using an undirected graph $\mathcal{G} = \{\mathcal{V}, \mathbf{E}\}$. In the node set $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$, each node $\mathbf{v}_i = \{\mathbf{p}_i, \mathbf{g}_i, \mathbf{q}_i\}$ consists of the attributes of the i-th robot, where $\mathbf{p}_i = [p_i^x, p_i^y]$ denotes its position, $\mathbf{g}_i = [g_i^x, g_i^y]$ denotes its goal position, and $\mathbf{q}_i = [q_i^x, q_i^y]$ denotes its velocities along x and y directions. The edge matrix $\mathbf{E} = \{a_{i,j}\}^{n\times n}$ represents the spatial adjacency of the robots, where $a_{i,j} = 1$, if the i-th robot and the j-th robot are within a radius; otherwise $a_{i,j} = 0$. We further define the state of the i-th robot $\mathbf{s}_i = [\mathbf{p}_i, \mathbf{g}_i, \mathbf{q}_i, c_i]$ as the concatenation of the robot's attributes and the distance c_i between the robot and its closest obstacle. We define the action of the i-th robot as $\mathbf{a}_i = [v_i^x, v_i^y]$, where v_i^x and v_i^y denote the robot's velocities in the x and y directions, respectively.

Our objective is to address the problems of subteaming and formation adaptation in the context of coordinated multi-robot navigation:

- **Formation Adaptation:** The capability of a robot team or subteam to maintain a desired formation while dynamically adjusting their relative positions to safely and efficiently navigate through the unstructured environment toward their goal positions, particularly in challenging scenarios such as narrow corridors.
- **Subteaming**: The capability of a robot team with a specific formation to autonomously divide into subteams with the same formation type when navigating environments too narrow for the entire robot team. After successfully passing through, the subteams must merge back into the full team, restoring the original formation.

B. High-Level Deep Graph Cut for Subteaming

Given the graph $\mathcal{G}$ as the representation of the robot team, we introduce a new deep graph cut approach at the high level of STAF to enable subteaming. We compute the embedding of the robot graph as $\mathcal{H} = \{\mathbf{h}_i\} = \omega(\mathcal{G})$, where $\mathbf{h}_i$ is the embedding of the i-th robot and ω is a graph attention network [60]. We project each node into a representation space by calculating $\mathbf{m}_i = \mathbf{W}^v \mathbf{p}_i$, where $\mathbf{m}_i$ denotes the projected feature vector of the i-th node, and $\mathbf{W}^v$ denotes the weight matrix. Then, we compute the attention $\alpha_{i,j}$ from the j-th node to the i-th node as follows:

$$\alpha_{i,j} = \frac{\exp\left(\mathrm{ReLu}([\mathbf{W}^a \mathbf{m}_i || \mathbf{W}^a \mathbf{m}_j])\right)}{\sum_{k\in\mathcal{N}(i)} \exp(\mathrm{ReLu}([\mathbf{W}^a \mathbf{m}_i || \mathbf{W}^a \mathbf{m}_k]))} \tag{1}$$

where ReLu denotes the rectified linear unit activation function, $\mathcal{N}(i)$ represents the set of adjacent nodes of the i-th node, $||$ denotes the concatenation operation, and $\mathbf{W}^a$ represents the weight matrix. The attention $\alpha_{i,j}$ is obtained by computing the similarity of the i-th node with its j-th adjacent nodes, followed by the SoftMax normalization. Then, the final embedding $\mathbf{h}_i$ for the i-th node is computed through aggregating the embeddings of all its adjacent nodes as follows:

$$\mathbf{h}_i = \mathbf{W}^h \mathbf{m}_i + \sum_{j\in\mathcal{N}(i)} \alpha_{i,j} \mathbf{W}^h \mathbf{m}_j \tag{2}$$

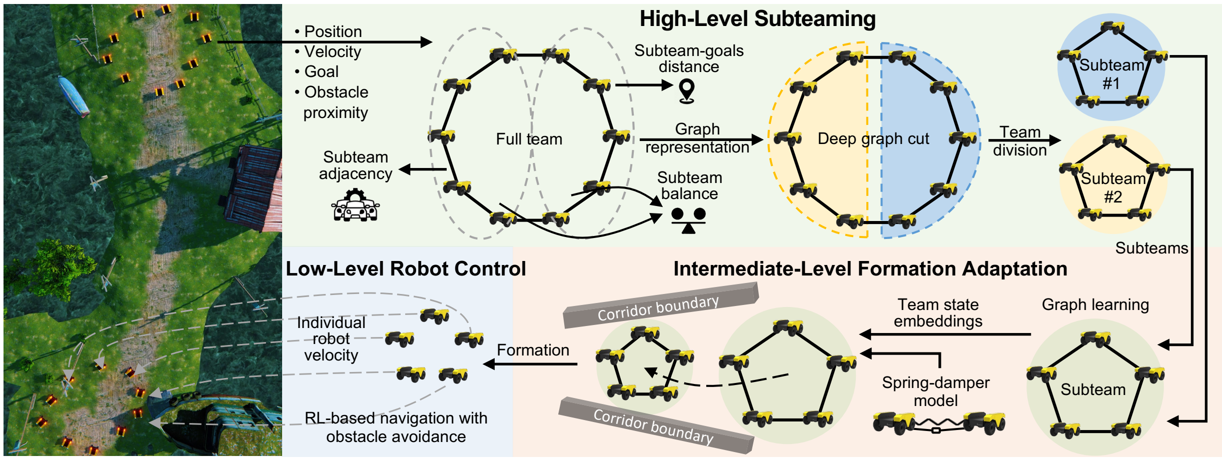


Fig. 2. Overview of STAF, which integrates three levels of robot learning within a unified hierarchical learning framework to enable coordinated multi-robot navigation. It comprises: (1) a high-level deep graph cut for team division, (2) intermediate-level graph learning for formation adaptation, and (3) low-level RL-based policy learning for individual robot control in navigation and obstacle avoidance.

where $\mathbf{W}^h$ is the weight matrix. We further utilize a multi-head mechanism [60] after the attention layers to enable the network to capture a richer embedding representation.

Given $\mathcal{H} = \{\mathbf{h}_i\}$, we formulate subteaming as a graph cut problem, which partitions the entire graph (representing the full team) into m subgraphs (representing subteams). In order to compute team division, we develop a classifier network $\tau(\mathcal{H})$ consisting of two fully connected linear layers followed by a SoftMax function, which outputs the team division results as $\mathbf{Y} = \tau(\mathcal{H}) = \{y_{i,j}\}^{n\times m}$, where $y_{i,j}$ is the probability of the i-th robot belonging to the j-th subteam, and $m < n$.

To ensure that robots within the same subteam group together, i.e., each robot is adjacent to its teammates within the same subteam, we define a loss function that maximizes the adjacency of robots within each subteam as $\mathbf{Y}(1-\mathbf{Y})^\top\mathbf{E}$, where $\mathbf{Y}(1-\mathbf{Y})^\top$ calculates the probability that a pair of robots belong to different subteams, and $\mathbf{E}$ encodes the adjacency of the robots. In addition, we aim to maintain balance in the sizes of robot subteams, encouraging each subteam to have the same or a similar number of robots. It can be mathematically modeled by a loss function $\sum_{j=1}^m \left(\sum_{i=1}^n y_{i,j} - \frac{n}{m}\right)$. The term $\frac{n}{m}$ calculates the optimal size of balanced subteams (e.g., when $n = 10$ and $m = 2$, each subteam would consist of 5 robots). Furthermore, we model the mission objective of reaching the goal position by minimizing the overall distance between the subteams and their respective goal positions. It can be mathematically defined as $\sum_{j=1}^m \left\| \frac{\sum_{i=1}^n y_{i,j}\mathbf{p}_i}{\sum_{i=1}^n y_{i,j}} - \frac{\sum_{i=1}^n y_{i,j}\mathbf{g}_i}{\sum_{i=1}^n y_{i,j}} \right\|_2$, where $\frac{\sum_{i=1}^n y_{i,j}\mathbf{p}_i}{\sum_{i=1}^n y_{i,j}}$ denotes the center position of the j-th subteam and $\frac{\sum_{i=1}^n y_{i,j}\mathbf{g}_i}{\sum_{i=1}^n y_{i,j}}$ denotes the center position of the goal for the subteam.

The high-level component of STAF performs an unsupervised graph cut to enable team division for subteaming by minimizing the following objective function:

$$\mathcal{L}_{st} = \overbrace{\mathbf{Y}(1-\mathbf{Y})^\top\mathbf{E}}^{\text{Subteam adjacency}} + \overbrace{\sum_{j=1}^m \left(\sum_{i=1}^n y_{i,j} - \frac{n}{m}\right)}^{\text{Subteam balance}} + \overbrace{\sum_{j=1}^m \left\| \frac{\sum_{i=1}^n y_{i,j}\mathbf{p}_i}{\sum_{i=1}^n y_{i,j}} - \frac{\sum_{i=1}^n y_{i,j}\mathbf{g}_i}{\sum_{i=1}^n y_{i,j}} \right\|_2}^{\text{Subteam-goals distance}} \tag{3}$$

which jointly accounts for subteam adjacency, subteam balance, and subteam-goal distances.

C. Intermediate-Level Graph Learning for Multi-robot Formation Adaptation

To enable adaptive multi-robot formation control, we develop a graph learning approach at the intermediate level of STAF, which coordinates multiple robots to maintain a specific formation while adapting it based on the surrounding environment. Given $\mathcal{G}$ that represents a team (or subteam) of robots along with the state $\mathbf{s}_i$ for each robot i, we develop a graph network ϕ to compute the embedding $\mathbf{f}_i = \phi(\mathbf{s}_i, \mathcal{G})$ of the team state with respect to the i-th robot, which encodes the spatial relationships between the i-th robot with others in the team. The network ϕ uses a linear layer to project the robot state $\mathbf{s}_i$ to the individual embedding $\mathbf{z}_i$ of the i-th robot by $\mathbf{z}_i = \mathbf{W}^z\mathbf{s}_i$, where $\mathbf{W}^z$ is the weight matrix of the linear layer. Then, for the i-th robot, ϕ aggregates individual state embeddings of all other teammates through message passing to compute the team state embedding $\mathbf{f}_i$ with respect to the i-th robot as follows:

$$\mathbf{f}_i = \mathbf{W}^f\mathbf{z}_i + \sum_{j\in\mathcal{N}(i)} \mathbf{W}^f(\mathbf{z}_j - \mathbf{z}_i) \tag{4}$$

where $\mathbf{W}^f$ is the weight matrix. The team state embedding $\mathbf{f}_i$ with respect to the i-th robot encodes not only its own states (captured in the first term), but also the relative spatial relationships with other teammates (captured in the second term), which facilitates the coordination of actions to maintain specific formations during multi-robot navigation.

Robot teams and subteams may encounter scenarios, such as narrow corridors, where rigidly maintaining their formations prevents successful navigation. To enable formation adaptation, we implement a spring-damper model [61, 62] that dynamically adjusts the shape of the formation within the same type. This spring-damper model includes two components: (1) The spring component ensures that robot pairs maintain a balance between staying close enough to navigate narrow corridors and keeping a sufficient distance to prevent collisions, with the flexibility to adjust formation and enable adaptation. This spring component can be modeled as $|d_{i,j} - p_{i,j}|$, where $d_{i,j}$ denotes the expected distance in the original rigid formation and $p_{i,j}$ represents the actual distance between the i-th and j-th robots, computed as $\|\mathbf{p}_i - \mathbf{p}_j\|_2$. (2) The damper component prevents oscillation and overshooting of each robot in the team during navigation by smoothing the relative velocities between pairs of robots, which is defined as $q_{i,j} = \|\mathbf{q}_i - \mathbf{q}_j\|_2$. Combining these two components, the spring-damper model for formation adaptation can be mathematically defined as:

$$R^{adp} = \sum_{\mathbf{v}_i, \mathbf{v}_j \in \mathcal{V}} -\lambda |d_{i,j} - p_{i,j}| - (1-\lambda) q_{i,j} \tag{5}$$

where λ is a hyperparameter that balances the importance of the spring and damper components. R^{adp} is incorporated into the reward function, which is later used to derive a loss function for jointly training STAF.

D. Low-Level Individual Robot Control for Navigation

At the low-level of STAF, we introduce a navigation control network that outputs velocity commands as actions for each individual robot to reach its designated goal position.

Given the state $\mathbf{s}_i$ for the i-th robot, we compute its state embedding $\mathbf{f}_i$. We design the network ψ, which consists of two linear layers followed by the ReLU activation function, maps this embedding to an action as $\mathbf{a}_i = \psi(\mathbf{f}_i)$. The network ψ is a part of the control policy $\pi_\theta(\mathbf{a}_i|\mathbf{s}_i)$, parameterized by θ, which is trained using the framework of reinforcement learning. To enable each robot to move toward its target position and reach the navigational goal, we design a reward function based upon the distance between the current positions of the robot and its goal position. To enable obstacle avoidance for safe navigation, we implement a reward function that imposes a penalty when a robot comes too close to nearby obstacles or other robots in the team. When robots are divided into subteams, and once all subteams pass through the narrow corridor into an open area that is large enough for the full team, the goal position of each individual robot is updated to align with the full team's goal, thereby recovering the subteams back into the full team with the original formation.

E. STAF Training and Execution

To train STAF as a three-level hierarchical learning model, we design an alternating training algorithm that iterates between using unsupervised learning to train the high level for subteaming and using Proximal Policy Optimization (PPO) [63] to jointly train the intermediate level for formation adaptation and the low level for individual robot control.

Specifically, the high-level training receives a 2D occupancy map of the environment (e.g., built using a SLAM approach [64]), as well as the starting and goal positions of the robots within the map as input. The high level is trained using ADMM as the optimization solver [65] by deriving the gradient of the unsupervised loss function in Eq. (3) to update the weights $\mathbf{W}^a$ and $\mathbf{W}^h$ of the deep graph cut network τ for subteam division, which considers subteam adjacency, subteam balance, and subteam-goal distance. In the same iteration, we fix the high-level model once its training is complete, and we utilize PPO to jointly train the intermediate and low levels of STAF. We design the overall reward as a weighted summation of the coordination reward in Eq. (5) for formation adaptation, and the navigation reward and obstacle avoidance reward for individual robot control. These rewards are used to compute the advantage function $A^{\pi_{old}}(\mathbf{s}_i, \mathbf{a}_i)$, which quantifies how much better taking action $\mathbf{a}_i$ in state $\mathbf{s}_i$ is compared to the old policy $\pi_{\theta_{old}}(\mathbf{s}_i, \mathbf{a}_i)$. Then, a loss value is computed by aggregating the differences for all robots between the output of the updated PPO policy $\pi_\theta(\mathbf{s}_i, \mathbf{a}_i)$ with and the old policy $\pi_{\theta_{old}}(\mathbf{s}_i, \mathbf{a}_i)$. To prevent instability in training due to large policy updates, a clipping function $\text{clip}(1-\delta, 1+\delta)$ is used to constrain the ratio between the updated and old policies, ensuring that training stays within a stable trust region defined by δ. Integrating the components above, the loss function can be expressed as:

$$\begin{aligned}\sum_{\mathbf{v}_i \in \mathcal{V}} \mathbb{E}_{\mathbf{s}_i, \mathbf{a}_i \sim d^{\pi_{\theta_{old}}}} \Bigg[\min \Bigg(&\frac{\pi_\theta(\mathbf{a}_i|\mathbf{s}_i)}{\pi_{\theta_{old}}(\mathbf{a}_i|\mathbf{s}_i)} A^{\pi_{old}}(\mathbf{s}_i, \mathbf{a}_i), \\ &\text{clip}\left(\frac{\pi_\theta(\mathbf{a}_i|\mathbf{s}_i)}{\pi_{\theta_{old}}(\mathbf{a}_i|\mathbf{s}_i)}, 1-\lambda, 1+\lambda \right) A^{\pi_{old}}(\mathbf{s}_i, \mathbf{a}_i) \Bigg) \Bigg]\end{aligned} \tag{6}$$

where $d^{\pi_{\theta_{old}}}$ represents the probability of encountering a state $\mathbf{s}_i$ and performing an action $\mathbf{a}_i$ while following the old policy θ_{old}, and $\mathbb{E}$ is the expectation over $d^{\pi_{\theta_{old}}}$ for all robots. Gradients computed from this objective are used to train the individual robot navigation policy π_θ at the low level, and backpropagated to the intermediate level to update weights $\mathbf{W}^f$ of the graph network ϕ for formation adaptation.

During execution, STAF performs centralized planning with decentralized execution. STAF assumes the 2D occupancy map of the environment along with the starting and goal positions of the robots as input, just as in training. The deep graph cut at the high level of STAF is performed in a centralized manner (although decentralization is possible, as discussed in Section V): Each robot broadcasts its state via wireless communication (e.g., Wi-Fi), and a designated robot collects the states from all its teammates to compute the subteam assignments. After subteam division, formation adaptation at STAF's intermediate level is performed in a decentralized manner. Through the same broadcasting mechanism via Wi-Fi, each robot can determine the relative positions of its teammates, which enables the robot to dynamically adjust its own position in relation to others to maintain and adapt the designated formation. Then, each robot executes velocity commands derived from the individual robot control policy at the low level of STAF.

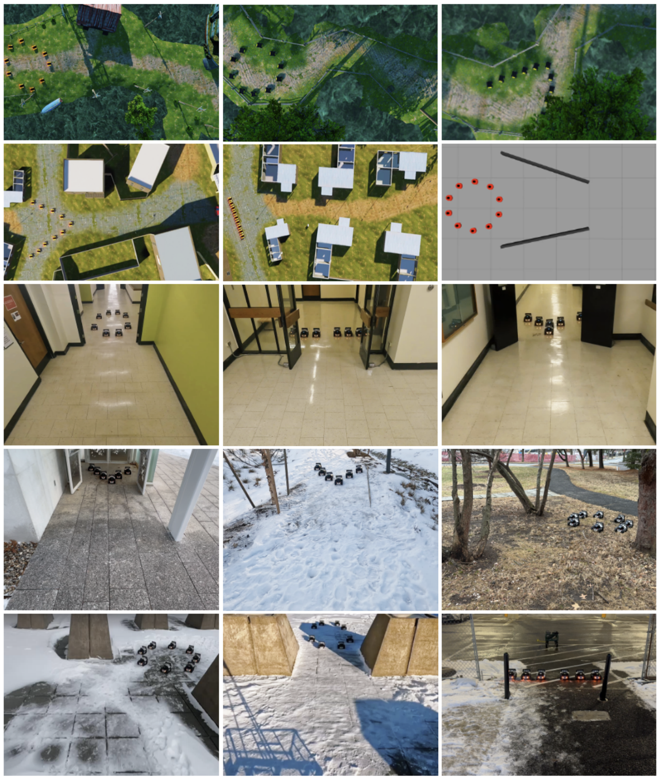


Fig. 3. Example scenarios used to comprehensively evaluate and validate out STAF approach in multi-robot simulations as well as using real physical robots in both indoor and outdoor environments.

IV. Experiments

In this section, we present our experimental setups, analyze the experimental results from multi-robot simulations and using real physical robots in both indoor and outdoor environments, and discuss characteristics of our approach. A video demo is submitted along with this manuscript to showcase the qualitative results of our STAF approach, demonstrating its execution using both simulated and real robot teams.

A. Experimental Setups

We comprehensively evaluate our STAF approach across three experimental setups: (1) a standard Gazebo simulation in ROS1, (2) a high-fidelity Unity-based 3D multi-robot simulator in ROS1, and (3) physical robot teams running ROS2. Each setup involves varying numbers of robots arranged in specific formations, including circle, wedge, and line. In all scenarios, the environment includes narrow corridors, such as bridges, doorways, and road entries, which prevent the entire team from passing through while maintaining its full formation. Several representative experimental scenarios are illustrated in Figure 3. To navigate and reach the other side of the corridor, the full robot team must divide into smaller subteams. Each subteam adaptively adjusts its formation to traverse the corridor. After passing through, the subteams regroup to restore the full team and return to the original formation. In simulation experiments, robot poses and environmental obstacles are obtained from Gazebo and high-fidelity Unity simulations. In experiments with real physical robots, each robot employs a SLAM approach [64] for state estimation and environmental mapping.

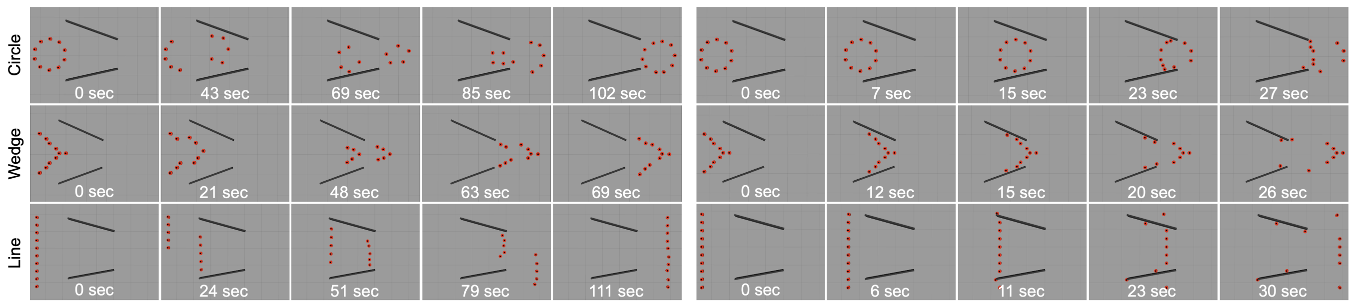
To implement STAF, the edges in the robot team graph are constructed by connecting the nearby robots within a radius setting to 2 meters. STAF's high-level deep graph cut network contains one linear layer with $\mathbf{W}^v$ setting to the dimension of 2×32 and three transformer layers with the parameter $\mathbf{W}^a$ and $\mathbf{W}^h$ setting to the dimension of 32×32. The intermediate-level GNN for formation adaptation contains one encoder with $\mathbf{W}^z$ setting to the dimension of 6×64 and one GNN layer with $\mathbf{W}^g$ setting to the dimension of 64×64. The hyper-parameter $\lambda = 0.6$ in Eq. (5) to balance spring and damper force. We generate synthetic data to train our STAF approach. Specifically, given a robot team formation, we randomly generate the positions of the robots within the formation. In total, we collect $10,000$ data instances to train our high-level network. The high-level network is trained for 100 epochs, while the adaptive formation control policy, involving both the intermediate-level and low-level neural networks, is trained over a total of 800 epochs. This alternating training of the high-level and joint intermediate-low-level networks continues until convergence.

We implement the complete STAF approach, incorporating all the three levels within the hierarchical learning framework, referred to as **STAF-full**. In this approach, the full team divides into subteams to navigate through narrow environments, and after passing through, the subteams regroup to restore the full team to its original formation. To analyze the performance of the subteams, we refer to the subteams as **STAF-sub#** (e.g., STAF-sub1 and STAF-sub2): The full team divides into a total of # subteams, and we evaluate the performance of the #-th subteam as it navigates to its goal position while maintaining its formation. For comparison with STAF, we further implement two previous methods for multi-robot coordinated navigation, including:

- Leader and Follower (**L&F**) [15]: one of the robots is designated as the "leader robot" that leads the movements of the other "follower robots" in the team while maintaining the formation.
- Decentralized GNN (**DGNN**) [18] a hierarchical learning framework to directly generate velocity controls for each individual robot for navigation, without considering team-level formations.

To quantitatively evaluate and compare with other methods, we employ three metrics for coordinated multi-robot navigation, including:

- Successful Rate (**SR**): the proportion of the robots within the full team that successfully reach their goal positions without any collisions.
- Travel Time (**TT**): total time used by the entire robot team to reach to the goal position in successful cases.
- Contextual Formation Integrity (**CFI**) measures the real-

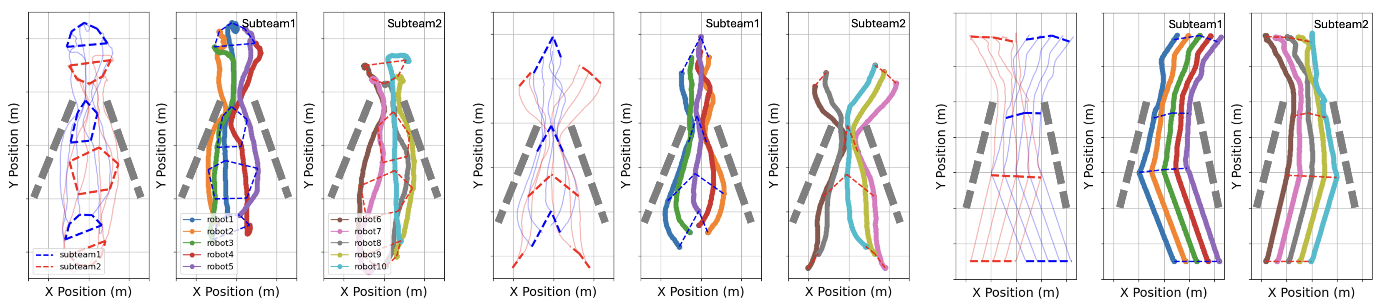


(a) Our STAF (b) L & F

Fig. 4. Qualitative results from Gazebo simulations in ROS1 on subteaming and formation adaptation using a team of ten robots that maintain circle, wedge, and line formations.

TABLE I
QUANTITATIVE RESULTS OF STAF COMPARED WITH PREVIOUS METHODS FROM GAZEBO SIMULATIONS IN ROS1. THE CIRCLE, WEDGE, AND LINE FORMATIONS ARE EVALUATED BASED ON THE METRICS OF SR, TT AND CFI.

Method	Circle Formation					Wedge Formation					Line Formation				
	SR (%)	TT (sec)	$\sigma < 0.5$	$\sigma < 0.1$	$\sigma < 0.01$	SR (%)	TT (sec)	$\sigma < 0.5$	$\sigma < 0.1$	$\sigma < 0.01$	SR (%)	TT (sec)	$\sigma < 0.5$	$\sigma < 0.1$	$\sigma < 0.01$
DGNN [18]	**100.00**	68.70	60.41	58.91	58.91	**100.00**	82.70	47.85	42.33	41.92	**100.00**	72.61	27.90	20.16	20.16
L&F [15]	40.00	**27.40**	67.28	64.54	62.69	70.00	**26.50**	69.70	62.11	59.47	60.00	**30.10**	63.76	55.89	55.89
STAF-full	**100.00**	102.10	**87.79**	**80.12**	**80.12**	**100.00**	69.30	**80.52**	**80.51**	**80.50**	**100.00**	111.50	**91.45**	**80.06**	**78.93**



(a) Circle formation (b) Wedge formation (c) Line formation

Fig. 5. Movement trajectories of a team of ten robots navigating through a narrow corridor with different formations. In Figures 5(a) to 5(c), the first subfigure illustrates the subteaming and formation adaptation behaviors, which displays the two subteams (indicated by red and blue) as they undergo team division, coordinated navigation with formation adaptation, and subsequent regrouping. The second and third subfigures show the trajectories of individual subteams at specific timesteps. Each robot's path is represented by a distinct color, while gray dashed lines denote obstacles. [Best viewed in color.]

time adherence of the robots to their designated formation, given a shape threshold that defines the strictness of the formation.

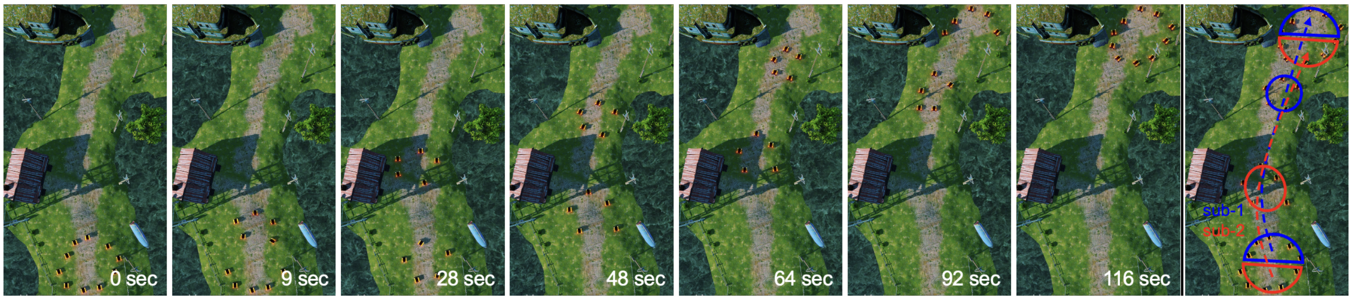
The CFI metric combines concepts of thresholds and uncertainty, which are commonly applied in computer vision [66]. It is formally defined and calculated as follows:

$$w\left(1-\sigma^{-1}\min\left(\left|r-(\eta+\sigma)\right|,\left|r-(\eta-\sigma)\right|\right)\right)+(1-w)\epsilon$$

The CFI metric evaluates the performance of adaptive formation control based on two factors: effective use of the corridor gap and the integrity of the formation. The first term assesses the team's efficiency in utilizing the corridor gap, where r is the robot team's maximum radius, η denotes the corridor width with a safety margin, and σ is a threshold with smaller values imposing stricter formation requirements. CFI's second term $\epsilon \in [0, 1]$ evaluates the integrity of the team shape. For example, for a team of 5 robots in a circle formation, $\epsilon = 1-\frac{1}{5}\sum_{i=1}^{5}\frac{\theta_i}{108}$, where θ_i represents the interior angle of the triangle with the i-th robot as the vertex, and 108 is the interior angle of the pentagon, approximating a circle when the team has 5 robots. CFI combines these two terms to evaluate how effectively a robot team uses the corridor space and maintains its formation, with the balance determined by the coefficient w. The metric CFI $\in [0, 1]$, where higher values indicate better performance. In our experiments, we set $w = 0.5$ to treat the gap usage and the formation integrity equally important. Additionally, we set σ to twice the width of the robot used in the corresponding experiments.

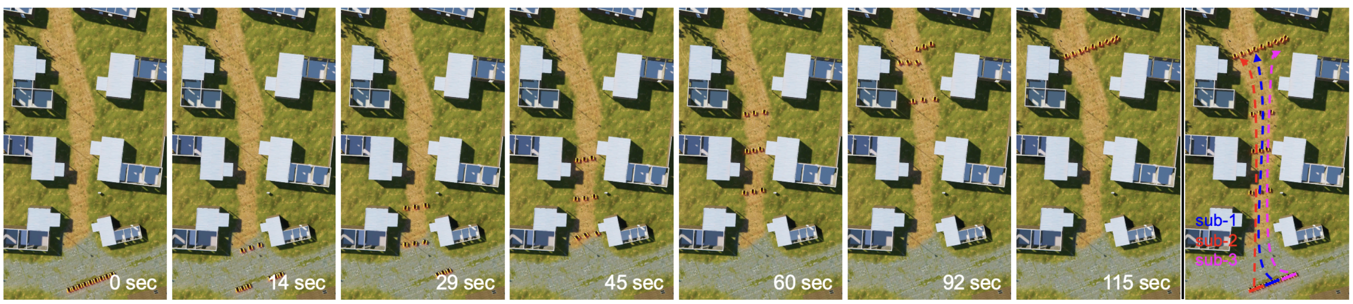
B. Results on Subteaming and Formation Adaptation in Multi-Robot Simulations

In the first set of experiments, we evaluate our STAF method using Gazebo simulations in ROS1, where a team of robots navigates through a narrow corridor toward a goal position on the other side, while maintaining designated formations such as circle, wedge, and line formations.



(a) A team of ten Warthog robots with a circle formation navigates through a narrow corridor on uneven terrain.

(b) A team of ten Warthog robots in a wedge formation traverses a progressively narrowing corridor between buildings.



(c) A team of nine robots in a line formation navigates through multiple narrow passages formed by closely spaced buildings.

Fig. 6. Qualitative results on subteaming and formation adaptation during coordinated multi-robot navigation using a high-fidelity Unity3D simulations in ROS1. The experiments adopt different numbers of differential-drive Warthog robots that maintain circle, wedge and line formations while traversing an unstructured outdoor field environment

The qualitative results in the Gazebo simulation are shown in Figure 4. We observe that L&F gets stuck in the narrow corridor due to the lack of subteaming and formation adaptation. In contrast, our method successfully navigates to the goal with the adaptation to both circle and wedge formations. Our high-level deep graph cut network autonomously splits the full team into subteams. To notice, the wedge team splitting considers more on the goal-distance objective instead of maximizing connectivity, which leads to more compact subteam generation. The first subteam, being closer to the goal, starts moving while maintaining a circle formation. Once the first subteam reaches the goal, it recovers into the full-team formation and waits for the second subteam. Similarly, the second subteam navigates through the narrow corridor with circle formation adaptation capability, then merges with the first subteam, recovering into the full robot team formation.

We further visualize the trajectories of a team of 10 robots navigating in different formations, as shown in Figure 5. The visualization clearly reveals subteaming behaviors within each formation (indicated by subteams in red and blue colors), including team division and regrouping. Additionally, within each subteam, formation adaptation occurs when navigating through narrow corridors (indicated by the individual robot trajectories). These results demonstrate the effectiveness of our proposed approach in enabling both subteaming and formation adaptation capabilities.

The quantitative results are shown in Table I. We observe that DGNN performs the worst, particularly in the CFI metrics. This poor performance is attributed to DGNN's sole focus on reaching the goal, neglecting formation control. L&F performs better than DGNN by incorporating formation control, but its success rate is only 40%. Since it does not consider subteaming and formation adaptation during navigation, it is challenging for the full robot team to navigate through narrow corridors. By addressing both subteaming and formation adaptation control, our method outperforms all others. Specifically, our subteams

(a) A team of eight physical robots with a circular formation navigates through a narrow doorway in a hallway.

(b) A team of six robots with a wedge formation navigates through a narrow exit from indoors to a partially open outdoor area.

(c) A team of six robots in a line formation navigates through a slightly wider but still confined doorway.

Fig. 7. Qualitative results of coordinated multi-robot navigation, including team division, formation adaptation, and team recovery, using varying numbers of differential-drive Limo robots that maintain circle, wedge, and line formations across different indoor scenarios.

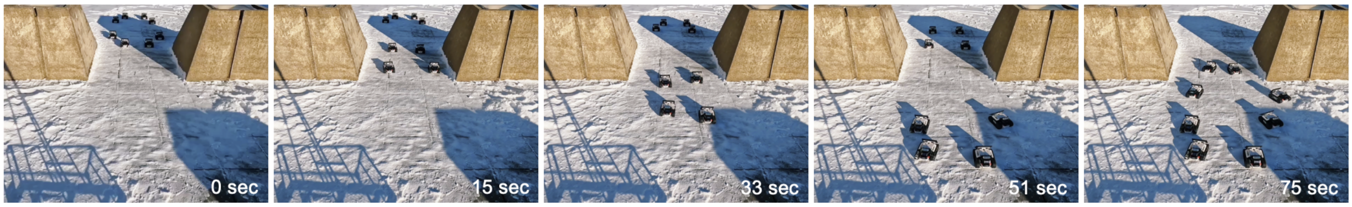
TABLE II
QUANTITATIVE RESULTS OF SUBTEAMS FROM GAZEBO SIMULATIONS. THE CIRCLE, WEDGE AND LINE FORMATIONS ARE EVALUATED BASED ON THE METRICS OF SR, TT AND CFI.

Subteam	Formation	Metrics				
		SR (%)	TT (sec)	$\sigma < 0.5$	$\sigma < 0.1$	$\sigma < 0.01$
STAF-sub1	Circle	100.00	84.80	81.56	71.69	70.59
	Wedge	100.00	58.80	77.22	71.79	69.86
	Line	100.00	78.51	91.13	79.53	77.43
STAF-sub2	Circle	100.00	59.50	87.72	82.67	80.11
	Wedge	100.00	46.80	80.99	80.15	75.79
	Line	100.00	90.06	91.78	81.74	80.43

achieve a 100% success rate due to their formation adaptation capability. Although existing methods have shorter travel times than ours, this is because our STAF method involves a complex navigation strategy (e.g., subteaming) and adaptive formation control, which increases the travel time. The results of subteam performance over adaptive formation control are presented in Table II. We can observe that each subteam can successfully reach their goals with 100% successful rate across all three formations. Furthermore, STAF maintains over %87, %80 and %91 formation integrity under the threshold $\sigma < 0.5$ for circle, wedge, and line formations respectively. These results highlight the effectiveness of our STAF approach in maintaining formation integrity and underline the importance of subteaming and adaptive formation control for coordinated navigation. in complex environments.

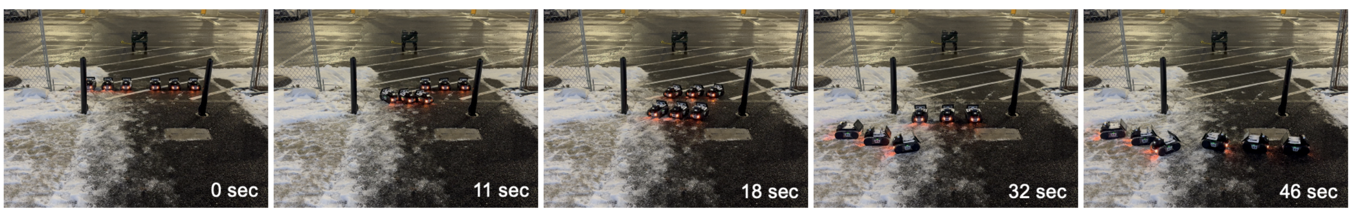
Beyond the Gazebo simulation, we further use a high-fidelity simulator that integrates both the Unity3D engine for enhanced visual fidelity with ROS1 for multi-robot perception and control, which simulates outdoor field environments that include narrow pathways and bridges over creeks. These experimental settings are challenging. First, instead of holonomic robots, we conduct experiments with varying numbers of differential-drive Warthog robots. To adapt our approach to differential-drive robots, we convert the linear velocity in the action $\mathbf{a}_i$ into wheel velocities to follow the same trajectory. Second, this scenario involves navigating along complex and curved trajectories, requiring continuous formation adjustments throughout navigation. Third, certain areas along the navigation path are very narrow, which requires the full team to divide into more than two subteams for effective traversal.

As illustrated in Figures 6(a) and 6(b), we can observe that our STAF approach can successfully divide a full team into subteams and smoothly adjust the actions of differential-drive subteam to navigate along complex and curved trajectories until reaching the goal. This validates the effectiveness of our method in handling formation adaptation and subteaming in challenging environments. In addition, Figure 6(c) presents that our approach is able to dynamically divide into three subteams, allowing all subteams to successfully navigate through the narrow corridor and regroup together afterward. This further validates the capability of STAF in handling subteaming beyond two subteams in extremely narrow scenarios.



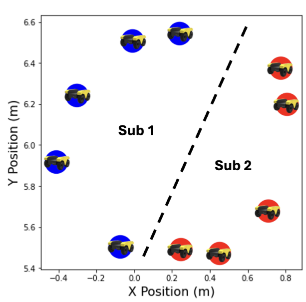
(a) A team of eight physical Limo robots with a circle formation traverses through a narrow passage between two concrete security bollards.

(b) A team of six robots in a wedge formation navigates through a forest-like environment with narrow corridors surrounded by scattered trees and obstacles.

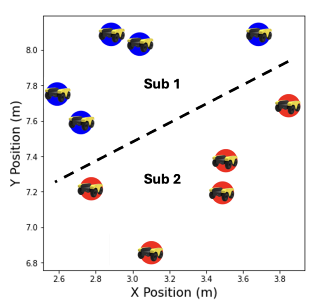


(c) A team of six Limo robots in a line formation navigates through a narrow pathway with boundaries marked by two sticks blocking vehicle access.

Fig. 8. Qualitative results of team division, formation adaptation, and team recovery during coordinated navigation using varying numbers of differential-drive Limo robots that maintain circle, wedge, and line formations across different unstructured outdoor environments.



(a) Normal situation



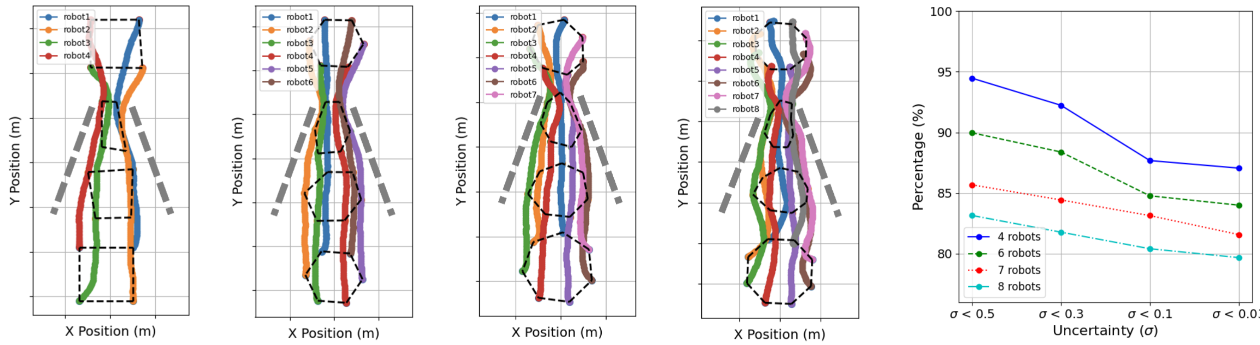
(b) Noisy situation

Fig. 9. STAF's robustness in graph-cut-based subteam division to noise.

C. Case Study Validation of STAF for Subteaming and Formation Adaptation on Physical Robot Teams

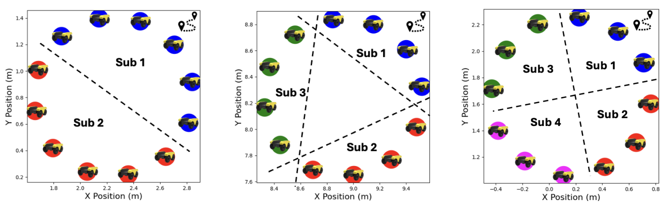
We validate our STAF method through case studies involving real physical multi-robot teams, using differential-drive Limo robots equipped with caterpillar tracks. Each Limo robot is equipped with an Intel NCU i7 processor, which enables it to execute our approach onboard. Each robot runs on ROS2 and communicates with other robots in the team via Wi-Fi-based broadcasting. The experiments are conducted in both real-world indoor and outdoor environments, and we present six representative scenarios in the paper, each highlighting various real-world challenges. The indoor environments involve navigating constrained spaces, such as a narrow doorway in a hallway, a tight exit from an indoor area to a partially open outdoor space, and a slightly wider but still confined corridor. The outdoor experiments are conducted on unstructured terrain, including a narrow passage between two concrete security bollards, a forest-like environment with narrow corridors surrounded by scattered trees and obstacles, and a pathway with boundaries marked by two sticks blocking vehicle access.

The experimental results using real robot teams in indoor environments are shown in Figure 7(a). Our approach allows 8 Limo robots to dynamically divide into 2 subteams and successfully navigate through a narrow doorway with formation adaptation. In the scenarios of narrow hallway and tight exit, as shown in Figures 7(b) and 7(c), our approach continues to effectively facilitate subteaming and formation adaptation within a robot team with 6 robots, ensuring smooth navigation through constrained spaces in the real world. The experimental results using Limo robot teams in outdoor environments are shown in Figures 8(a), 8(b) and 8(c). The results indicate strong adaptation capability of our approach to unknown environments, subteaming and formation adaptation can well be performed on snowy and uneven terrain, where wheel slippage poses introduce large action uncertainty. By effectively coordinating robots within a team or subteam, our method achieves stable and adaptive navigation, ensuring efficient team coordination even in highly uncertain and unknown environments.



(a) 4 robots (b) 6 robots (c) 7 robots (d) 8 robots (e) CFI values across varying team sizes

Fig. 10. Quantitative results demonstrate the generalizability of STAF to robot teams of varying sizes. Figures (a)-(d) show the movement trajectories of teams with 4 to 8 robots in a circle formation to navigate through a narrow corridor with formation adaptation. Figure (e) presents the variation in CFI values across different team sizes and σ values. [Best viewed in color.]



(a) 2 subteams (b) 3 subteams (c) 4 subteams

Fig. 11. Qualitative results indicates STAF's generalizability for team division to different numbers of subteams.

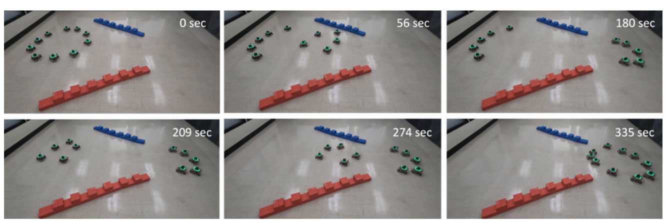


Fig. 12. Applicability of STAF to a team of holonomic robots for coordinated navigation, with the support of an external tracking system using OptiTrack.

D. Discussion

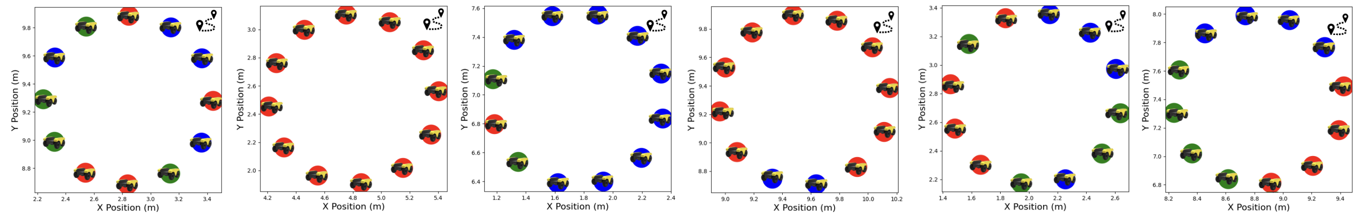
We further explore the characteristics of our STAF approach, focusing on the robustness in subteam division to noise, the generalizability to different team sizes, numbers of robots, and robot configurations. In addition, we provide an ablation study on subteam division. While circle formations are included to highlight the characteristics of our approach, we have conducted additional experiments using other formations and observed similar results.

1) Robustness in Subteam Division to Noise: In order to analyze STAF's robustness to noise in subteam division, we first present the graph cut performance in Figure 9(a) under normal conditions. These conditions are defined as the experimental setups where robot positions are uniformly distributed along a circular edge, with no noise introduced. Our STAF approach clearly achieves an even division of the robot team into two subteams, ensuring maximum adjacency within each subteam and minimum distance between subteams and their respective goals. Then, to simulate noise in robot state estimation, which is often modeled as Gaussian [67], we add standard Gaussian noise to the robot positions, as illustrated in Figure 9(b). Despite the added noise, our approach preserves a consistent subteam division, which indicates the robustness of our STAF approach against positional perturbations.

2) Generalizability to Different Team Sizes: Figures 10(a)-10(d) presents the qualitative results on formation adaptation across different team sizes, including a team consisting of 4, 6, 7, and 8 robots that maintain a circle formation. The qualitative results validate the generalizability of our STAF approach in adapting formations across varying team sizes. Figure 10(e) presents the quantitative performance using the CFI metric given different team sizes. For a small team of 4 robots, our approach maintains 87% formation integrity under a strict threshold $\sigma < 0.03$. Even with 8 robots, the team maintains at least 80% integrity, demonstrating its effectiveness in formation adaptation across different team sizes during navigation.

3) Generalizability to Different Numbers of Subteams: We further investigate the generalizability of STAF in dynamically dividing the full team into varying numbers of subteams. The experimental results, shown in Figure 11, include divisions into 2, 3, and 4 subteams. The results demonstrate that STAF effectively performs subteam division and can generalize to different numbers of subteams. Figure 6(c) illustrates a scenario in the Unity3D multi-robot simulation, where a full team of nine robots in a line formation is divided into three subteams, as the corridor is too narrow to accommodate teams larger than four robots with a line formation.

4) Applicability to Different Robot Platforms: We further demonstrate the applicability of our STAF approach to different robot types. In high-fidelity Unity3D simulation in ROS1, we test it on differential-drive Warthog robots, while real-world experiments involve Limo robots. Additionally, we assess its performance with 10 holonomic-drive robots. As illustrated in Figure 12, our STAF approach successfully enables a new team of holonomic robots to perform subteaming and adaptive



(a) ST balance (b) ST adjacency (c) ST-goals distance (d) w/o ST balance (e) w/o ST adjacency (f) w/o ST-goals distance

Fig. 13. Ablation study that analyzes the impact of different components for subteam division, including subteam balance, subteam adjacency, and subteam-goals distance. In each figure, ST is used as an abbreviation of subteam. [Best viewed in color.]

formation control to navigate through narrow corridors, with the support of an external tracking and state estimation system using OptiTrack.

5) Ablation Study on Subteam Division: We conduct an ablation study to evaluate the effectiveness of each component in the objective function defined in Eq. (3) for team division. From Figure 13(a), we observe that when optimizing only the subteam balance term, the 12 robots are evenly split into three subteams. Figure 13(b) shows that when solely maximizing adjacency, all robots are assigned to the same subteam, thus maximizing the adjacency. Figure 13(c) illustrates that when only minimizing the distance between subteam centers and their respective goals, each subteam's center aligns along a straight path toward the goals (in the upper right), ensuring goal-directed division. These results indicate the effectiveness of each component for team division. In addition, we remove each component in the objective function to evaluate their importance for team division. Figure 13(d) shows an unbalanced team division due to the removal of the subteam balance term. Figure 13(e) presents three evenly divided subteams but the robots within each subteam are not positioned together, which leads to uncompact subteams. Figure 13(f) illustrates that without minimizing the subteam-goal distance, the subteams remain in a circular formation rather than aligning toward the goal, which requires additional reconfiguration of the subteams before moving toward the goal, which increases travel time and reduces overall efficiency in reaching the destination. These results further indicate the importance of enforcing subteam balance, maximizing adjacency, and minimizing subteam-goals distance for robot team division.

V. Limitations

Our approach presents several limitations that suggest directions for future research. First, although STAF's intermediate and low levels are executed in a decentralized fashion, STAF's high level for team division is executed in a centralized fashion. One direction for future research is to decentralize the high-level team division, such as by replacing the current global graph cut optimization with a distributed consensus algorithm (e.g., gossip [68] or max-consensus [69]). These decentralized methods would enable each robot to determine its subteam based upon the information shared by its teammates through broadcasting, and iteratively reach a consensus and converge to a stable subteam assignment through negotiation. Second, the alternating training algorithm we use, which iteratively trains the high-level and joint intermediate-low levels, is considered a limitation, as it may lead to suboptimal integration of these levels and difficulties with training error propagation. In the future, we plan to integrate the high-level graph cut together with the joint intermediate-low level training into an end-to-end training algorithm, where the training error from the low level will be propagated not only to the intermediate level but also to the high level, which enables updates to the network parameters across all three levels. To achieve this, we will adopt a centralized training with decentralized execution strategy, where all levels of the hierarchy can leverage global information during training, while ensuring decentralized execution during deployment. The third limitation is that the number of subteams, as a hyperparameter, is decided manually. A future direction is to dynamically and adaptively determine this hyperparameter by selecting the minimum number of subteams such that the smallest formation of each subteam can successfully navigate through the narrowest corridor in the environment. The width of a corridor can be identified either by analyzing the environment map (using a prior map or built by a SLAM method) or through real-time robotic sensing.

VI. Conclusion

In this paper, we propose STAF as a novel approach to enable subteaming and formation adaptation during coordinated multi-robot navigation. STAF integrates three levels of robot learning with a unified hierarchical learning framework, including a high-level deep graph cut for dynamic team division into subteams, an intermediate-level graph learning component with a spring-damper model for subteam coordination and adaptive formation control, and a low-level RL policy for individual robot control to enable collision-free navigation while maintaining formation. We conduct extensive experiments to validate the effectiveness, robustness, and generalizability of STAF, utilizing ROS Gazebo simulations, a high-fidelity Unity3D simulation in ROS, and teams of physical robots operating in both indoor and outdoor environments. Experimental results demonstrate that our STAF approach enables new multi-robot capabilities for subteaming and formation adaptation, and significantly outperforms existing methods on coordinated multi-robot navigation.

References

[1] P. Gao, S. Siva, A. Micciche, and H. Zhang, "Collaborative scheduling with adaptation to failure for heterogeneous robot

teams," in *IEEE International Conference on Robotics and Automation*, 2023.
[2] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle *et al.*, "Argos: A modular, parallel, multi-engine simulator for multi-robot systems," *Swarm Intelligence*, vol. 6, pp. 271–295, 2012.
[3] T. Balch and M. Hybinette, "Social potentials for scalable multi-robot formations," in *IEEE International Conference on Robotics and Automation*, 2000.
[4] P. Singh, R. Tiwari, and M. Bhattacharya, "Navigation in multi robot system using cooperative learning: A survey," in *International Conference on Computational Techniques in Information and Communication Technologies*, 2016.
[5] J. P. Queralta, J. Taipalmaa, B. C. Pullinen, V. K. Sarker, T. N. Gia, H. Tenhunen, M. Gabbouj, J. Raitoharju, and T. Westerlund, "Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision," *IEEE Access*, vol. 8, pp. 191 617–191 643, 2020.
[6] Q. Yang and R. Parasuraman, "Needs-driven heterogeneous multi-robot cooperation in rescue missions," in *IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2020.
[7] J. L. Baxter, E. Burke, J. M. Garibaldi, and M. Norman, "Multi-robot search and rescue: A potential field based approach," *Autonomous Robots and Agents*, pp. 9–16, 2007.
[8] R. Han, S. Chen, and Q. Hao, "Cooperative multi-robot navigation in dynamic environment with deep reinforcement learning," in *IEEE International Conference on Robotics and Automation*, 2020.
[9] V. Indelman, "Cooperative multi-robot belief space planning for autonomous navigation in unknown environments," *Autonomous Robots*, vol. 42, pp. 353–373, 2018.
[10] A. Amanatiadis, C. Henschel, B. Birkicht, B. Andel, K. Charalampous, I. Kostavelis, R. May, and A. Gasteratos, "Avert: An autonomous multi-robot system for vehicle extraction and transportation," in *IEEE International Conference on Robotics and Automation*, 2015.
[11] D. Koung, O. Kermorgant, I. Fantoni, and L. Belouaer, "Cooperative multi-robot object transportation system based on hierarchical quadratic programming," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6466–6472, 2021.
[12] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *IEEE International Conference on Robotics and Automation*, 2000.
[13] B. Tang, K. Xiang, M. Pang, and Z. Zhanxia, "Multi-robot path planning using an improved self-adaptive particle swarm optimization," *International Journal of Advanced Robotic Systems*, vol. 17, no. 5, p. 1729881420936154, 2020.
[14] D. Cappello, S. Garcin, Z. Mao, M. Sassano, A. Paranjape, and T. Mylvaganam, "A hybrid controller for multi-agent collision avoidance via a differential game formulation," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 4, pp. 1750–1757, 2021.
[15] B. Reily, C. Reardon, and H. Zhang, "Leading multi-agent teams to multiple goals while maintaining communication," in *Robotics Science and Systems*, 2020.
[16] M. Goarin and G. Loianno, "Graph neural network for decentralized multi-robot goal assignment," *IEEE Robotics and Automation Letters*, 2024.
[17] S. Zhang, K. Garg, and C. Fan, "Neural graph control barrier functions guided distributed collision-avoidance multi-agent control," in *Conference on Robot Learning*, 2023.
[18] J. Blumenkamp, S. Morad, J. Gielis, Q. Li, and A. Prorok, "A framework for real-world multi-robot systems running decentralized GNN-based policies," in *International Conference on Robotics and Automation*, 2022.
[19] Y. Hu, J. Fu, and G. Wen, "Graph soft actor–critic reinforcement learning for large-scale distributed multirobot coordination," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.
[20] H. Zhu, J. Juhl, L. Ferranti, and J. Alonso-Mora, "Distributed multi-robot formation splitting and merging in dynamic environments," in *International Conference on Robotics and Automation*, 2019.
[21] B. Reily, C. Reardon, and H. Zhang, "Representing multi-robot structure through multimodal graph embedding for the selection of robot teams," in *IEEE International Conference on Robotics and Automation*, 2020.
[22] W. J. Jose and H. Zhang, "Learning for dynamic subteaming and voluntary waiting in heterogeneous multi-robot collaborative scheduling," in *IEEE International Conference on Robotics and Automation*, 2024.
[23] G. A. Cardona, K. Leahy, and C.-I. Vasile, "Temporal logic swarm control with splitting and merging," in *IEEE International Conference on Robotics and Automation*, 2023.
[24] T. Wu, K. Xue, and P. Wang, "Leader-follower formation control of usvs using APF-based adaptive fuzzy logic nonsingular terminal sliding mode control method," *Journal of Mechanical Science and Technology*, vol. 36, no. 4, pp. 2007–2018, 2022.
[25] H. Xiao and C. P. Chen, "Leader-follower consensus multi-robot formation control using neurodynamic-optimization-based nonlinear model predictive control," *IEEE Access*, vol. 7, pp. 43 581–43 590, 2019.
[26] D. Roy, A. Chowdhury, M. Maitra, and S. Bhattacharya, "Multi-robot virtual structure switching and formation changing strategy in an unknown occluded environment," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018.
[27] N. Abujabal, R. Fareh, S. Sinan, M. Baziyad, and M. Bettayeb, "A comprehensive review of the latest path planning developments for multi-robot formation systems," *Robotica*, vol. 41, no. 7, pp. 2079–2104, 2023.
[28] D. Roy, A. Chowdhury, M. Maitra, and S. Bhattacharya, "Virtual region based multi-robot path planning in an unknown occluded environment," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2019.
[29] J. Alonso-Mora, E. Montijano, T. Nägeli, O. Hilliges, M. Schwager, and D. Rus, "Distributed multi-robot formation control in dynamic environments," *Autonomous Robots*, vol. 43, pp. 1079–1100, 2019.
[30] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013.
[31] R. Bohlin and L. E. Kavraki, "Path planning using lazy PRM," in *IEEE International Conference on Robotics and Automation*, 2000.
[32] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *The International Journal of Robotics Research*, vol. 36, no. 9, pp. 1000–1021, 2017.
[33] J. Alonso-Mora, E. Montijano, M. Schwager, and D. Rus, "Distributed multi-robot formation control among obstacles: A geometric and optimization approach with consensus," in *IEEE International Conference on Robotics and Automation*, 2016.
[34] B. Reily, T. Mott, and H. Zhang, "Adaptation to team composition changes for heterogeneous multi-robot sensor coverage," in *IEEE International Conference on Robotics and Automation*, 2021.
[35] D. Koung, O. Kermorgant, I. Fantoni, and L. Belouaer, "Cooperative multi-robot object transportation system based on hierarchical quadratic programming," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6466–6472, 2021.
[36] W. Luo, S. Yi, and K. Sycara, "Behavior mixing with minimum global and subgroup connectivity maintenance for large-scale multi-robot systems," in *IEEE International Conference on*

Robotics and Automation, 2020.

[37] D. Roy, M. Maitra, and S. Bhattacharya, "Exploration of multiple unknown areas by swarm of robots utilizing virtual-region-based splitting and merging technique," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3459–3470, 2021.

[38] S. Swaminathan, M. Phillips, and M. Likhachev, "Planning for multi-agent teams with leader switching," in *IEEE International Conference on Robotics and Automation*, 2015.

[39] S. Novoth, Q. Zhang, K. Ji, and D. Yu, "Distributed formation control for multi-vehicle systems with splitting and merging capability," *IEEE Control Systems Letters*, vol. 5, no. 1, pp. 355–360, 2020.

[40] Á. Calvo and J. Capitán, "Optimal task allocation for heterogeneous multi-robot teams with battery constraints," in *IEEE International Conference on Robotics and Automation*, 2024.

[41] T. Guo, S. D. Han, and J. Yu, "Spatial and temporal splitting heuristics for multi-robot motion planning," in *IEEE International Conference on Robotics and Automation*, 2021.

[42] T. Guo and J. Yu, "Efficient heuristics for multi-robot path planning in crowded environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2023.

[43] R. Han, S. Chen, S. Wang, Z. Zhang, R. Gao, Q. Hao, and J. Pan, "Reinforcement learned distributed multi-robot navigation with reciprocal velocity obstacle shaped rewards," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 5896–5903, 2022.

[44] N. Hacene and B. Mendil, "Behavior-based autonomous navigation and formation control of mobile robots in unknown cluttered dynamic environments with dynamic target tracking," *International Journal of Automation and Computing*, vol. 18, no. 5, pp. 766–786, 2021.

[45] Q. Li, F. Gama, A. Ribeiro, and A. Prorok, "Graph neural networks for decentralized multi-robot path planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020.

[46] Z. Gao, G. Yang, and A. Prorok, "Co-optimization of environment and policies for decentralized multi-agent navigation," *arXiv*, 2024.

[47] R. Han, S. Chen, and Q. Hao, "Cooperative multi-robot navigation in dynamic environment with deep reinforcement learning," in *IEEE International Conference on Robotics and Automation*, 2020.

[48] P. Gao, Y. Shen, and M. C. Lin, "Collaborative decision-making using spatiotemporal graphs in connected autonomy," *IEEE International Conference on Robotics and Automation*, 2024.

[49] Ö. Özkahraman and P. Ögren, "Collaborative navigation-aware coverage in feature-poor environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2022.

[50] R. Wang, Z. Hua, G. Liu, J. Zhang, J. Yan, F. Qi, S. Yang, J. Zhou, and X. Yang, "A bi-level framework for learning to solve combinatorial optimization on graphs," *Advances in Neural Information Processing Systems*, vol. 34, pp. 21 453–21 466, 2021.

[51] M. Bettini, A. Shankar, and A. Prorok, "Heterogeneous multi-robot reinforcement learning," *arXiv*, 2023.

[52] J. Zhang, J. Ge, S. Li, S. Li, and L. Li, "A bi-level network-wide cooperative driving approach including deep reinforcement learning-based routing," *IEEE Transactions on Intelligent Vehicles*, 2023.

[53] W. Wang, L. Mao, R. Wang, and B.-C. Min, "Multi-robot cooperative socially-aware navigation using multi-agent reinforcement learning," in *IEEE International Conference on Robotics and Automation*, 2024.

[54] P. Feng, J. Liang, S. Wang, X. Yu, X. Ji, Y. Chen, K. Zhang, R. Shi, and W. Wu, "Hierarchical consensus-based multi-agent reinforcement learning for multi-robot cooperation tasks," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2024.

[55] B. Bischoff, D. Nguyen-Tuong, I. Lee, F. Streichert, A. Knoll *et al.*, "Hierarchical reinforcement learning for robot navigation," in *Proceedings of The European Symposium on Artificial Neural Networks, Computational Intelligence And Machine Learning*, 2013.

[56] Y. Jin, S. Wei, J. Yuan, and X. Zhang, "Hierarchical and stable multiagent reinforcement learning for cooperative navigation control," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 1, pp. 90–103, 2021.

[57] J. Wöhlke, F. Schmitt, and H. van Hoof, "Hierarchies of planning and reinforcement learning for robot navigation," in *IEEE International Conference on Robotics and Automation*, 2021.

[58] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, "Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14 413–14 423, 2020.

[59] W. Zhu and M. Hayashibe, "A hierarchical deep reinforcement learning framework with high efficiency and generalization for fast and safe navigation," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 5, pp. 4962–4971, 2022.

[60] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *International Conference on Representation Learning*, 2017.

[61] Z. Deng, P. Gao, W. J. Jose, and H. Zhang, "Multi-robot collaborative navigation with formation adaptation," *arXiv*, 2024.

[62] C. Gabellieri, A. Palleschi, and L. Pallottino, "Force-based formation control of omnidirectional ground vehicles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2021.

[63] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv*, 2017.

[64] Q. Zou, Q. Sun, L. Chen, B. Nie, and Q. Li, "A comparative analysis of LiDAR SLAM-based indoor navigation for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6907–6921, 2021.

[65] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[66] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[67] T. D. Barfoot, *State estimation for robotics*. Cambridge University Press, 2024.

[68] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.

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