2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Field Evaluation of a Prioritized Path-Planning Algorithm for Heterogeneous Agricultural Tasks of Multi-UGVs

Yuseung Jo[1,2] and Hyoung Il Son[1,2,3], *Senior Member, IEEE,*

Abstract— **This paper introduces a prioritized path-planning algorithm for heterogeneous tasks performed by multiple unmanned ground vehicles (UGVs) in agricultural environments. The algorithm considers varying robot priorities, thereby extending the traditional multi-agent path finding (MAPF) approach. The proposed algorithm is evaluated in scenarios occurring during representative agricultural operations: harvesting and transportation. An experimental validation is conducted in agriculture-like settings by using multiple simultaneous localization and mapping systems and navigation systems. The results revealed that the path of agent$_1$, which was assigned the highest priority in both the indoor and outdoor environments, was shortened considerably (3.38 m, 3.6 m, and 5.6 m, respectively). Especially in the face scenario, the sum of changes in distance, calculated using the proposed algorithm was negative, meaning that traffic congestion in the multi-robot system used in the experiment was alleviated without the need for inter-robot communication.**

I. INTRODUCTION

Owing to the exponential population growth worldwide, interest in food resources is increasing. Consistent with this trend, the agricultural sector is expanding steadily. In this context, the increased labor burden in large-scale agricultural fields poses a new challenge for the agriculture sector. Agricultural robots are an attractive option for solving this challenge, and the above-described circumstances have motivated active research on agricultural robots. However, the development of a multi-purpose agricultural robot that can be deployed in an unstructured and unknown environment is difficult. These challenges highlight the importance of designing task-specific robots by using various robotic platforms, including unmanned ground vehicles (UGVs) [1], unmanned aerial vehicles (UAVs) [2], mobile manipulators [3], and aerial manipulators [4]. Among them, UGVs, as reliable platforms, have served as the basis of numerous studies. Especially in the agriculture domain, the development of UGV-based agricultural robots combined with autonomous driving has reached the commercialization stage.

However, these studies were conducted by focusing primarily on a single robot. Because of this limitation, it is difficult to deploy the robots developed in the aforementioned studies in real-world applications. To overcome this challenge, the multi-robot system (MRS) has emerged as one of the most effective approaches [5], [6], [7]. Specifically, collaboration among heterogeneous robots has the potential to significantly enhance the efficiency of robots [8].

Deployment of multi-robots should be studied carefully because it greatly increases system complexity [9]. A team of robots can be controlled using two main types of control methods: centralized and decentralized. The decentralized control method is beneficial from the perspectives of local information utilization and system scalability [10]. However, the potential for competition and the local minima encountered in task optimization are limitations of the decentralized control method in the context of agricultural tasks [11]. Further, in centralized control, the robot team must follow the constraints of a single robot, must not collide with each other, and move along optimal paths to the extent possible. To address these challenges, a task-aware path-planning algorithm for a centralized system should be proposed [12].

In this study, the authors propose a prioritized path-planning algorithm for the heterogeneous tasks of multi-UGVs used in the agriculture domain. The proposed algorithm targets multiple agricultural robots performing heterogeneous agricultural tasks (especially harvesting and transportation) based on the multi-agent pathfinding problem (MAPF) [13]. In addition, it considers the priority of a robot for performing heterogeneous tasks in agriculture. The priorities of multiple robots are added to an existing MAPF algorithm to realize global work optimization.

The contributions and novelty of this study can be summarized as follows:

1) We propose a prioritized path planning algorithm for a heterogeneous agricultural robot team that can allocate priorities to improve global work efficiency.
2) We evaluate the proposed algorithm in the harvesting and transportation scenarios by conducting a field experiment.
3) We configure a fully autonomous safe interval MRS that can ensure the interval of each robot for implementing a discretized algorithm in a continuous world.

II. PATH PLANNING AND COORDINATION ALGORITHM

Agricultural environments are typically dense and contain numerous obstacles. In such environments, multiple robots cannot travel freely because their movement pathways are narrow [14]. In this case, planning the paths of a large number of robots efficiently is essential [15]. Traditionally, this problem is considered a MAPF problem [16], [17]. In

[1]Y. Jo and H. I. Son are with the Department of Convergence Biosystems Engineering, Chonnam National University, 77 Yongbong-ro, Gwangju 61186, Republic of Korea (e-mail: chossbb68@jnu.ac.kr; hison@jnu.ac.kr).
[2]Y. Jo and H. I. Son are with the Interdisciplinary Program in IT-Bio Convergence System, Chonnam National University, Yongbong-ro 77, Gwangju 61186, Republic of Korea
[3]H. I. Son is with the Research Center for Biological Cybernetics, Chonnam National University, Yongbong-ro 77, Gwangju 61186, Republic of Korea

Algorithm 1 CBS-HT.

```
Root.constraint = ∅
Root.solution = find individual path by the low level()
Root.cost = sum of individual cost of Root.solution
Insert Root to OPEN // OPEN = empty set
for(let i = 0; i ≤ N; i++)
    allocate v_i to each workspace
≺_main ← ≺_init
while OPEN not empty do
    P ← best node from OPEN // lowest cost
    Validate the paths in P untill a conflict occurs
    if P has no conflict then
        return P.solution // P is goal
end while
C ← first conflict (a_i, a_j, v, t) in P
foreach agent a_i in C do
    A ← new node
    A.constraint ← P.constraint + (a_i, v, t)
    A.solution ← P.solution
    Compare(≺_main,i, ≺_main,j)
    if ≺_main,i is higher/smaller than ≺_main,j then
        if a_i = transporting and a_j = harvesting
            ≺_main,i −1
        elif a_i = harvesting and a_j = harvesting
            Compare(R.solution.cost(a_i, a_j))
            ≺_main −1 to agent with higher cost
        else
            Compare(R.solution.cost(a_i, a_j))
            ≺_main −1 to agent with higher cost
            return ≺_main
    Add constraint to agent with lower priority
        Update A.solution
        A.cost = sum of A.solution.cost
        if A.cost < ∞
            Insert A to OPEN
```

this section, we provide detailed descriptions of the MAPF theory and the proposed algorithm.

A. Problem Statement

The MAPF problem is a generalized form of the single-agent pathfinding problem extended to multiple agents [13]. Single-agent pathfinding algorithms such as A* search for paths between two vertices s_1, g_1 in an environment represented as a mapped graph G. Similarly, the MAPF problem involving k agents can be represented by the tuple $< G, s, g >$.

$$G = (V, E), \tag{1}$$

$$s, g : [1, \cdots, k] \rightarrow V, \tag{2}$$

where V represents a vertex, and E represents an edge. The environment is mapped onto a graph G, where the vertices can be occupied by agents. s denotes the relationship among the initial positions of the agents, and g represents the relationship among the goal positions of the agents. The MAPF problem can be expressed as follows:

$$\pi = (a_1, \cdots, a_n), \tag{3}$$

$$A = (\pi_1, \cdots, \pi_i), \tag{4}$$

$$\pi_i [|\pi|] = g(i). \tag{5}$$

This problem involves finding a set of collision-free paths A for agents in an environment mapped onto a graph $G(V, E)$. Here, π_i represents the path of agent i, which consists of a set of actions a_n of agent i. $\pi_i[t]$ denotes the position of agent i at discrete time step t, $g(i)$ denotes the goal position of agent i, and $i, j \in \{1, \cdots, k\}$. In the MAPF problem, collisions are defined as follows:

$$\pi_i [t] = \pi_j [t], \tag{6}$$

$$\pi_i [t+1] = \pi_j [t+1]. \tag{7}$$

The MAPF problem has a specific cost function [16]. Traditionally, the typical approach has entailed global minimization of the sum of costs of agents. Accordingly, in this paper, the cost function π is defined as follows:

$$\pi = \sum_{1 \le i \le k} |\pi_i|. \tag{8}$$

In the discretized environment, one action consumes one time step. Thus, reducing π physically leads to the global work optimization.

B. Prioritized Planning Algorithm for Heterogeneous Tasks

Agriculture requires collaborative work (e.g., harvesting, transportation, spraying, and monitoring) [18]. This differentiates it from the typical industrial multi-robot system that perform a single task or multiple tasks [19], [20]. Additionally, agricultural are sequential in nature, and the sequence or order of tasks is the same in most agricultural field (e.g., transportation after harvest). When performing heterogeneous tasks, prioritized planning is one of the most efficient approaches to solving MAPF problems [17], [21]. Prioritized planning assigns priority to agents in the path-planning process [22]. The MRS path is planned, and conflicts between agents, including those assigned the highest priority to those assigned the lowest priority are avoided.

The proposed algorithm is based on the conflict-based search (CBS) algorithm [23], which is the most commonly used conventional MAPF algorithm. The CBS algorithm returns a conflict-free path by performing a global search for each agent with constraints C, as follows:

$$C = (a_i, a_j, v, t), \tag{9}$$

where a_i and a_j denote agent i and agent j, respectively. v and t represent a collision between agents at a specific vertex

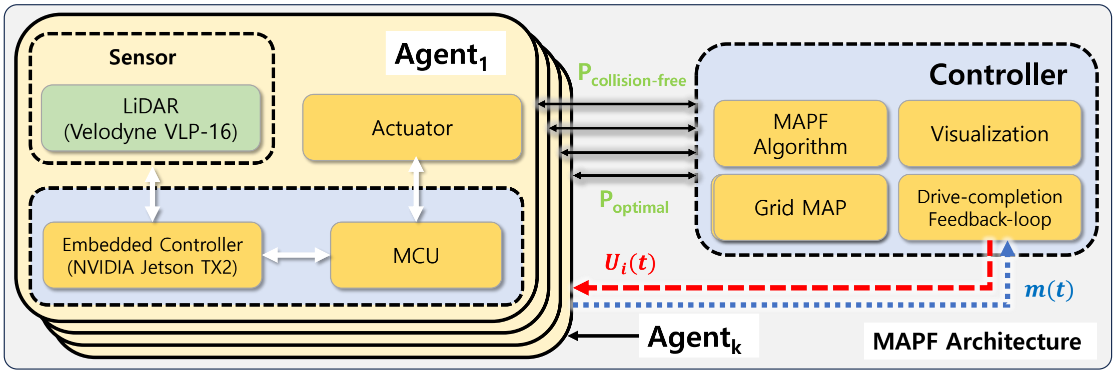

Fig. 1. System Overview.

Algorithm 2 Create Grid Map.

function CREATEGRIDMAP(width, height, cellSize)
  gridMap ← empty 2D array with dimensions (width/cellSize) × (height/cellSize)
  for x from 0 to (width/cellSize) − 1 **do**
    for y from 0 to (height/cellSize) − 1 **do**
      gridMap[x × (height/cellSize) + y] ← CREATEGRIDCELL(x × cellSize, y × cellSize, cellSize)
    end for
  end for
  return gridMap
end function

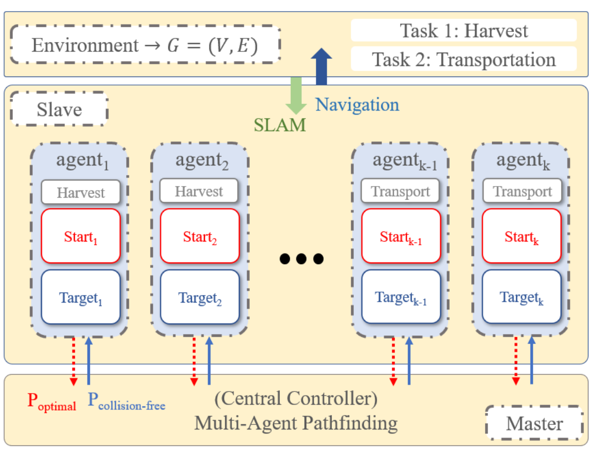

Fig. 2. Schematic of cooperative multi-robot navigation system.

and at a specific time step, respectively. The pseudo-code of the proposed algorithm is presented in Algorithm 1. The CBS algorithm finds the optimal solution of a multi-agent system by generating a set of optimal paths for a single agent (lines 2, 8-12). To this end, it compares sets of optimal paths (line 9). In this process, agent$_i$ and agent$_j$ may collide at vertex v in time step t (line 14). The algorithm compares the optimal path of agent$_i$ to that of agent$_j$ with added constraints and vice versa (lines 15-18). This process is repeated until a solution is found and a global search of all paths of all agents becomes possible (line 34).

The proposed algorithm CBS-HT addresses the limitations of global search by comparing priorities $\prec$ and imposing constraints on robots that are less critical to the task (lines 5-7). To this process, it compares the predefined priorities of agent$_i$ and agent$_j$ (line 19). Constraints are imposed on low-priority agents, which is not done in the basis algorithm, that is, CBS. The robots on which these constraints are imposed refrain from moving to vertex v at time step t, implying that they either wait or find an alternative path (line 30).

However, this method does not address the robots that share the same priorities. This challenge is tackled by adjusting priorities based on the robot type and the associated cost π (line 19-30). In a previous work, we defined this priority and verified the resulting approach by performing a simulation [24]. The proposed system can be used not only for harvesting and transportation, as done in this study, but also for other agricultural tasks by determining the priorities of those tasks.

III. SYSTEM CONFIGURATION

To evaluate the effectiveness of the proposed algorithm, we configured a cooperative multi-robot navigation system. An overview of the configured system is depicted in Fig. 1. A schematic of the navigation system is depicted in Fig. 2. A LiDAR-based simultaneous localization and mapping (SLAM) and navigation system was installed on all agents. One randomly selected robot mapped the environment and shared the resulting map with all the other robots. Robot localization and map updates occurred concurrently, and they were refined using Bayesian filter-based beliefs [25].

Control of the four-wheeled mobile robots used herein can be expressed as follows:

$$U_{(1:t)} = \left(V_{(1:t)}, w_{(1:t)}\right), \tag{10}$$

where $U_{(1:t)}$ denotes control inputs, and V and w denote the linear and angular velocities of a robot, respectively. The process of updating the states and maps of the robots is expressed as follows:

$$P(X_t | Z_{(1:t)}, U_{(1:t)}), \tag{11}$$

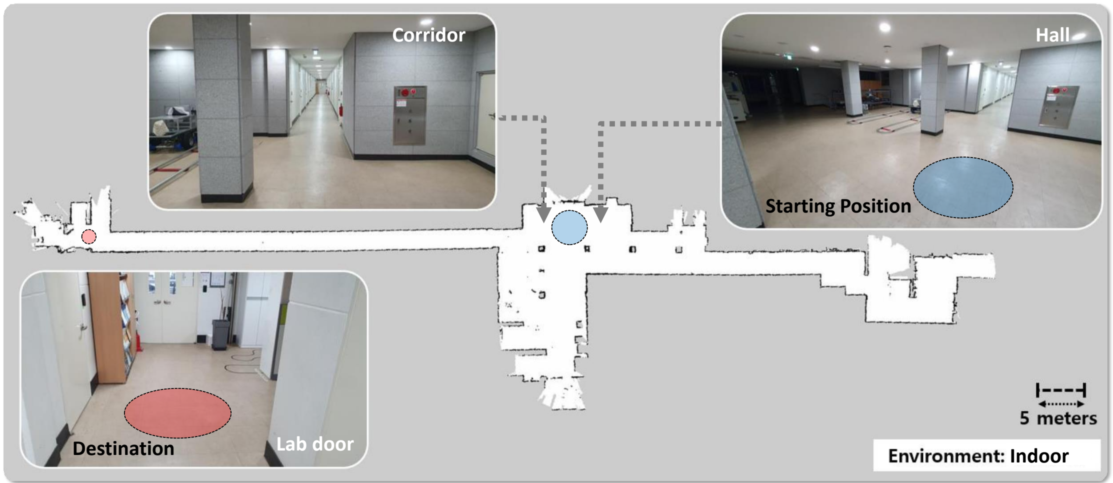

Fig. 3. Summarized map of the indoor test. Two robots start from the hall in the center of the map and move to the lab door on the left. The length of the corridor, which is the ground truth of the traveling distance, is 30m.

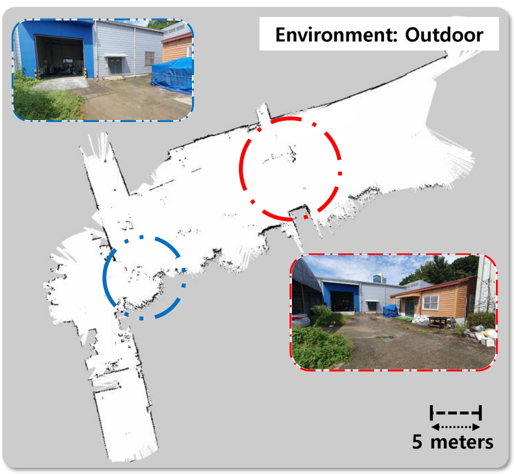

Fig. 4. Summarized map of the outdoor test. The starting point and destination of each robot is defined according to the experimental scenario.

$$P(M|Z_{(1:t)}, X_{(1:t)}), \tag{12}$$

where $Z_{(1:t)}$ denotes sensor measurements, $X_{(1:t)}$ denotes robot pose estimates, and M denotes the environment map. The Bayesian filter-based beliefs are updated as follows:

$$P(X_t, M|Z_{(1:t)}, U_{(1:t)}). \tag{13}$$

The MAPF problem assumes an environment that is represented by a grid. Consequently, the generated map must be divided into grids. The pseudo-code of the grid-partitioning algorithm is presented in Algorithm 2. In this context, "width" and "height" refer to the horizontal and vertical dimensions of the entire map. The variable "cell-Size" indicates the size of each grid cell, and herein, it is determined considering the lengths and widths of the robots.

As illustrated in Fig. 2, in the configured system, there is no direct communication among the robots; instead, the robots communicate exclusively with a master node. This open-loop navigation system is limited in terms of adapting to environmental changes and coordinating with other robots, especially when operating within a discretized time domain where synchronization-related challenges may arise. Therefore, we implemented a closed-loop control system to ensure that all robots detect their arrival at waypoints and proceed to the next waypoint. The control input $U_i(t)$ of agent i at time step t can be expressed as follows:

$$U_i(t) = K(P_i, \theta_i, X_i^{\text{way}}(t+1), m(t)), \tag{14}$$

$$m(t) \in \{0, 1\}, \tag{15}$$

where m indicates whether all the robots have reached waypoint $X_i^{\text{way}}(t)$ at time step t. If m is 1, agent i needs to adjust its position P_i and orientation θ_i to reach the next waypoint $X_i^{\text{way}}(t+1)$. K denotes the control algorithm, and in this paper, PID control is used.

IV. Experimental Design

The robotic platforms configured for the experiments relies on VLP-16 LiDAR, an inertial measurement unit sensor, and a wheel odometry. Because agricultural environments are representative GPS-denied environments, a GPS sensor was not included in the sensor network. Each robot configuration had its own controller and an independent SLAM and navigation system. For SLAM, we used the highly reliable gmapping algorithm [25].

A. Experimental Scenario

The experiment was conducted considering two cases: baseline and CBS-HT. In the baseline case, we constructed a naive MRS that did not use the proposed algorithm. In this case, the A* algorithm was used in the navigation system instead. Consequently, each agent attempted to traverse the optimal path and treated other robots as dynamic obstacles detected by its sensors. The experimental environment was categorized into two settings: indoor and outdoor.

1) Indoor Test: The indoor environment was a part of a building with a long and featureless corridor in the center. Given the small-scale setting, the robot team consisted of two robots. The robot team embarked from the initial position in the hallway and proceeded through the corridor toward one end of the building. Each robot was assigned a priority order from first to second. The SLAM map for indoor testing is depicted in Fig. 3.

2) Outdoor Test: In the outdoor environment, the team consisted of three robots, namely two harvesting robots and one transport robot. The tests aimed to simulate coordination among multiple robots, a scenario that is commonly encountered in high-density environments, such as greenhouses or orchards. The SLAM map for the outdoor tests is depicted in Fig. 4. The features of the outdoor environment were sparser than those of the indoor environment, and uncertain and unpredictable features such as bushes were used for mapping. This may have lowered the quality of mapping relatively. However, we confirmed that the localization error was within 0.09 m in this environment. This localization error is small compared to the grid size of 990 mm by 699 mm, which was determined by the size of the robot. Each robot was assigned a priority order from first to third.

B. Performance Metrics

Unlike algorithms that assume a discrete-time domain, a robotic system operates continuously. Therefore, when evaluating the proposed algorithm within a robotic system, metrics other than cost π are needed. To assess the system, we employed the sum of travel distance if agent i in a single travel instance, denoted by D_i:

$$D_i = \int_0^t \sqrt{\frac{\mathrm{d}x_i^2}{\mathrm{d}t} + \frac{\mathrm{d}y_i^2}{\mathrm{d}t}} dt, \tag{16}$$

where x_i and y_i denote the x and y coordinates of agent a_i, respectively. Because the proposed algorithm includes waiting actions, the path length does not exclusively represent time. Therefore, the arrival time T_i of each robot in one travel instance should be considered as well:

$$T_i = \frac{1}{N}\sum_1^N t_i, \tag{17}$$

where N represents the number of repetitions of the experiment. According to the definition of the MAPF problem, the collision count N_c should be compared as well:

$$N_c = \sum_1^N n_c, \tag{18}$$

where n_c denotes the collisions that occurred in a single driving instance. The experiments were conducted with five repetitions of the indoor and outdoor scenarios.

V. Experimental Results

The results of the indoor experiments are summarized in Table 1. These results clearly demonstrate that the proposed algorithm effectively prevented collisions in the indoor experiments. Notably, it reduced the total travel distance of the robots by 3.26 m. Of particular significance was the fact that the distance traveled by Agent 1 decreased remarkably by 3.38 meters, which underscored the efficacy of the proposed priority-based approach. Moreover, the arrival times of each robot decreased by 1.1 s, and the travel time of Agent 1 decreased by 1.5 s.

The results of the outdoor experiments are summarized and depicted in Table 2 and Fig. 5. In both scenarios, the highest-priority agent, Agent 1, experienced the most significant reduction in travel time. In the ”crossover” scenario, the lowest-priority agent, that is, Agent 3, had longer travel times owing to actions involving bypassing or waiting. Notably, agricultural tasks are sequential (e.g., harvest before transportation), and therefore, the algorithm allowed relatively slow deployment of the lower-priority robots.

Conversely, in ”face” scenarios, both the distance and time of all agents decreased. A comparison between Figs. 5(c) and (d) revealed that these results were obtained because the proposed algorithm effectively solved the congestion situation. Without inter-robot communication, the robots would attempt to travel their own optimal paths. In this case, traffic congestion, as shown in Fig. 5(c), would likely occur. Moreover, in the above case, all agents attempt to avoid dynamic obstacles by using their own computational resources. This would lead to unnecessary and uncertain planning and traveling. Furthermore, as depicted in Fig 5(a), Agent 1 may appear to move back and forth and left to right without moving significantly out of its place.

VI. Conclusions

In this paper, we presented a prioritized path planning algorithm tailored for managing heterogeneous tasks involving multiple UGVs in agricultural settings with a focus on harvesting and transportation. The algorithm builds on traditional MAPF approaches and accommodates the varied priorities of different robots. Experimental validation conducted in dense agricultural-like environments by using multiple SLAM and navigation systems demonstrated the effectiveness of the algorithm. Additionally, a closed-loop feedback mechanism was introduced to address synchronization.

In the experiments, the environmental conditions and number of robots involved were varied. According to the results, the algorithm consistently yielded improvements in term of MRS reliability and reduced unnecessary movements. This suggests that the proposed algorithm increased the efficiency of robots in agricultural tasks.

Acknowledgment

This work was supported by a National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2023R1A2C1003701).

TABLE I

RESULTS OF INDOOR TEST.

Environment	Agent ($\prec$)	Case	D_i (m)	T_i (sec)	N_c (times)
Indoor	Agent 1 (1st)	Baseline	34.40	22.4	1
		CBS-HT	31.02 (-3.38)	20.9 (-1.5)	0
	Agent 2 (2nd)	Baseline	33.14	22.6	1
		CBS-HT	33.26 (+0.12)	23.0 (+0.4)	0

TABLE II

RESULTS OF OUTDOOR TESTS.

Environment	Scenario	Agent ($\prec$)	Case	D_i (m)	T_i (sec)	N_c (times)
Outdoor	Crossover	Agent 1 (1st)	Baseline	31.8	29.3	0
			CBS-HT	28.2 (-3.6)	24.1 (-5.2)	0
		Agent 2 (2nd)	Baseline	13.5	17.6	0
			CBS-HT	14.7 (+1.2)	19.2 (+1.6)	0
		Agent 3 (3rd)	Baseline	23.1	22.5	0
			CBS-HT	28.4 (+5.3)	30.0 (+7.5)	0
	Face	Agent 1 (1st)	Baseline	18.1	20.3	0
			CBS-HT	12.5 (-5.6)	14.0 (-6.3)	0
		Agent 2 (2nd)	Baseline	17.8	20.6	0
			CBS-HT	15.3 (-2.5)	18.4 (-2.2)	0
		Agent 3 (3rd)	Baseline	19.7	26.1	0
			CBS-HT	17.0 (-2.7)	20.3 (-5.8)	0

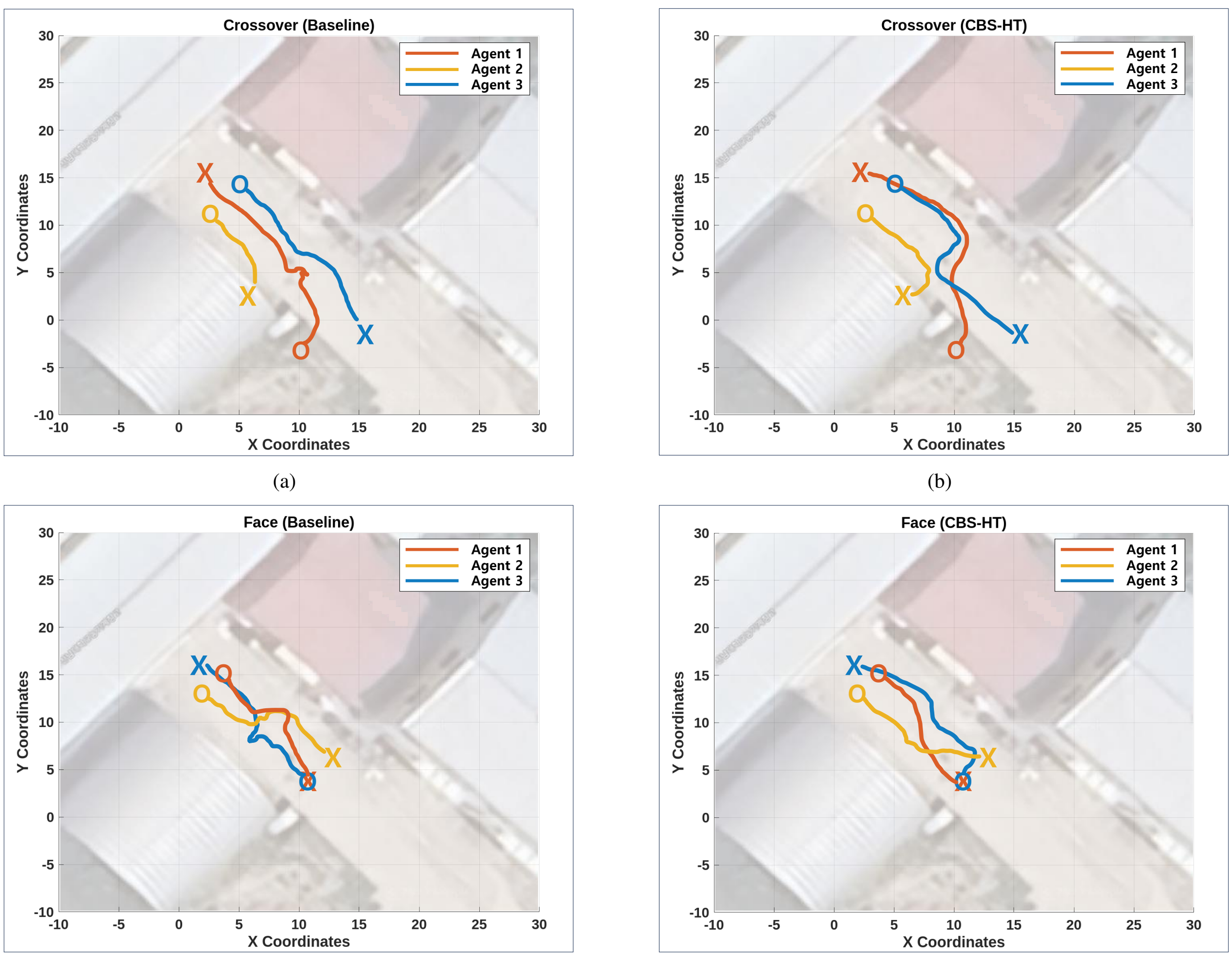

Fig. 5. Trajectories of 3 robots and their priorities in outdoor test, the "O" symbol means the starting point and the "X" symbol means the destination: (a) crossover: baseline, (b) crossover: CBS-HT, (c) face: baseline, (d) face: CBS-HT.

References

[1] J. Kim, J. Seol, S. Lee, S.-W. Hong, and H. I. Son, "An intelligent spraying system with deep learning-based semantic segmentation of fruit trees in orchards," in *2020 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2020, pp. 3923–3929.

[2] J. Kim, S. Kim, C. Ju, and H. I. Son, "Unmanned aerial vehicles in agriculture: A review of perspective of platform, control, and applications," *IEEE Access*, vol. 7, pp. 105 100–105 115, 2019.

[3] Y. Park, J. Seol, J. Pak, Y. Jo, C. Kim, and H. I. Son, "Human-centered approach for an efficient cucumber harvesting robot system: Harvest ordering, visual servoing, and end-effector," *Computers and Electronics in Agriculture*, vol. 212, p. 108116, 2023.

[4] C. D. Bellicoso, L. R. Buonocore, V. Lippiello, and B. Siciliano, "Design, modeling and control of a 5-dof light-weight robot arm for aerial manipulation," in *2015 23rd Mediterranean Conference on Control and Automation (MED)*, 2015, pp. 853–858.

[5] C. Hu, C. Hu, D. He, and Q. Gu, "A new ros-based hybrid architecture for heterogeneous multi-robot systems," in *The 27th Chinese Control and Decision Conference (2015 CCDC)*. IEEE, 2015, pp. 4721–4726.

[6] C. Ju and H. I. Son, "A hybrid systems-based hierarchical control architecture for heterogeneous field robot teams," *IEEE Transactions on Cybernetics*, 2021.

[7] A. Mohiuddin, T. Tarek, Y. Zweiri, and D. Gan, "A survey of single and multi-uav aerial manipulation," *Unmanned Systems*, vol. 8, no. 02, pp. 119–147, 2020.

[8] C. Ju, J. Kim, J. Seol, and H. I. Son, "A review on multirobot systems in agriculture," *Computers and Electronics in Agriculture*, vol. 202, p. 107336, 2022.

[9] J. Yu and S. LaValle, "Structure and intractability of optimal multi-robot path planning on graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 27, no. 1, 2013, pp. 1443–1449.

[10] S. Patil, V. Vyatkin, and M. Sorouri, "Formal verification of intelligent mechatronic systems with decentralized control logic," in *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)*. IEEE, 2012, pp. 1–7.

[11] I. Draganjac, D. Miklić, Z. Kovačić, G. Vasiljević, and S. Bogdan, "Decentralized control of multi-agv systems in autonomous warehousing applications," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 4, pp. 1433–1447, 2016.

[12] F. J. Mendiburu, M. R. Morais, and A. M. Lima, "Behavior coordination in multi-robot systems," in *2016 IEEE International Conference on Automatica (ICA-ACCA)*. IEEE, 2016, pp. 1–7.

[13] A. Felner, R. Stern, S. Shimony, E. Boyarski, M. Goldenberg, G. Sharon, N. Sturtevant, G. Wagner, and P. Surynek, "Search-based optimal solvers for the multi-agent pathfinding problem: Summary and challenges," in *Proceedings of the International Symposium on Combinatorial Search*, vol. 8, no. 1, 2017, pp. 29–37.

[14] A. Ahmadi, L. Nardi, N. Chebrolu, and C. Stachniss, "Visual servoing-based navigation for monitoring row-crop fields," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 4920–4926.

[15] M. Edmonds and J. Yi, "Efficient multi-robot inspection of row crops via kernel estimation and region-based task allocation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 8919–8926.

[16] R. Stern, N. Sturtevant, A. Felner, S. Koenig, H. Ma, T. Walker, J. Li, D. Atzmon, L. Cohen, T. Kumar, *et al.*, "Multi-agent pathfinding: Definitions, variants, and benchmarks," in *Proceedings of the International Symposium on Combinatorial Search*, vol. 10, no. 1, 2019, pp. 151–158.

[17] K. Brown, O. Peltzer, M. A. Sehr, M. Schwager, and M. J. Kochenderfer, "Optimal sequential task assignment and path finding for multi-agent robotic assembly planning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 441–447.

[18] P. Tokekar, J. Vander Hook, D. Mulla, and V. Isler, "Sensor planning for a symbiotic uav and ugv system for precision agriculture," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 5321–5326.

[19] X. Cai, B. Schlotfeldt, K. Khosoussi, N. Atanasov, G. J. Pappas, and J. P. How, "Non-monotone energy-aware information gathering for heterogeneous robot teams," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 8859–8865.

[20] Z. Ren, S. Rathinam, and H. Choset, "Loosely synchronized search for multi-agent path finding with asynchronous actions," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 9714–9719.

[21] M. Čáp, P. Novák, A. Kleiner, and M. Seleckỳ, "Prioritized planning algorithms for trajectory coordination of multiple mobile robots," *IEEE transactions on automation science and engineering*, vol. 12, no. 3, pp. 835–849, 2015.

[22] J. Kottinger, S. Almagor, and M. Lahijanian, "Conflict-based search for multi-robot motion planning with kinodynamic constraints," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 13 494–13 499.

[23] W. Hönig, S. Kiesel, A. Tinka, J. Durham, and N. Ayanian, "Conflict-based search with optimal task assignment," in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems*, 2018.

[24] Y. Jo and H. I. Son, "A path planning and coordination algorithm for heterogeneous tasks of multi-ugv in smart farm: Work in progress," in *2022 22nd International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 2022, pp. 1387–1390.

[25] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters," *IEEE transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007.