Multiple Unmanned Ground Vehicles Cooperative Trajectory Planning with Flexible Safety Guarantees

Yicong Liu
School of Vehicle and Mobility
Tsinghua University
Beijing, China
liuyicon20@mails.tsinghua.edu.cn

Chaoyi Chen
School of Vehicle and Mobility
Tsinghua University
Beijing, China
cerochen@foxmail.com

Mengchi Cai
School of Vehicle and Mobility
Tsinghua University
Beijing, China
caimengchi1996@163.com

Heye Huang
School of Vehicle and Mobility
Tsinghua University
Beijing, China
hhy18@mails.tsinghua.edu.cn

Qing Xu
School of Vehicle and Mobility
Tsinghua University
Beijing, China
qingxu@tsinghua.edu.cn

Jianqiang Wang*
School of Vehicle and Mobility
Tsinghua University
Beijing, China
wjqlws@tsinghua.edu.cn

2023 IEEE International Conference on Unmanned Systems (ICUS) | 979-8-3503-1630-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICUS58632.2023.10318414

***Abstract*—The growing demand for unmanned ground vehicles (UGVs) in various applications necessitates the development of efficient trajectory planning techniques that can ensure both safety and cooperative behavior. This paper presents a novel approach for multi-UGV cooperative trajectory planning with flexible safety guarantees. The proposed method addresses the challenge of finding safe trajectories for multiple UGVs by incorporating risk assessment into the planning process. A field-based risk assessment model is used to quantify the risk levels between UGVs, and a predefined threshold is adopted to define risk conflicts. To solve the cooperative trajectory planning problem, an interactive trajectory planning approach is introduced. It combines path planning with trajectory validation to ensure safety and address the risk conflicts. Vehicle characteristics are considered in the planning process. The simulation results show that the proposed interactive trajectory planning algorithm achieves a high success rate while maintaining a maximal risk level below a predefined threshold. It also outperforms the benchmark algorithms in terms of safety performance, with reduced risk levels and improved adaptability to varying conditions.**

***Index Terms*—Multi-UGVs, trajectory planning, risk assessment, safety guarantee.**

I. Introduction

Cooperative trajectory planning is a fundamental problem for the implementation of theories and methods for multiple Unmanned Ground Vehicles (UGVs). The problem can be defined as follows: multiple UGVs are given, along with their start and goal locations in an environment; the task is to cooperatively plan safe trajectories from the start to the goal positions. Multi-UGV trajectory planning finds applications in various areas, including structured or unstructured environments and for civilian or special purposes, for instance, formation control [1], transport management [2], area search and exploration [3], etc. Unlike path planning, trajectory planning contains speed profiles and can be executed directly by vehicle controllers. Hence, it provides a more precise representation of multi-UGV motion and coordination. In almost all multi-UGV applications, guaranteeing trajectory safety is vital. While collision-free is a fundamental requisite, incorporating risk assessment can provide additional safety margins that could be useful when accounting for the inevitable vehicle perception and control imperfections [4], [5].

As a subroutine of cooperative trajectory planning, Multi-Agent Path Finding (MAPF) plans collision-free paths for multiple agents. MAPF is an NP-hard problem. There are three categories of approaches to solve MAPF: prioritized planning [6], [7], extended A* [8], [9], and hierarchical search [10], [11]. In prioritized planning, agents are planned one by one in a designated order. This includes hierarchical cooperative A* [6] and its derivative with dynamic planning windows and priorities [7]. Extended A* simplifies the search branch using operator decomposition [8] or independence detection for problem decoupling and dimension reduction [9]. Conflict-Based Search (CBS) is an influential hierarchical search approach [10]. Its higher level resolves conflicts among agents, while the lower level plans constrained paths for individual agents. To improve CBS in practical scenarios, researchers have proposed enhancements considering agent shapes and kinematics [11]. However, these MAPF approaches mainly focus on collision-free path planning with simplified agent models, which may not be directly applicable or provide sufficient safety margins in real-world scenarios.

The overall goal of MAPF solvers is to find collision-free cooperative paths. Collisions include vertex, edge, following, cycle, and swapping conflicts between ideal mass-point agents [12]. To bridge the gap between ideal agents and practical robots, larger agents with geometric shapes and kinematic constraints are examined [11], [13]. However, these solutions have two drawbacks: 1) there is still a disparity between paths and trajectories; 2) the collision-free guarantee is not resistant to practical sensing inaccuracies [14] and

*Corresponding Author

This work was supported by the National Natural Science Foundation of China through Grant No.52221005.

control errors [5]. In the context of UGVs, risk assessment can enhance safety by providing flexible safety margins. Field-based approaches for assessing multidimensional risks, such as Artificial Potential Field (APF) [15], have been developed for several decades. A comprehensive model of driving safety, addressing the interaction among drivers, vehicles, and roads, has been proposed and applied to autonomous vehicles [16]. Risk assessment also considers collision probability and intention randomness in traffic environments [17], [18]. Therefore, risk assessment can be used in multi-UGV trajectory planning to generate cooperative trajectories with predefined risk levels.

To drive the UGVs, vehicle controllers need trajectories instead of paths or rough trajectories solved by MAPF methods. A trajectory, with spatio-temporal information, determines vehicle motion in an environment. Trajectory planning can be classified as direct or decoupled, depending on how the spatio-temporal information is solved. As a prevalent approach, the optimization-based planner uses cooperative paths to solve optimization problems while defining corresponding trajectories [19]. Mixed integer programming can be used to optimally solve the trajectory planning problem [20]. Trajectory planning is formulated as mixed-integer quadratic programming based on a cooperative maneuver automaton to calculate trajectories effectively and dependably [21]. Furthermore, convex or nonconvex optimizations are used to compute safe local trajectories based on the global planner for multi-robot collision avoidance in dynamic environments [22]. Introducing a receding horizon to the graph search algorithm improves the efficiency of trajectory planning, which makes it more efficient when using motion primitives for multiple vehicles [23].

These trajectory planners are beneficial in considering vehicle kinematics through the design of objectives and constraints. Recently, researchers have applied optimization-based trajectory planning to multiple vehicles in the lane-changing scenario [1]. However, converting the given path to a fine trajectory and utilizing the MAPF solution for multi-UGV coordination would weaken the commonly used collision-free safety guarantee. Therefore, an interaction between the MAPF solver and the trajectory planner is necessary to finalize the feedback loop and ensure vehicle safety.

Few studies have addressed the practical need to enable risk assessment and provide safety guarantees for coordinating multiple UGVs in various applications. The contributions of this paper are summarized: 1) a multi-UGV interactive trajectory planning method with flexible and strict safety guarantees is proposed based on vehicle risk assessment and cooperative pathfinding; 2) we conduct simulations in an unstructured environment. The comparative results demonstrate that the proposed method exhibits reduced risk levels and improved adaptability to varying conditions.

II. Multi-UGV Safe Path Planning

This study employs decoupled trajectory planning for multi-UGV, incorporating cooperative path planning, independent trajectory planning, and interactive trajectory planning. In terms of cooperative path planning, vehicles in conflict are determined by field-based risk assessment, and vehicle paths are generated with limited risk hierarchically.

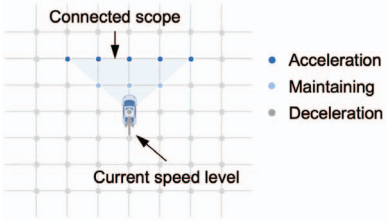

Fig. 1. The connectivity of the vertices in a grid map.

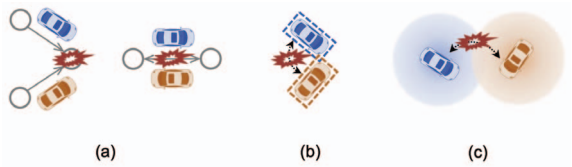

Fig. 2. An illustration of vehicle conflict types. From left to right: (a) the classical node and edge conflicts, (b) the body conflict, (c) the risk conflict.

A. Problem Definition

The problem of multi-UGV safe path planning in this paper is based on the classical MAPF problem with some variations specific to the characteristics of the vehicles. In this problem, the input consists of a tuple $\langle \mathcal{M}, starts, goals \rangle$ where $\mathcal{M} = (\mathcal{V}, \mathcal{E})$ is an undirected graph that represents the map with vertices represented by $\mathcal{V}$ and edges connecting the vertices by $\mathcal{E}$. The mapping, $starts : [1, ..., n] \rightarrow \mathcal{V}$ assigns each vehicle to the corresponding starting position, while $goals : [1, ..., n] \rightarrow \mathcal{V}$ assigns each vehicle to the desired goal position. The assumption of discretized time is made, allowing each vehicle to be at one vertex at each time step and perform a single *action*. An action allows a vehicle to either move from one vertex to a connected vertex or remain at the current vertex. In our assumption, the connectivity between vertices depends on the heading angle of the vehicle, and the execution of actions is limited by the vehicle's speed level, which considers the maneuverability and speed variation constraints of the vehicle. Fig. 1 presents the connectivity of the vertices in a grid map with the neighborhood movement model in [12] and [24].

For each vehicle i, we assign a movement model denoted as π_i that conforms to the connectivity. The next position of the vehicle is dependent on the last and current positions and the action model. Formally, the next position of the vehicle is determined as $(x_{t+1},\ y_{t+1},\ t+1) = \pi_i\left[(x_{t-1}, y_{t-1}, t-1), (x_t, y_t, t)\right]$. Each vehicle i has a specific path denoted by a sequence of positions (x, y, t), which starts from its initial position $starts(i)$ and goes up to the final destination $goals(i)$. The problem is solved by a set of n independent paths, where each path represents the route of a single vehicle. The solution must adhere to safety requirements for each time step t.

B. Vehicle Risk Conflict

The safety requirements of classical MAPF problem demand the execution of independent paths without collisions. MAPF solvers use the concept of *conflict* while planning, where a plan is *valid* only if there is no conflict between any two paths. The classification of conflicts can be various. The majority of theoretical studies in MAPF and its solution techniques assume agents as ideal mass-points and identify node, and edge conflicts, illustrated in Fig. 2(a). The node conflict prohibits two or more agents from occupying the same position simultaneously, and the edge conflict restricts two agents from exchanging positions in consecutive time steps. To meet the collision-free requirements in practical scenarios involving robots or vehicles, the body conflict has been introduced as illustrated in Fig. 2(b). The body conflict considers the geometric shape of the robot or vehicle and restricts any two of them from overlapping.

In this study, we expand upon the current research by introducing a risk assessment component to multi-UGV safe path planning to increase safety margins and improve resistance to path tracking errors. The quantification of risk is conducted through the use of field-based risk assessment methods, thus allowing for the assessment of risks across multiple dimensions. We utilize the repulsive APF model outlined in [15] and express the potential field U_{rep} as

$$U_{\text{rep}}(r) = \begin{cases} \dfrac{k_p}{2}\left(\dfrac{1}{r} - \dfrac{1}{r_{\max}}\right)^2, & 0 < r \le r_{\max} \\ 0, & r > r_{\max} \end{cases}, \tag{1}$$

where the parameter r represents the distance between two vehicles, and a maximal limit of $r_{\max}$ is set to reflect that any distance beyond this limit can be considered negligibly low in terms of risk assessment; k_p is the coefficient.

The repulsive field force is determined by the derivative of the potential field, represented as $F_{\text{rep}} = \nabla_r U_{\text{rep}}$. To account for practical considerations, the field force is bounded by the distance between two vehicles, which is described by

$$F_{\text{rep}}(r) = \begin{cases} F_{\text{rep,max}}, & 0 < r \le r_{\min} \\ -\dfrac{k_p}{r^3}\left(1 - \dfrac{r}{r_{\max}}\right), & r_{\min} < r < r_{\max} \\ 0, & r \ge r_{\max} \end{cases}, \tag{2}$$

where $r_{\min}$ is the smallest permissible distance between two vehicles, below which the repulsive force reaches its maximum $F_{\text{rep,max}} = -k_p/r_{\min}^3(1 - r_{\min}/r_{\max})$ and the negative sign indicates the repulsive direction that causes the vehicles to separate. In practice, zero distance between two vehicles is impossible. The value of $r_{\min}$ is determined based on their geometrical shape.

The *risk conflict* is used to refer to a situation where the risk level between any two vehicles surpasses a predetermined threshold. The normalized repulsive field force is utilized to represent the risk level, and it is defined as

$$\rho(r) = \frac{F_{\text{rep}}(r)}{F_{\text{rep,max}}}, \tag{3}$$

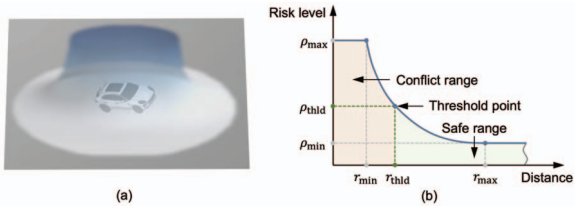

Fig. 3. The field-based risk conflict definition. From left to right: (a) the bounded potential field, (b) the risk level and the risk conflict.

where $\rho \in [0, 1]$ denotes the risk level. To determine the presence of a risk conflict, the risk level is compared to a predetermined threshold ρ_{thld}, as illustrated in Fig. 3. A risk conflict arises if $\rho \ge \rho_{\text{thld}}$. It should be noted that the threshold can range from 0 to 1 and that a lower threshold indicates a more rigorous determination of risk conflict, leading to an increase in safety margins during path planning.

C. Cooperative Path Planning

Based on the approaches of hierarchical cooperative A* and prioritized planning, we conduct cooperative path planning for multi-UGVs considering the risk conflicts. The environment is modeled using a grid map, formulated as

$$\mathcal{M} = \{(x, y, t)_o\}_{W \times H \times T}, \tag{4}$$

where $\mathcal{M}$ denotes the spatio-temporal map, $(x, y, t)_o$ represents an obstacle that occupies (x, y) at time t and is thereby inaccessible for vehicles, where W and H are the width and height of the map, and T is the largest time.

In order to plan cooperative paths, the priorities of the vehicles should be determined firstly. This study uses the Euclidean distance heuristic to prioritize the vehicles. In this method, vehicles with smaller distances are given a higher priority in path planning. Fig. 4 presents a case of cooperative path planning for three vehicles. A* search is used repeatedly and sequentially for each vehicle to plan a safe path, which is represented as a series of (x, y, t) coordinates in

$$\begin{aligned} P_i = &[(x_0, y_0, t_0)_i, (x_1, y_1, t_1)_i, \ldots, \\ &(x_{N_i}, y_{N_i}, t_{N_i})_i, (x_{N_i}, y_{N_i}, t_{N_i+1})_i, \ldots], \end{aligned} \tag{5}$$

where P_i represents the path of vehicle i, $(x_0, y_0)_i$ denotes its starting position $starts(i)$, $(x_{N_i}, y_{N_i})_i$ denotes its goal position $goals(i)$, and the vehicle starts at time step t_0 and remains stationary at the goal from time step N_i onwards. The path must meet the safety requirement of ensuring that there are no risk conflicts between any two vehicles by

$$\rho\left(r_{ij}(k)\right) < \rho_{\text{thld}}, \quad \forall\, 0 \le k \le T, \forall\, j \in \mathcal{P}_i, \tag{6}$$

where $r_{ij}(k)$ denotes the Euclidean distance between the positions $(x_k, y_k)_i$ and $(x_k, y_k)_j$ of vehicles i and j, respectively, $\mathcal{P}_i$ is the set of vehicles which are prior than vehicle i. In the event that the constraint (6) is violated, the map is updated by $\mathcal{M} = \mathcal{M} \cup \{(x_k, y_k, t_k)_i\}$ and the path P_i should be recalculated using A* algorithm with the updated map $\mathcal{M}$. Conversely, if the constraint (6) is met, P_i serves as a fixed safe path for vehicle i, and $\mathcal{M}$ is updated by $\mathcal{M} = \mathcal{M} \cup P_i$.

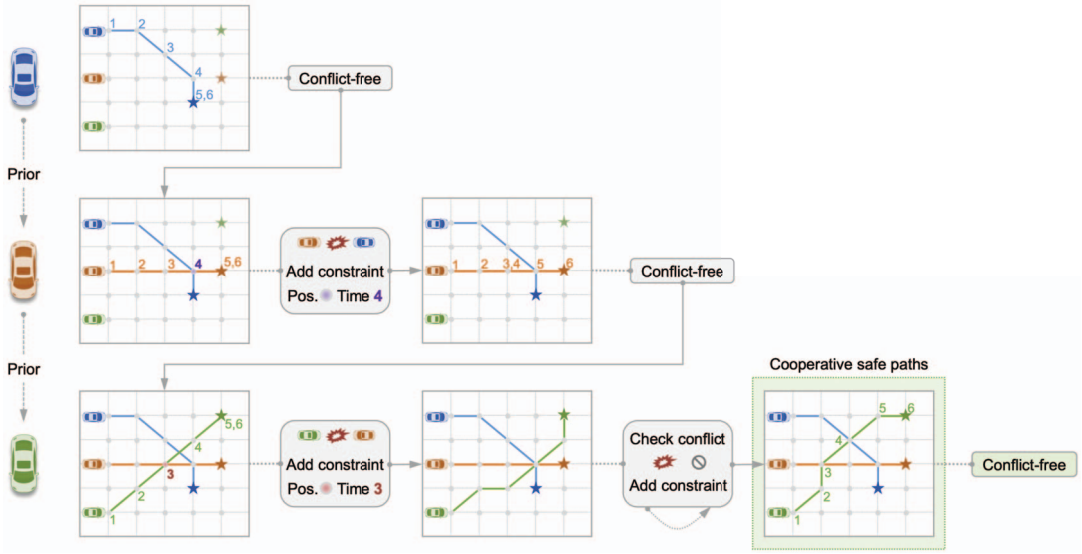

Fig. 4. An example of cooperative path planning process for three vehicles.

Then the updated map is shared with the subsequent vehicle, culminating with the last vehicle.

III. Interactive Trajectory Planning

The cooperative paths presented in Sec. II are derived from an A*-based search in the grid map. However, the path points are sparse, thus not suitable for UGVs to track accurately. This study centers on feasible trajectories that can serve as inputs to vehicle control systems. Interactive trajectory planning, which is grounded on the curved path and speed planning, is introduced in the following section.

A. Vehicle Path Curve

Bézier curves and B-spline curves are commonly employed in computer-aided design and trajectory planning for autonomous vehicles. A continuous curve can be determined by specifying the control points. B-spline curves offer robustness and practicality in modeling the vehicle path and support any number of control points, which enables the efficient generation of single-segment vehicle paths. Therefore, B-spline curves have been adopted for vehicle path curve generation, utilizing the cooperative path points as the control points. Note that B-spline curves may not cross the control points, the resultant safety issues can be eliminated by employing following trajectory planning and validation process.

B. Optimal Speed Planning

The optimal speed profile on the path curve is determined using the vehicle kinematic bicycle model as

$$\dot{\mathbf{z}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v\sin\theta \\ v\cos\theta \\ \frac{v}{L}\tan\delta \\ a \end{bmatrix}, \tag{7}$$

where $\mathbf{z}$ denotes the state of the vehicle, represented by $[x, y, \theta, v]^\top$, including the vehicle's position in global coordinates (x, y), the heading angle θ, and the speed v; a denotes the acceleration, δ denotes the steering angle, and L represents the wheel-base length, which indicates that the vehicle's position is located at the center of the rear axle.

We assume that the vehicle lateral control is perfect. As a result, the vehicle's position is updated based on the acceleration within each control time interval as

$$\begin{aligned} s(k+1) &= s(k) + v(k)\Delta t, \\ v(k+1) &= v(k) + a(k)\Delta t, \end{aligned} \tag{8}$$

where the vehicle's position s is represented by the length of the curve from its starting point $s(0)$; Δt represents the control time interval, which contrasts to the planning time interval ΔT in the grid map. Aiming to minimize the control energy of the accumulated acceleration, the optimization problem for the specific vehicle i can be described as

$$\begin{aligned} \min_{\mathbf{a}} \quad & \mathbf{a}^\top \mathbf{a}, \\ \text{subject to} \quad & (8), \\ & s(N_t k) = S_i(k),\ \forall\, 0 \le k \le N_i, \\ & v(0) = v(N_t N_i) = 0, \end{aligned} \tag{9}$$

where $\mathbf{a}$ represents the acceleration vector that consists of each acceleration value in each control time interval. N_t is an integer representing the number of control time intervals within the planning time interval $(\Delta T/\Delta t)$, which assumes the planning time interval is divisible by the control time interval. $S_i(k)$ represents the total length of the path curve from the starting point to the nearest point of $(x_k, y_k)_i$ on the path curve, according to (5), which ensures compliance of the trajectory with the given path in cooperative pathfinding. Moreover, the vehicle is constrained to be stationary at both its start and goal positions. Note that additional constraints can be incorporated into the optimization problem if necessary.

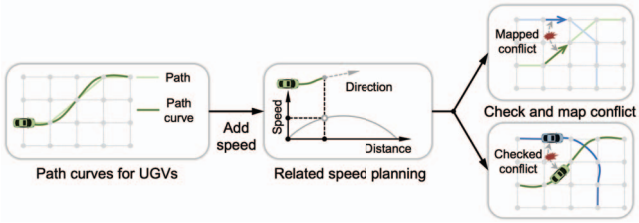

Fig. 5. The trajectory planning and validation process.

C. Trajectory Planning and Validation

Despite cooperative path planning ensuring the safety of cooperative paths on the grid map, the safety of the derivative trajectories cannot be guaranteed. One reason is that the path curves generated by B-splines may not pass through the control points of the cooperative paths. Another reason is that the denser trajectory points have yet to be validated for safety.

To address these concerns, an interactive trajectory planning approach that incorporates path planning is proposed, as shown in Fig. 5. The approach consists of conducting a point-by-point conflict check on the trajectories. If a conflict is detected, it is mapped onto the grid map and used to enforce constraints on the closest grid position and corresponding time step. The planning and validation process iteratively continue until all trajectories are determined to be safe and free of risk conflicts or other types of conflicts. For more details, refer to Algorithm 1. The grid map is updated and shared among the vehicles to expedite the joint planning process.

IV. Simulation Experiment

In this section, we introduce the simulation setup, which encompasses the simulation scenario, the proposed and benchmark algorithms, and the related parameters for the simulation setup. In the latter part, the results of performance comparison based on these algorithms are presented.

A. Simulation Setup

The simulations were conducted in a representative unstructured environment. The setup is outlined as follows.

1) Scenario: The Dragon Age Origin (DAO) maps are utilized for simulations, and they are well-known benchmark scenarios in the MAPF research field [12]. We have chosen *Arena*, one of the DAO maps, as shown in Fig. 6. The map information is publicly accessible and can be converted into a grid map with boundary and obstacle grids formulated as (4).

2) Algorithms: In addition to our proposed interactive trajectory planning with risk conflict resolution (ITP-RC) algorithm as explained in Algorithm 1, we also implement two benchmark algorithms: CBS-BC, a recent car-like robot planning algorithm based on CBS with body conflict resolution [11], and ITP-BC, the interactive trajectory planning algorithm with body conflict resolution. While the CBS algorithm is proven to be optimal for path planning, our ITP algorithm is sub-optimal. We will compare ITP-RC with ITP-BC to demonstrate the effectiveness of incorporating risk assessment.

Algorithm 1: Interactive trajectory planning.

Input: Grid map $\mathcal{M}_{W\times H\times T}$, start and goal positions of UGVs $starts$ and $goals$, risk threshold ρ_{thld}
Output: Safe trajectories of UGVs $T_1, \ldots, T_n$
Calculate vehicle priorities $\mathcal{O}_p \leftarrow [\text{ord}_1, \ldots, \text{ord}_n]$ as well as sets of prior vehicles $\mathcal{P}_1, \ldots, \mathcal{P}_n$;
for $i \in \mathcal{O}_p$ **do**
  Initialize $pass \leftarrow$ False;
  while *not pass* **do**
    Plan path P_i on $\mathcal{M}$ by A* search;
    for $j \in \mathcal{P}_i$ **do**
      Check conflicts on paths;
      if *conflicts exist* **then**
        Update $\mathcal{M}$ by the conflicts;
      end
    end
    if *no conflict* **then**
      Raise $pass \leftarrow$ True;
      Plan trajectory T_i based on P_i;
      for $j \in \mathcal{P}_i$ **do**
        Check conflict on trajectories;
        if *conflicts exist* **then**
          Raise $pass \leftarrow$ False;
          Update $\mathcal{M}$ by the mapped conflicts;
        end
      end
    end
  end
  Update $\mathcal{M}$ by P_i;
end

TABLE I
Setup of Simulation Parameters

Parameter	Value
A* speed level	$[0, 1, 2]$
Arena map size	$49\text{m} \times 49\text{m}$
Map grid size	$1\text{m} \times 1\text{m}$
UGV size	$1.5\text{m} \times 2.5\text{m}$
Risk field range	$[1.5\text{m}, 5\text{m}]$
Planning time interval	1s
Control time interval	0.2s
Runtime limit	180s
UGV number	$[3, 4, 5, 6, 7, 8, 9]$
Risk threshold	$[0.02, 0.04, 0.08, 0.16, 0.32, 0.64]$
Random case number	100

Additionally, we will compare ITP-RC with CBS-BC to examine the impact of optimality on application performance.

3) Parameters: The parameter values for conducting simulations on this map are listed in Table I. The speed level of A* search is comparable to the 2^k neighborhood movement model described in [12], where k represents the speed level. Additionally, the change in the vehicle's heading angle between two consecutive time steps will be limited to $\pi/4$ due to the vehicle's characteristics. The range of the risk field is defined by $r_{\min}$ and $r_{\max}$, as shown in (2).

The simulations were conducted using Python 3.9.16 on a laptop powered by the Apple M1 chip.

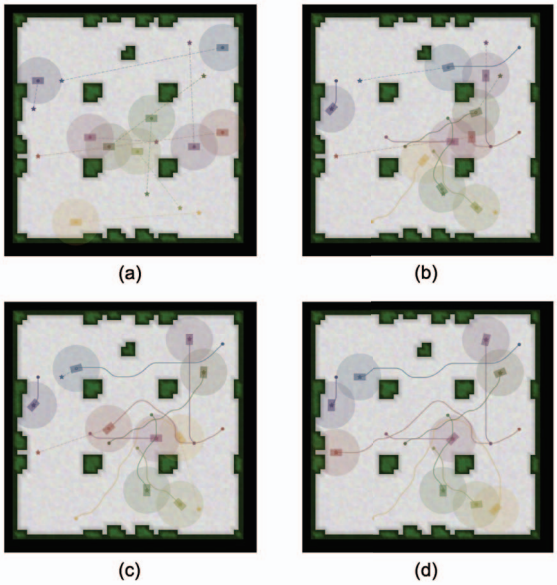

Fig. 6. An ITP-RC simulation case (UGV number of 9, risk threshold of 0.02) involving vehicles tracking their trajectories. Sequentially: (a) starting at 0s, (b) tracking at 8s, (c) tracking at 16s, (d) stopping at 24s.

B. Comparative Trajectory Planning for Multi-UGVs

The safety and efficiency performance of the proposed ITP-RC is compared with the benchmarks ITP-BC and CBS-BC through numerous simulations. The evaluation metrics are presented as the foundation, followed by a planning case and statistical results.

1) Evaluation Metrics: The efficiency performance is evaluated based on the *success rate* and *makespan*, while the safety performance is evaluated using the *max. risk level* and *max. time-to-collision inverse*.

- The *success rate* α, defined as $\alpha = \mathcal{N}_{\text{succ}}/\mathcal{N}_{\text{tot}}$, represents the proportion of successful cases among all simulated cases. A case is deemed successful if the cooperative trajectories can be solved within the runtime limit, as indicated in Table I.
- The *makespan* T_{ms} in a case, defined as $T_{\text{ms}} = \max_i N_i t_{\text{plan}}$, where t_{plan} is the planning time interval, represents the time at which the last vehicle reaches its goal in accordance with the cooperative trajectories.
- The *max. risk level* $\rho_{\max}$ in a case, is defined as $\rho_{\max} = \max_i \rho_{i,\max}$, where $\rho_{i,\max}$ denotes the maximal risk level of vehicle i for all the time steps.
- The *max. time-to-collision inverse* $TTCi_{\max}$ in a case, is defined as $TTCi_{\max} = \max_i TTCi_{i,\max}$, where $TTCi_{i,\max}$ denotes the maximal $TTCi$ between vehicle i and any other vehicle for all the time steps.

2) Planning Case: We present a cooperative trajectory planning case utilizing our ITP-RC algorithm. In this case, 9 UGVs move from their starting positions to their respective goals while maintaining a maximal risk level of 0.02. The details of this case are illustrated in Fig. 6. The vehicles are represented by different colors, with dashed lines connecting the vehicles to their goals, respectively. The solid line illustrates their past trajectory, and the circle area indicates the risk field range. As various control methods exist and are not the primary focus of this study, we assume that the vehicles perfectly follow their trajectories. It is evident that the vehicles are coordinated safely throughout the process, and the trajectories are well-suited for the vehicles to follow.

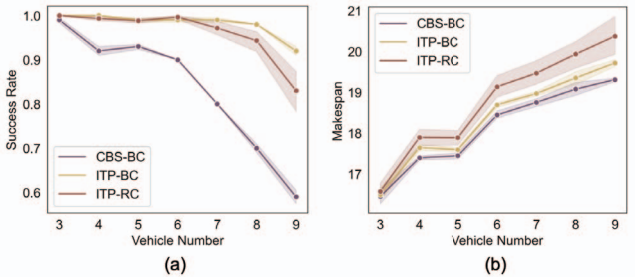

Fig. 7. The efficiency performance of the comparative algorithms. From left to right: (a) the success rate, (b) the makespan.

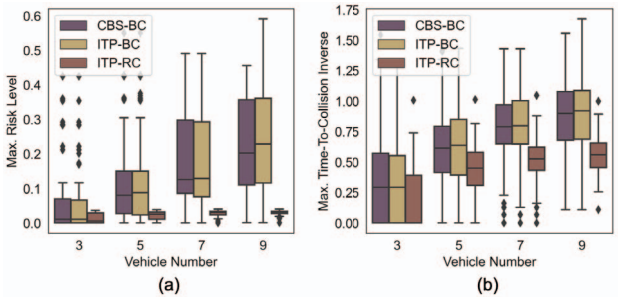

Fig. 8. The safety performance of the comparative algorithms with risk threshold of 0.04. From left to right: (a) the max. risk level, (b) the max. time-to-collision inverse.

3) Statistical Results: A number of simulations were conducted to compare the performance of the algorithms. The variables mainly consist of the number of vehicles and the risk threshold. The objective is to examine the correlation between the efficiency and safety performance and the variables, in terms of the comparative algorithms. The simulations are statistically initialized and conducted for each random seed. Afterwards, the average or limiting values of the evaluation metrics are calculated and analyzed.

Fig. 7(a) demonstrates a decrease in success rate as the number of vehicles increases for all the algorithms. CBS-BC has the lowest makespan cost, as seen in Fig. 7(b), which indicates the optimality of the CBS. However, due to consuming more time in node expansion and tree search, its success rate is lower. ITP-RC has a slightly lower success rate compared to ITP-BC. This is due to a wider range of the risk conflict than the body conflict, leading to a higher probability of conflicts and longer cooperative planning time. With an increase in the number of vehicles, the makespan is expected to increase naturally, but the presence of risk conflicts amplifies this increase.

Concerning safety performance, Fig. 8(a) exhibits that the proposed ITP-RC algorithm inherently ensures that the cooperative trajectories remain below the predefined risk threshold. Conversely, the body conflict based algorithms exhibit greater

TABLE II
THE MAX. RISK LEVEL REDUCTION OF ITP-RC OVER CBS-BC.

Risk threshold	4 UGVs	6 UGVs	8 UGVs
0.02	95.3%	95.9%	96.2%
0.08	81.4%	83.8%	84.2%
0.32	33.2%	37.4%	39.0%

and unpredictable risk levels. Fig. 8(b) also illustrates that there is a risk reduction in the cooperative trajectories when taking the risk conflict in account using the classical time-to-collision inverse metric. An adjustment in risk reduction is possible through modifying the pre-set risk threshold, as illustrated in Table II, which can result in different degrees of risk reduction. This indicates that the ITP-RC algorithm offers a strict yet adaptable safety margin for cooperative trajectory planning of multi-UGVs.

V. CONCLUSION

This paper presents a novel approach for multi-UGV cooperative trajectory planning with flexible safety guarantees. The proposed method addresses the challenge of finding safe trajectories for multiple UGVs by incorporating risk assessment into the planning process. The integration of path planning, trajectory planning and validation, and optimal speed planning improves the quality of the generated trajectories, thereby facilitating the direct application to vehicle controllers.

The results of simulation experiments conducted against benchmark algorithms indicate that the proposed interactive trajectory planning algorithm for 9 UGVs in the Arena map achieves a success rate of over 80% within a solving time of 180 seconds, while ensuring that the maximal risk level remains below a flexible predefined threshold. The algorithm surpasses the benchmark algorithms in terms of safety performance, exhibiting reduced risk levels and improved adaptability to varying conditions. This comprehensive approach enables UGVs to navigate complex environments while ensuring safety and cooperative behavior simultaneously.

REFERENCES

[1] M. Cai, Q. Xu, C. Chen, J. Wang, K. Li, J. Wang, and Q. Zhu, "Formation control for connected and automated vehicles on multi-lane roads: Relative motion planning and conflict resolution," *IET Intelligent Transport Systems*, vol. 17, no. 1, pp. 211–226, 2023.

[2] C. Chen, M. Cai, J. Wang, K. Li, Q. Xu, J. Wang, and K. Li, "Cooperation method of connected and automated vehicles at unsignalized intersections: Lane changing and arrival scheduling," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 11, pp. 11351–11366, 2022.

[3] M. De Petrillo, J. Beard, Y. Gu, and J. N. Gross, "Search planning of a uav/ugv team with localization uncertainty in a subterranean environment," *IEEE Aerospace and Electronic Systems Magazine*, vol. 36, no. 6, pp. 6–16, 2021.

[4] M. Gao, L. Jin, Y. Jiang, and B. Guo, "Manifold siamese network: A novel visual tracking convnet for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1612–1623, 2019.

[5] Q. Xu, Y. Liu, J. Pan, J. Wang, J. Wang, and K. Li, "Reachability analysis plus satisfiability modulo theories: An adversary-proof control method for connected and autonomous vehicles," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 3, pp. 2982–2992, 2022.

[6] D. Silver, "Cooperative pathfinding," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 1, pp. 117–122, 2005.

[7] Z. Bnaya and A. Felner, "Conflict-oriented windowed hierarchical cooperative a," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3743–3748, IEEE, 2014.

[8] T. Standley, "Finding optimal solutions to cooperative pathfinding problems," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 24, pp. 173–178, 2010.

[9] M. Goldenberg, A. Felner, N. Sturtevant, R. C. Holte, and J. Schaeffer, "Optimal-generation variants of epea," in *Proceedings of the International Symposium on Combinatorial Search*, vol. 4, pp. 89–97, 2013.

[10] G. Sharon, R. Stern, A. Felner, and N. R. Sturtevant, "Conflict-based search for optimal multi-agent pathfinding," *Artificial Intelligence*, vol. 219, pp. 40–66, 2015.

[11] L. Wen, Y. Liu, and H. Li, "Cl-mapf: Multi-agent path finding for car-like robots with kinematic and spatiotemporal constraints," *Robotics and Autonomous Systems*, vol. 150, p. 103997, 2022.

[12] 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, pp. 151–158, 2019.

[13] J. Li, P. Surynek, A. Felner, H. Ma, T. S. Kumar, and S. Koenig, "Multi-agent path finding for large agents," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 7627–7634, 2019.

[14] M. Gao, L. Jin, Y. Jiang, and J. Bie, "Multiple object tracking using a dual-attention network for autonomous driving," *IET Intelligent Transport Systems*, vol. 14, no. 8, pp. 842–848, 2020.

[15] F. Bounini, D. Gingras, H. Pollart, and D. Gruyer, "Modified artificial potential field method for online path planning applications," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 180–185, IEEE, 2017.

[16] J. Wang, J. Wu, and Y. Li, "The driving safety field based on driver–vehicle–road interactions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2203–2214, 2015.

[17] Y. Liu, J. Wang, C. Chen, Q. Xu, and L. Li, "Vehicle safety analysis at non-signalised intersections at different penetration rates of collision warning systems," *IET Intelligent Transport Systems*, vol. 14, no. 13, pp. 1759–1768, 2020.

[18] H. Huang, J. Liu, Y. Yang, and J. Wang, "Risk generation and identification of driver–vehicle–road microtraffic system," *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, vol. 8, no. 3, p. 04022029, 2022.

[19] J. Chen, J. Li, C. Fan, and B. C. Williams, "Scalable and safe multi-agent motion planning with nonlinear dynamics and bounded disturbances," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 11237–11245, 2021.

[20] D. Ioan, I. Prodan, S. Olaru, F. Stoican, and S.-I. Niculescu, "Mixed-integer programming in motion planning," *Annual Reviews in Control*, vol. 51, pp. 65–87, 2021.

[21] J. Eilbrecht and O. Stursberg, "Optimization-based maneuver automata for cooperative trajectory planning of autonomous vehicles," in *2018 European Control Conference (ECC)*, pp. 82–88, IEEE, 2018.

[22] J. Alonso-Mora, P. Beardsley, and R. Siegwart, "Cooperative collision avoidance for nonholonomic robots," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 404–420, 2018.

[23] P. Scheffe, M. V. Pedrosa, K. Flaßkamp, and B. Alrifaee, "Receding horizon control using graph search for multi-agent trajectory planning," *IEEE Transactions on Control Systems Technology*, 2022.

[24] Z. Boroujeni, D. Goehring, F. Ulbrich, D. Neumann, and R. Rojas, "Flexible unit a-star trajectory planning for autonomous vehicles on structured road maps," in *2017 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, pp. 7–12, IEEE, 2017.