Letter

Early-Awareness Collision Avoidance in Optimal Multi-Agent Path Planning With Temporal Logic Specifications

Yiwei Zheng, Aiwen Lai, Xiao Yu, *Member, IEEE*, and Weiyao Lan, *Senior Member, IEEE*

Dear Editor,

This letter investigates a multi-agent path planning problem in a road network with the requirement of avoiding collisions among all agents in the partitioned environment. We first abstract the agents to a set of transition systems, and construct a team transition system from these individual systems. A mechanism is designed for the team transition system to detect all collisions within the synthesized run. Then, "wait" and "go-back" transitions are added to the individual transition systems, while removing all possible collisions in the team transition system. Finally, two examples are given to illustrate the effectiveness of the proposed approach.

Path planning is a classic problem aiming to synthesize feasible trajectories for unmanned ground mobile robots [1], aerial vehicles [2] and surface vessels [3] to navigate from one place to the other [4]. Formal method is an accurate and user-friendly way to solve path planning problem of MASs in an intuitive but mathematically precise manner [5], [6]. Recently, path planning satisfying particular mission specifications has been widely studied, while optimizing a certain performance index, such as time, efficiency [7], correctness [8], or some other relevant metric [9]. Although these methods are effective in finding trajectories satisfying the temporal logic specifications, yet collision avoidance is not considered in these methods.

Considering the MASs in practical situations, one single collision could lead to the failure of the whole system. Thus, the synthesized trajectories need to avoid all collisions among agents and all static obstacles along the way. Various methods have been proposed for finding collision-free paths. In [10], an overall solution was presented to solve both inter-agent collisions and deadlocks with low-speed obstacles. In [11], the possible motion conflict between robots is resolved by reinforcement learning. Although the aforementioned methods appear effective in collision avoidance, yet the essence of these methods is still to react to incoming collisions from which the performance of synthesized trajectories may suffer.

This letter is motivated to develop a multi-agent path planning method which guarantees synthesized runs collision-free while maximizing efficiency. The main contributions of this letter can be summarized as follows. First, a novel offline planning method is put forward based on formal method. Compared with the algorithm in [12] where the requirement of collision avoidance was not considered, our proposed algorithm can synthesize the optimal individual runs for MASs which are guaranteed collision-free among all agents in the partitioned environments. Second, the conflict-based searching approaches [13] and the variants of safe interval path planning [14] were adopted for finding the optimal collision-free path to a designated position in a dynamic environment, but the application to linear temporal logic (LTL) specifications was not considered, while our proposed method is able to synthesize a set of optimal trajectories under LTL specifications for MASs.

Corresponding author: Xiao Yu.

Citation: Y. W. Zheng, A. W. Lai, X. Yu, and W. Y. Lan, "Early-awareness collision avoidance in optimal multi-agent path planning with temporal logic specifications," *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 5, pp. 1346–1348, May 2023.

Y. W. Zheng, A. W. Lai, and W. Y. Lan are with the Department of Automation, Xiamen University, Xiamen 361005, and also with the Key Laboratory of Control and Navigation (Xiamen University), Fujian Province University, Xiamen 361005, China (e-mail: ywzheng1201@stu.xmu.edu.cn; aiwenlai@xmu.edu.cn; wylan@xmu.edu.cn).

X. Yu is with the Department of Automation, Xiamen University, Xiamen 361005, and also with the Innovation Laboratory for Sciences and Technologies of Energy Materials of Fujian Province (IKKEM), Xiamen 361005, China (e-mail: xiaoyu@xmu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2022.106043

Problem formulation: A team of m controllable agents are deployed in a road network with intersections. The behavior of these agents can be modeled as weighted transition systems (TSs) $\{\mathbf{T}_1,\ldots,\mathbf{T}_m\}$, and TS $\mathbf{T}_i$ is a tuple

$$\mathbf{T}_i := (Q_i, q_i^0, \delta_i, \Pi_i, L_i, w_i)$$

where Q_i is a finite set of states, $q_{i,0}\in Q_i$ is the initial state, $\delta_i \subseteq Q_i\times Q_i$ is the transition relation, Π_i is a finite set of atomic propositions (APs), $L_i: Q_i\to 2^{\Pi_i}$ is a map giving the set of APs satisfied in a state, $w_i:\delta_i\to\mathbb{N}^+$ is a map assigning a positive integer weight to each transition. For agent $\mathbf{T}_i$, the transition relation δ_i is abstracted from the road network, the set of states Q_i are abstracted from the intersections, and the weight w_i corresponds to the end-to-end traveling times. The run of the agent $\mathbf{T}_i$ is an infinite sequence $r_i = q_{i,0},\ q_{i,1},\ \ldots,\ q_{i,k},\ \ldots,\ k\in\mathbb{N}$. The trace of a run is an infinite sequence $\mathcal{L}_i(r_i) = L_i(q_{i,0}),\ L_i(q_{i,1}),\ \ldots, L_i(q_{i,k}),\ \ldots,$ with $k\in\mathbb{N}$, and $L_i(q_{i,k})\subseteq 2^{\Pi_i}$. With all m controllable agents deployed in the same road network, if we denote the set of APs of the road work as Π, then we have $\Pi_i\subseteq\Pi$ for agent $\mathbf{T}_i$.

The high-level mission specifications of agents $\{\mathbf{T}_1,\ \ldots,\ \mathbf{T}_m\}$ are described by LTL formulas. An LTL formula consisting of a set of APs Π, boolean operators and temporal operators, is formed according to the following syntax:

$$\phi := \text{TRUE} \mid \alpha \mid \phi_1\wedge\phi_2 \mid \neg\phi \mid \bigcirc\phi \mid \phi_1\,\mathcal{U}\,\phi_2$$

where $\alpha\in\Pi$ is an AP, and temporal operators $\bigcirc$ and $\mathcal{U}$ mean "next" and "until", respectively. The above definitions also induce temporal operators such as $\vee$ (conjunction), $\diamond$ (eventually), $\square$ (always), and $\rightarrow$ (implication), in which $\phi_1\vee\phi_2 := \neg(\neg\phi_1\wedge\neg\phi_2)$, $\diamond\phi := \text{TRUE}\,\mathcal{U}\,\phi$, $\square\phi := \neg\diamond\neg\phi$, and $\phi_1\rightarrow\phi_2 = \neg\phi_1\,\mathcal{U}\,\phi_2$. Given a run r_i of TS $\mathbf{T}_i$, we say r_i satisfies an LTL formula ϕ if the trace $\mathcal{L}_i(r_i)$ satisfies ϕ.

In this letter, it is assumed that there exist n uncontrollable agents moving along the fixed trajectories in this road network. Collisions between any two of the agents may lead to the asynchronization of the system, or even damage to the agent or the cargo. We first give the definitions of collisions and then the problem statement as follows.

Definition 1 (Collision): A collision happens when any two of the agents are at the same intersection of the road network, or in the same road and move toward each other.

Problem 1: Given a road network with intersections, a LTL formula $\phi := \varphi\wedge(\square\diamond\pi)$ over the atomic proposition set Π, a team of m controllable agents modeled as TSs $\{\mathbf{T}_1,\ldots,\ \mathbf{T}_m\}$, and a group of n uncontrollable agents moving along a set of fixed trajectories, if all of the end-to-end traveling time among $m+n$ agents are identical, synthesize individual runs for the m controllable agents such that the cost function

$$J(\mathbb{T}^\pi) = \limsup_{k\to+\infty}(\mathbb{T}^\pi(k+1)-\mathbb{T}^\pi(k)) \tag{1}$$

is minimized and collisions among all $m+n$ agents are avoided.

In Problem 1, for the LTL formula $\phi := \varphi\wedge(\square\diamond\pi)$, the proposition π must be satisfied infinitely many times, and the maximum time between successive satisfactions of π must be minimized subject to (1). The assumption that all $m+n$ agents shares identical end-to-end traveling time means that for all $q_k,\ q_l\in Q_i\cap Q_j,\ i,\ j,\ k,\ l\in\mathbb{N}^+$, $1\le i,\ j\le m+n$, $w_i((q_k,q_l)) = w_j((q_k,q_l))$. With the trajectories of the uncontrollable agents fixed, we use TSs $\mathbf{T}_j := (Q_j, q_j^0, \delta_j, \Pi_j, L_j, w_j)$ to model these uncontrollable agents, where $m+1\le j\le m+n,\ j\in\mathbb{N}$, $Q_j\subseteq\cup_{k=1}^m Q_k,\ \delta_j\subseteq\cup_{k=1}^m\delta_k,\ \Pi_j=\emptyset$, and $\forall q_j\in Q_j$, and the number successor states $\|Post(q_j)\| = 1$. With a slight abuse of notations, we use $r(k,i)$ to denote the ith element in the kth state of the team run r, i.e., for $r = (q_{1,0},\ldots,q_{i,0},\ldots),\ \ldots,(q_{1,k},\ldots,q_{i,k},\ldots),\ \ldots,\ \ r(k,i) = q_{i,k}$. Then, according to Definition 1, we categorize the defined collisions into two groups:

1) Singleton collision: Agents $\mathbf{T}_i$ and $\mathbf{T}_j$ move toward the same vertices and arrive simultaneously, i.e.,

$$r(k,i)=r(k,j), r(k,i)\in Q_i, r(k,j)\in Q_j, i,j,k\in\mathbb{N}. \tag{2}$$

2) Pairwise collision: Agents $\mathbf{T}_i$ and $\mathbf{T}_j$ are in the same transition and move towards each other, which is categorized into the following situations:

$$(r(k,i)=q_1\rightarrow r(k+n,i)=q_2)\wedge(r(k,j)=q_2\rightarrow r(k+n,j)=q_1) \tag{3}$$

$$(r(k,i)=(q_1,q_2,n)\wedge r(k,j)=q_2)\rightarrow r(k+n,j)=q_1 \tag{4}$$

or

$$r(k-n,j)=q_2\rightarrow(r(k,i)=(q_1,q_2,n)\wedge r(k,j)=q_1) \tag{5}$$

where $q_1,q_2\in Q_i\cap Q_j$ are arbitrary non-traveling states, (q_1,q_2,n) is a traveling state, and $r(k-n,j)$ and $r(k+n,j)$, $n\in\mathbb{N}$ are for the successive non-traveling states of $r(k,j)$. Now, with all of the $m+n$ agents modeled as TSs $\{\mathbf{T}_1,\ldots,\mathbf{T}_m,\mathbf{T}_{m+1},\ldots,\mathbf{T}_{m+n}\}$ and the LTL formula $\phi:=\varphi\wedge(\square\diamond\pi)$, we can construct the team TS $\mathbf{T}$ that captures the joint synchronous motions of all agents in the environment, find the optimal team run $r^*_{\mathbf{T}}$ minimizing the cost function (1) by the OPTIMAL-RUN algorithm, and project $r^*_{\mathbf{T}}$ to individual TSs for optimal individual runs $\{r^*_1,\ldots,r^*_m\}$. More details on the traveling states of the team TS and the OPTIMAL-RUN algorithm can be found in [12] and [7], respectively. Although the optimality and correctness of the mission specifications can be guaranteed in the aforementioned algorithm, yet collisions may happen between any two of the agents in the run.

To synthesize optimal collision-free runs for agents $\{\mathbf{T}_1,\ldots,\mathbf{T}_{m+n}\}$, we find collisions within the team run $r^*_{\mathbf{T}}$ according to (2)–(5). Then, we introduce "wait" and "go-back" transitions for controllable agents as follows to avoid singleton and pairwise collisions, respectively.

Definition 2: A "wait" introduced to TS $\mathbf{T}_i$ at state $q\in Q_i$ is a transition originating and ending at state q and denoted as (q,q). The cost of the "wait" transition is equivalent to the minimum cost of w_i, i.e., $w_i((q,q))=\min(w_i(q_k,q_l))$, $\forall q_k,\ q_l\in Q_i,\ k\neq l$.

Definition 3: A "go-back" introduced to TS $\mathbf{T}_i$ at state $q\in Q_i$ is a set of two transitions (q,q') and (q',q), and the state $q'\in Q_i$ is chosen according to the following rules:

1) $(q,q')\in\delta_i$, $(q',q)\notin\delta_i$: $w_i((q,q'))=\min(\{w_i((q,q_k))\}\cup\{w_i((q_k,q))\})$;

2) $(q,q')\notin\delta_i$, $(q',q)\in\delta_i$: $w_i((q',q))=\min(\{w_i((q,q_k))\}\cup\{w_i((q_k,q))\})$;

3) $(q,q')\in\delta_i$, $(q',q)\in\delta_i$: $w_i((q',q))+w_i((q,q'))=\min(\{w_i((q,q_k))+w_i((q_k,q))\})$;

for all $q_k\in Q_i$. The undefined transitions (q,q') or (q',q) will be added to TS $\mathbf{T}_i$, and the cost of the undefined transitions in a "go-back" is given as

$$w_i((q',q))=w_i((q,q'))+g,\quad \text{if } (q,q')\in\delta_i,\ (q',q)\notin\delta_i$$
$$w_i((q,q'))=w_i((q',q))+g,\quad \text{if } (q,q')\notin\delta_i,\ (q',q)\in\delta_i$$

where $g\in\mathbb{N}^+$ is the go-back additional cost.

Therefore, the requirement of collision avoidance can be satisfied by a scheme of adding "wait" and "go-back" transitions to the controllable TSs within $\{\mathbf{T}_1,\ldots,\mathbf{T}_{m+n}\}$.

Main results: In this section, we propose the algorithm which resolves the problem of multi-agent path planning with collision avoidance. The main idea of our approach is to use the optimal collision-free subpaths to replace the subpaths with collisions, and use the "wait" and "go-back" transitions to guarantee the existence of collision-free subpaths. Based on this idea, the following Lemma 1 is given. With a slight abuse of notations, we use $q_k\in r_i$ to denote state q_k is in the run r_i.

Lemma 1: Given a TS $\mathbf{T}_i$, the optimal path $r^*=q_i\,q_{i+1},\ldots,q_j$ from state q_i to q_j which minimizes the cost function $\Sigma_{k=i}^{j-i}w_i((q_k,q_{k+1}))$, and the states $q_{c,1},\ldots,q_{c,n}\in r^*$, $n\in\mathbb{N}^+$ where collisions happen, if for each state before and after $q_{c,n}$, $n\in\mathbb{N}$ in r^*, there exist at least one adjacent collision-free state in TS $\mathbf{T}_i$, then the optimal collision-free path $r^{*\prime}$ can be constructed by bypassing $q_{c,1},\ldots,q_{c,n}$ with collision-free subpaths.

According to Lemma 1, the optimal collision-free run of the team TS $\mathbf{T}$ exists if there are collision-free subpaths which bypass the collisions in the original optimal run. Then, We propose the following Lemma 2 to ensure that those collision-free subpaths exist.

Lemma 2: Given a set of individual TSs $\{\mathbf{T}_1,\ldots,\mathbf{T}_n\}$, $n\in\mathbb{N}^+$ and the corresponding team TS $\mathbf{T}$, assume that all the states in the individual TSs can be repeatedly arrived. If a "wait" transition $(q_{i,j},q_{i,j})$ is added to the individual TS $\mathbf{T}_i$, then there are a set of states $Q^w_{i,j}=\{(q_1,\ldots,q_{i,j},\ldots,q_n)|\exists q_n\in Q_n, n\in\mathbb{N}^+\backslash i\}$ and the corresponding transitions introduced to the team TS $\mathbf{T}$.

According to Lemma 2, we use the collision-free states $\{(q_1,\ldots,q_{i,j},\ldots,q_n)|\exists q_n\in Q_n, n\in\mathbb{N}^+\backslash i: q_n\notin\{q_1,\ldots,q_{n-1}\}\cup\{q_{i,j}\}\}\subsetneq Q^w_{i,j}$ as the origin of the collision-free subpaths. Now, we are ready to state the main result of this letter. and present Algorithm 1 that synthesizes the optimal collision-free individual runs. The result of Algorithm 1 is stated in Theorem 1, and the computational complexity is stated in Remark 1.

Theorem 1: Given a team of $m+n$ controllable TSs $\{\mathbf{T}_1,\ldots,\mathbf{T}_{m+n}\}$, $m,n\in\mathbb{N}^+$, and an LTL formula $\phi:=\varphi\wedge(\square\diamond\pi)$, the set of optimal collision-free run $\{r^*_1,\ldots,r^*_m\}$ can be synthesized for the controllable TSs $\{\mathbf{T}_1,\ldots,\mathbf{T}_m\}$ by Algorithm 1.

Remark 1: Given an LTL formula ϕ, a team of m TSs with the number of Q states and Δ transitions, and another n uncontrollable agents, assume that for all $m+n$ agents, $w_i((q_k,q_l))=w_j((q_k,q_l))$ holds $\forall i,\ j,\ k,\ l\in\mathbb{N}^+$, $1\le i,\ j\le m+n$, $\forall q_k,\ q_l\in Q_i\cap Q_j$, the length of the LTL specification is $|\phi|$, and the largest transition cost is $\omega_{\max}$. The worst-case complexity of the algorithm in [12] is $O((Q^{m+n}+\Delta^{m+n}\omega_{\max})^3\cdot 2^{O(|\phi|)})$ and the worst-case complexity of Algorithm 1 is $O((Q^{m+n}+\Delta^{m+n}\omega_{\max})^3\cdot 2^{O(|\phi|)}+(Q^{m+n}+(2\Delta+Q)^m\Delta^n(\omega_{\max}+g))^3\cdot 2^{O(|\phi|)})$. Even though inter-agent collision avoidance is additionally achieved, the complexity of our algorithm still remains at the same level as the algorithm in [12].

Examples: We demonstrate the proposed algorithm by two examples. First, a team of two Amigobots are deployed in a road network illustrated in Fig. 1 for a persistent surveillance mission. We denote the two controllable Amigobots as $\mathbf{T}_1$ and $\mathbf{T}_2$, respectively. For TSs $\mathbf{T}_1$ and $\mathbf{T}_2$, the initial states $q_{1,0}=U1$, and $q_{2,0}=U2$, and the LTL formula is given as $\phi=\square(gather\rightarrow(a1gather\wedge a2gather))\wedge\square(a1gather\rightarrow\bigcirc(\neg\, a1gather\,\mathcal{U}\,a1upload)\wedge\ \square(a2gather\ \rightarrow\bigcirc(\neg\, a2gather\,\mathcal{U}\,a2upload)\wedge\square\diamond\pi$, and $\pi=a1gather\wedge a2gather$, which means TSs $\mathbf{T}_1$ and $\mathbf{T}_2$ must synchronously gather data infinitely often, and must upload data before gathering data again.

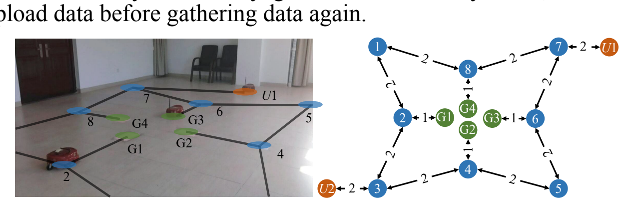

Fig. 1. The physical layout and abstracted TS of the road network.

In this road network, there exists another uncontrollable autonomous Amigobot moving along the fixed trajectory $r^*_3=(3\,4\,5\,6\,7\,8\,1\,2)^\omega$, where ω means that the sequence will be repeated infinitely many times. We construct TS $\mathbf{T}_3$ based on the run r^*_3. Then, we use TSs $\{\mathbf{T}_1,\mathbf{T}_2,\mathbf{T}_3\}$ and the LTL formula ϕ as input of Algorithm 1, and the synthesized optimal runs are illustrated in Fig. 2, which are verified correct and collision-free. Besides, we use the MULTI-ROBOT-OPTIMAL-RUN algorithm in [12] for comparison, in which the

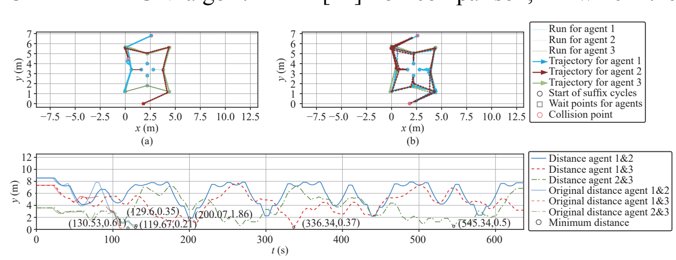

Fig. 2. Experimental results: (a) and (b) illustrate the trajectories for robots in MULTI-ROBOT-OPTIMAL-RUN and our algorithm, respectively, and (c) shows relative distances between robots of the two algorithms.

number of singleton and pairwise collisions are 114 and 6, respectively. Next, we use the synthesized runs for a physical experiment. As is illustrated in Fig. 2, since the size of Amigobot is about $0.28\ \mathrm{m}\times 0.31\ \mathrm{m}$, it is obvious that the trajectories synthesized by our proposed algorithm are collision-free, while collision happens at 119.67 s for the trajectories of the MULTI-ROBOT-OPTIMAL-RUN algorithm. The experimental result shows the Amigobots can successfully complete the high-level mission specifications ϕ while avoiding all collisions without proximate sensing.

Algorithm 1 Early-Awareness Collision Avoidance Multi-Agent Optimal Run

Input: Individual TSs $\{\mathbf{T}_1,\ldots,\mathbf{T}_m,\ldots,\mathbf{T}_{m+n}\}$; LTL formula ϕ;
Output: Optimal individual runs $\{r_1^*,\ldots,r_m^*\}$; Optimal team run $r_{\mathbf{T}}^*$;
Construct team TS $\mathbf{T}$ with $\{\mathbf{T}_1,\ldots,\mathbf{T}_m,\ldots,\mathbf{T}_{m+n}\}$;
Find $r_{\mathbf{T}}^*$ using OPTIMAL-RUN;
if no collisions in $r_{\mathbf{T}}^*$ **then**
  foreach agent $\mathbf{T}_i$, $1\le i\le m$ **do**
    Project $r_{\mathbf{T}}^*$ onto $\mathbf{T}_i$ to obtain $\{r_1^*,\ldots,r_m^*\}$;
else
  Collect singleton collisions $\{(q_{1,k},\ldots,q_{S,i,k},q_{S,j,k},\ldots)|q_{S,i,k}=q_{S,j,k}\}$ in $r_{\mathbf{T}}^*$;
  Collect pairwise collisions $\{((q_{1,m},\ldots,q_{P,i,m},q_{P,j,m},\ldots),\ (q_{1,n},\ldots,q_{P,i,l},q_{P,j,l},\ldots))\}$ according to (3)–(5);
  foreach controllable TS $\mathbf{T}_i$, $1\le i\le m$ **do**
    foreach singleton collision at $q_{S,i,k}$, $k\in\mathbb{N}^+$ **do**
      Find last non-traveling state $q_{S,p,k-1}\in Q_i$ before $q_{S,i,k}$;
      Find successive non-traveling state $q_{S,p,k+1}\in Q_i$ after $q_{S,i,k}$;
      Add "*wait*" transitions $(q_{S,i,k-1},q_{S,i,k-1})$ and $(q_{S,i,k+1},\ q_{S,i,k+1})$;
    foreach pairwise collision on $(q_{P,i,m},q_{P,i,l})$, $m,l\in\mathbb{N}^+$ **do**
      Find non-traveling states $q_{P,i,m-m'}$, $q_{P,i,l+l'}\in Q_i$ for $q_{P,i,m}$ and $q_{P,i,l}$, respectively.;
      Find state $q'_{i,m-m'}\in Q_i$ with lowest two-way cost $w_i((q_{P,i,m-m'},q'_{i,k}))+w_i((q'_{i,k},q_{P,i,m-m'}))$;
      if $(q_{P,i,m-m'},q'_{i,k})\notin\delta_i$, $(q'_{i,k},q_{P,i,m-m'})\in\delta_i$ **then**
        Add "go-back" transition $(q_{P,i,m-m'},q'_{i,k})$;
        $w_i((q_{P,i,m-m'},q'_{i,k}))\leftarrow w_i((q'_{i,k},q_{P,i,m-m'}))+g$;
      else if $(q_{P,i,m-m'},q'_{i,k})\in\delta_i$, $(q'_{i,k},q_{P,i,m-m'})\notin\delta_i$ **then**
        Add "go-back" transition $(q'_{i,k},q_{P,i,m-m'})$;
        $w_i((q'_{i,k},q_{P,i,m-m'}))\leftarrow w_i((q_{P,i,m-m'},q'_{i,k}))+g$;
      Add "wait" transition $(q_{P,i,l+l'},q_{P,i,l+l'})$;
  Reconstruct team TS $\mathbf{T}$ with modified TSs $\{\mathbf{T}_1,\ldots,\mathbf{T}_m,\ldots,\mathbf{T}_{m+n}\}$;
  Delete singleton collision states $\{(q_{1,k},\ldots,q_{S,i,k},q_{S,j,k},\ldots)\}$ in TS $\mathbf{T}$;
  Delete transitions to pairwise collisions $\{((q_{1,m-1},\ldots,q_{P,i,m-1},\ q_{P,j,m-1},\ldots),(q_{1,m},\ldots,q_{P,i,m},q_{P,j,m},\ldots))\}$;
  Delete transitions from pairwise collisions $\{((q_{1,l},\ldots,q_{P,i,l},\ q_{P,j,l},\ldots),(q_{1,l+1},\ldots,q_{P,i,l+1},q_{P,j,l+1},\ldots))\}$;
  Re-run OPTIMAL-RUN to find optimal run $r_T^{*\prime}$;
  $r_t^*\leftarrow r_T^{*\prime}$;
  foreach *TS* $\mathbf{T}_i$, $1\le i\le m$ **do**
    Project $r_{\mathbf{T}}^*$ onto $\mathbf{T}_i$ to obtain r_i^*;
return Optimal team run $r_{\mathbf{T}}^*$;
    Optimal individual runs $\{r_1^*,\ldots,r_m^*\}$;

Second, to further illustrate the potential practical feasibility of the proposed approach, we conduct a simulation including 20 agents in the partitioned environment extended from the environment of Fig. 1. In this simulation, 4 controllable TSs are deployed for the mission specified by LTL formula $\phi=\square(gather\rightarrow(a1gather\wedge a2gather\wedge a3gather\wedge a4gather))\wedge\square\diamond\pi$, and $\pi\ =\ a1gather\ \wedge\ a2gather\wedge a3gather\wedge a4gather$, i.e., controllable TSs $\mathbf{T}_1$, $\mathbf{T}_2$, $\mathbf{T}_3$ and $\mathbf{T}_4$ are required to synchronously gather data infinitely often, while the other 16 TSs are uncontrollable and move along the fixed trajectories. These TSs are used to construct the team TS such that the number of states in the product automaton will increase. Thus, in this example, more states are searched so as to find the optimal collision-free trajectories by the proposed algorithm. We perform the proposed algorithm for the 20 TSs, and the synthesized trajectories are shown in Fig. 3, which is correct and life-long collision-free.

Conclusion: In this letter, the multi-agent path planning problem with the requirement of collision avoidance has been investigated.

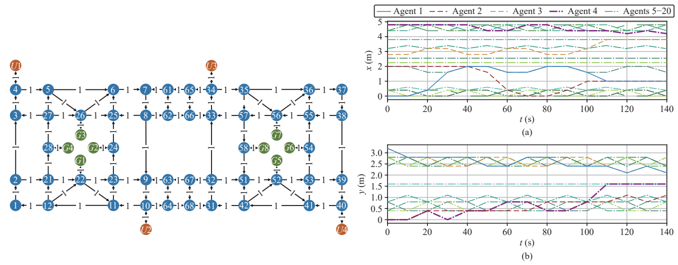

Fig. 3. Partitioned environment and synthesized trajectories for TSs $\mathbf{T}_1-\mathbf{T}_{20}$ of the simulation, in which (a) and (b) are x and y trajectories for agents 1–20, respectively.

We first propose a mechanism to detect all singleton and pairwise collisions in the synthesized run of the team TS. Then, by adding the "wait" and "go-back" transitions to the individual TSs, the optimal collision-free individual runs can be synthesized for the controllable agents in the stage of offline planning. The examples show that the agents can successfully complete the high-level mission specifications while avoiding all collisions without proximate sensing.

Acknowledgment: This work was supported in part by the National Key R&D Program of China (2021ZD0112600), the National Natural Science Foundation of China (61873219, 62103344, 62173283), the Natural Science Foundation of Fujian Province of China (2021-J01051, 2021J05015), and Science and Technology Projects of Innovation Laboratory for Sciences and Technologies of Energy Materials of Fujian Province (IKKEM) (HRTP-[2022]-25).

References

[1] C. Zu, C. Yang, J. Wang, W. Gao, D. Cao, and F.-Y. Wang, "Simulation and field testing of multiple vehicles collision avoidance algorithms," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 1045–1063, 2020.

[2] Z. Zuo, C. Liu, Q.-L. Han, and J. Song, "Unmanned aerial vehicles: Control methods and future challenges," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 4, pp. 601–614, 2022.

[3] Z. Peng, J. Wang, D. Wang, and Q.-L. Han, "An overview of recent advances in coordinated control of multiple autonomous surface vehicles," *IEEE Trans. Ind. Inf.*, vol. 17, no. 2, pp. 732–745, 2020.

[4] J. Wang, J. Wang, and Q.-L. Han, "Receding-horizon trajectory planning for under-actuated autonomous vehicles based on collaborative neurodynamic optimization," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 11, pp. 1909–1223, 2022.

[5] C. Baier and J.-P. Katoen, *Principles of Model Checking*. Cambridge, USA: MIT Press, 2008.

[6] P. Schillinger, M. Bürger, and D. V. Dimarogonas, "Simultaneous task allocation and planning for temporal logic goals in heterogeneous multi-robot systems," *Int. J. Rob. Res.*, vol. 37, no. 7, pp. 818–838, 2018.

[7] S. L. Smith, J. Tumova, C. Belta, and D. Rus, "Optimal path planning for surveillance with temporal-logic constraints," *Int. J. Rob. Res.*, vol. 30, no. 14, pp. 1695–1708, 2011.

[8] K. Leahy, E. Cristofalo, C.-I. Vasile, A. Jones, E. Montijano, M. Schwager, and C. Belta, "Control in belief space with temporal logic specifications using vision-based localization," *Int. J. Rob. Res.*, vol. 38, no. 6, pp. 702–722, 2019.

[9] X. Ding, S. L. Smith, C. Belta, and D. Rus, "Optimal control of markov decision processes with linear temporal logic constraints," *IEEE Trans. Autom. Control*, vol. 59, no. 5, pp. 1244–1257, 2014.

[10] J. Alonso-Mora, J. A. DeCastro, V. Raman, D. Rus, and H. Kress-Gazit, "Reactive mission and motion planning with deadlock resolution avoiding dynamic obstacles," *Auton. Rob.*, vol. 42, no. 4, pp. 801–824, 2018.

[11] X. Li, Z. Serlin, G. Yang, and C. Belta, "A formal methods approach to interpretable reinforcement learning for robotic planning," *Sci. Rob.*, vol. 4, no. 37, p. eaay6276, 2019.

[12] A. Ulusoy, S. L. Smith, X. C. Ding, C. Belta, and D. Rus, "Optimality and robustness in multi-robot path planning with temporal logic constraints," *Int. J. Rob. Res.*, vol. 32, no. 8, pp. 889–911, 2013.

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

[14] Z. A. Ali and K. Yakovlev, "Prioritized sipp for multi-agent path finding with kinematic constraints," in *Proc. Intell. Conf. Interact. Collab. Rob.*, 2021, pp. 1–13.