



# Task allocation and trajectory planning for multiple agents in the presence of obstacle and connectivity constraints with mixed-integer linear programming

Rubens J.M. Afonso<sup>1,2</sup> | Marcos R.O.A. Maximo<sup>3</sup> | Roberto K.H. Galvão<sup>2</sup>

<sup>1</sup>Institute of Flight System Dynamics, Department of Aerospace and Geodesy, Technical University of Munich, Bavaria, Germany

<sup>2</sup>Electronic Engineering Division, Instituto Tecnológico de Aeronáutica, São Paulo, Brazil

<sup>3</sup>Autonomous Computational Systems Lab (LAB-SCA), Computer Science Division, Instituto Tecnológico de Aeronáutica, São Paulo, Brazil

## Correspondence

Rubens J.M. Afonso, Divisão de Engenharia Eletrônica, Instituto Tecnológico de Aeronáutica, São José dos Campos, SP 12228-900, Brazil.  
Emails: rubensjm@ita.br, rubens.afonso@tum.de

## Funding information

Alexander von Humboldt-Stiftung; Conselho Nacional de Desenvolvimento Científico e Tecnológico, Grant/Award Number: 303393/2018-1; Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Grant/Award Number: 88881.145490/2017-01; Fundação de Amparo à Pesquisa do Estado de São Paulo, Grant/Award Number: 2016/03647-3

## Summary

This article addresses the problem of maneuvering multiple agents that must visit a number of target sets, while enforcing connectivity constraints and avoiding obstacle as well as interagent collisions. The tool to cope with the problem is a formulation of model predictive control including binary decision variables. In this regard, two mixed-integer linear programming formulations are presented, considering a trade-off between optimality and scalability between them. Simulation results are also shown to illustrate the main features of the proposed approaches.

## KEYWORDS

connectivity maintenance, cooperative multiple agents, mixed-integer linear programming, obstacle avoidance

## 1 | INTRODUCTION

Miniaturization, enhancement in communication capabilities and reduction in costs for building autonomous agents have enabled the development of multiagent systems (MASs). A cooperative team of agents may accomplish tasks, such as patrolling,<sup>1</sup> area coverage,<sup>2</sup> search and rescue,<sup>3</sup> and exploration,<sup>4</sup> in a better manner than a single agent. For complex tasks, the performance of the agent team depends on coordination strategies, which often require communication between agents.<sup>5</sup> In this regard, maintaining connectivity of the communication network during the whole duration of the

**Abbreviations:** MAS, multiagent system; MILP, mixed-integer linear programming; MPC, model predictive control.

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2020 The Authors. *International Journal of Robust and Nonlinear Control* published by John Wiley & Sons, Ltd.

task is necessary to avoid interruptions of the coordination algorithms.<sup>6</sup> For modeling purposes, a common assumption states that a communication link is established between two agents if they are within a given distance. The kind of problem addressed in the present work is similar to Kantaros and Zavlanos (2016)<sup>7</sup> and Stephan et al (2017).<sup>8</sup> In Kantaros and Zavlanos (2016)<sup>7</sup> spatially distributed objectives (therein termed *tasks*) are assigned to a subgroup of the agents (the so-called *leaders*), while the remaining ones move to comply with the connectivity constraints. However, the initial task assignment does not consider obstacles and collisions between the agents, which are taken into account later in the planning phase. Another recent work that performs the task assignment and the planning in two distinct hierarchical layers can be found in Stephan et al (2017).<sup>8</sup>

Considering a predefined task assignment, that is, an a priori definition of the final configurations for each agent, Filotheou et al (2018)<sup>9</sup> addressed the problem of guiding MAS in which each agent presented nonlinear second-order Lagrangian dynamics while avoiding collisions and maintaining connectivity. Recently Karkoub et al (2019)<sup>10</sup> proposed an algorithm for the control of MAS with unicycle dynamics endowed with interagent collision avoidance and connectivity maintenance capabilities. However, the article is restricted to the agent steering problem, under the assumption that a preplanned trajectory is available. The higher layer concerning trajectory planning was not addressed.

Many control laws have been proposed in the literature to maintain connectivity.<sup>6</sup> In a more precise manner, the interest here is to develop algorithms that guarantee the communication graph stays connected if it is so at the beginning of the task. Early researches in this field usually worked with a concept termed *local connectivity*, where the control law is developed to ensure that each initially active communication link will remain active for the rest of the task.

Although keeping the local connectivity of each communication link guarantees the connectivity of the overall communication graph, it is often overly conservative and the agent team may be better off to fulfill its objectives if global connectivity is considered. At each time instant, global connectivity refers to the usual connectivity concept from graph theory: the communication graph may have missing links, as long as a path exists between each pair of agents. Since the agents move, creation, and deletion of links may happen dynamically, so global connectivity does not ensure local connectivity.

Task allocation and motion planning stem as fundamental problems in many applications. Many motion planning techniques for MASs are extensions of the ones developed for a single-mobile agent, which may be categorized mainly into potential fields,<sup>11</sup> graph-based,<sup>12</sup> or sampling-based<sup>13</sup> approaches. In a MAS, the agents should not only avoid collision with environment obstacles but also take interagent collision avoidance into account.<sup>5</sup> One alternative commonly present in the literature to simplify the problem is to assume that some agents are “leaders” and others are their so-called “followers.” In such formulations, usually only the leaders are assigned tasks and the followers are required to be kept connected, but not necessarily perform any task. One interesting result for single-integrator agents are the bounds on the ratio of leaders to followers that ensure connectivity can be kept.<sup>14</sup> However, such formulations usually attain suboptimal results in view of the simplifications brought by the assignment of agents to the roles of either leaders or followers. Thus, in the present article we aim at coping with a topology in which each agent has the same role, being free to move as to perform the tasks (herein target set visitations) optimally, provided that the constraints are enforced, including connectivity maintenance and collision avoidance.

An approach that uses model predictive control (MPC) with a mixed-integer linear programming (MILP) encoding to allow trajectory planning endowed with circumvention of polygonal obstacles was proposed in Richards and How (2006)<sup>15</sup> in the context of a single-agent subject to disturbances. In the undisturbed case, a similar approach to a MAS was proposed in Schouwenaars et al (2001).<sup>16</sup> Moreover, Prodan et al (2012)<sup>17</sup> proposed a scheme with two layers, namely, a higher layer that assigns tasks represented as target states to each agent at the end of the horizon and a lower layer that optimizes the trajectory of the agents toward their assigned tasks while avoiding collisions between them. Although this method solves task allocation and trajectory planning within a MPC framework, it does so in separate layers, resulting in suboptimality, since the optimization solver is not able to exploit the relationship between task allocation and trajectory planning.

This article presents a MILP formulation for task allocation and trajectory planning, which respects collision avoidance and connectivity constraints. Our work follows on the steps of previous MILP formulations for trajectory planning,<sup>15,16,18-20</sup> but novel contributions are brought to the field. To the best of our knowledge, our formulation is the first one to explicitly consider global connectivity in MAS trajectory planning using an optimization framework. We may add that a MPC formulation permits us to handle constraints that are relevant for real-world agents, such as actuator saturation. In this regard, many proposed control laws only guarantee connectivity maintenance if high bandwidth

unbounded control effort is available,<sup>21</sup> which is unrealistic on real-world agents. Furthermore, in our problem, the agents may collect rewards within specific regions during the task. Any agent may collect any reward, thus we also encode this task allocation within the MILP, allowing an optimal choice of which agent should collect which rewards.

The main contributions are 2-fold: (i) the global connectivity problem is reformulated in a graph theoretical framework and we propose constraints that enforce that the graph has a spanning tree, which ensures connectivity. Such an approach in the context of global connectivity maintenance in a MAS is novel, to the best of our knowledge; (ii) we propose alternative constraints to ensure the existence of a spanning tree within the graph, reducing the growth in the number of constraints from exponential to linear regarding the number of agents, at the cost of additional conservatism. Moreover, we propose an algorithm to perform the initial enumeration of the agents and demonstrate that it yields a feasible optimization problem, even with the more conservative constraints.

The remaining of this article is organized as follows. Section 2 provides theory background and explains the mathematical formulation of the trajectory planner. In Section 3, simulation results are shown to validate the MILP approach presented here. Finally, Section 4 concludes and shares our ideas for future investigation.

## 1.1 | Notation

$V$	Set of vertices
$E$	Set of edges
$G(V,E)$	Graph formed by the vertices in $V$ and edges in $E$
$A_G$	Adjacency matrix of the graph $G$
$ \bullet $	Number of elements of a set $\bullet$
$\bullet^T$	Transpose of a matrix $\bullet$
$\mathbf{x} \in \mathbb{R}^{n_x}$	State vector of the agent
$\mathbf{u} \in \mathbb{R}^{n_u}$	Control vector of the agent
$n_x \in \mathbb{N}$	Dimension of the state vector of the agent
$n_u \in \mathbb{N}$	Dimension of the control vector of the agent
$\mathbf{x}_k \in \mathbb{R}^{n_x}$	State vector of the agent at sample time $k$
$\mathbf{u}_k \in \mathbb{R}^{n_u}$	Control vector of the agent at sample time $k$
$x$	Position of the agent regarding a coordinate axis
$y$	Position of the agent regarding a coordinate axis perpendicular to $x$
$v_x$	Velocity of the agent regarding the $x$ -axis
$v_y$	Velocity of the agent regarding the $y$ -axis
$a_x$	Acceleration of the agent regarding the $x$ -axis
$a_y$	Acceleration of the agent regarding the $y$ -axis
$A \in \mathbb{R}^{n_x \times n_x}$	State transition matrix
$B \in \mathbb{R}^{n_x \times n_u}$	Control matrix
$Q \subset \mathbb{R}^{n_x}$	Target set
$\mathcal{O}_j \subset \mathbb{R}^{n_x}$	$j$ th obstacle
$N_o \in \mathbb{N}$	Number of obstacles within the scenario
$\mathcal{X} \subset \mathbb{R}^{n_x}$	Set of admissible states
$\mathcal{U} \subset \mathbb{R}^{n_u}$	Set of admissible controls
$N \in \mathbb{N}$	Variable horizon
$\gamma \in \mathbb{R}_+$	Weight of the control term within the cost function
$N_{os} \in \mathbb{N}$	Number of facets (half-planes) of each obstacle
$n_p \in \mathbb{N}$	Number of position dimensions

$C \in \mathbb{R}^{n_p \times n_x}$	Matrix that extracts position information from the state vector
$P_j^{\text{obs}} \in \mathbb{R}^{N_{\text{os}} \times n_p}$	Constant matrix in the left-hand side of the half-plane inequalities defining obstacle $\mathcal{O}_j$
$\mathbf{q}_j^{\text{obs}} \in \mathbb{R}^{N_{\text{os}}}$	Constant vector in the right-hand side of the half-plane inequalities defining obstacle $\mathcal{O}_j$
$\mathbf{p}_{j,n}^{\text{obs}} \in \mathbb{R}^{n_p}$	Constant column vector equal to the transpose of the $n$ th row of $P_j^{\text{obs}}$
$q_{j,n}^{\text{obs}} \in \mathbb{R}$	Constant equal to the $n$ th element of $\mathbf{q}_j^{\text{obs}}$
$M \in \mathbb{R}$	“Big-M” constant
$b_{j,n,k}^{\text{obs}} \in \{0, 1\}$	Binary variable to impose that the position at the $k + 1$ th instant is to the outer side of the $n$ th half-plane of the $j$ th obstacle
$\epsilon \in \mathbb{R}_+$	Constant subtracted from the right-hand side of the nonstrict inequalities “ $\leq$ ” to make the left-hand side obey a strict inequality “ $<$ ” with respect to the remaining terms on the right-hand side
$T \in \mathbb{N}$	Fixed horizon for the reformulation of the time-fuel minimal maneuvering problem as a MILP
$N_{\text{ts}} \in \mathbb{N}$	Number of facets (half-planes) of the target set $Q$
$P^{\text{target}} \in \mathbb{R}^{N_{\text{ts}} \times n_p}$	Constant matrix in the left-hand side of the half-plane inequalities defining the target set $Q$
$\mathbf{q}^{\text{target}} \in \mathbb{R}^{N_{\text{ts}}}$	Constant vector in the right-hand side of the half-plane inequalities defining the target set $Q$
$b_k^{\text{hor}} \in \{0, 1\}$	Binary variable to impose that the position at the $k + 1$ th instant is within the target set $Q$
$\mathbf{1}_\bullet \in \mathbb{R}^\bullet$	Column vector with $\bullet$ elements all equal to one
$N_a \in \mathbb{N}$	Number of agents
$\mathbf{x}_{\ell,k} \in \mathbb{R}^{n_x}$	State vector of the $\ell$ th agent at sample time $k$
$\mathbf{u}_{\ell,k} \in \mathbb{R}^{n_u}$	Control vector of the $\ell$ th agent at sample time $k$
$x_{\ell,k}$	Position of the $\ell$ th agent regarding a coordinate axis at sample time $k$
$Y_{\ell,k}$	Position of the $\ell$ th agent regarding a coordinate axis perpendicular to $x$ at sample time $k$
$v_{x,\ell,k}$	Velocity of the $\ell$ th agent regarding the $x$ axis at sample time $k$
$v_{y,\ell,k}$	Velocity of the $\ell$ th agent regarding the $y$ axis at sample time $k$
$a_{x,\ell,k}$	Acceleration of the $\ell$ th agent regarding the $x$ axis at sample time $k$
$a_{y,\ell,k}$	Acceleration of the $\ell$ th agent regarding the $y$ axis at sample time $k$
$b_{\ell,j,n,k}^{\text{obs}} \in \{0, 1\}$	Binary variable to impose that the position of the $\ell$ th agent at the $k + 1$ th instant is to the outer side of the $n$ th half-plane of the $j$ th obstacle
$x_{j,\text{min}} \in \mathbb{R}$	$x$ coordinate of the lower left extreme point of the $j$ th rectangular obstacle
$Y_{j,\text{min}} \in \mathbb{R}$	$y$ coordinate of the lower left extreme point of the $j$ th rectangular obstacle
$x_{j,\text{max}} \in \mathbb{R}$	$x$ coordinate of the upper right extreme point of the $j$ th rectangular obstacle
$Y_{j,\text{max}} \in \mathbb{R}$	$y$ coordinate of the upper right extreme point of the $j$ th rectangular obstacle
$N_{\text{cs}} \in \mathbb{N}$	Number of facets (half-planes) for interagent collision avoidance
$P^{\text{col}} \in \mathbb{R}^{N_{\text{cs}} \times n_p}$	Constant matrix in the left-hand side of the half-plane inequalities for interagent collision avoidance
$\mathbf{q}^{\text{col}} \in \mathbb{R}^{N_{\text{cs}}}$	Constant vector in the right-hand side of the half-plane inequalities for interagent collision avoidance
$b_{i,\ell,n,k}^{\text{col}} \in \{0, 1\}$	Binary variable to impose that the position of the $\ell$ th agent at the $k$ th instant is to the outer side of the $n$ th half-plane of the interagent collision avoidance polyhedron around the $i$ th agent
$\mathbf{p}_n^{\text{col}} \in \mathbb{R}^{n_p}$	Constant column vector equal to the transpose of the $n$ th row of $P^{\text{col}}$
$q_n^{\text{col}} \in \mathbb{R}$	Constant equal to the $n$ th element of $\mathbf{q}^{\text{col}}$
$J$	Cost function used in the optimization problem
$b_{\ell,m,k}^{\text{target}} \in \{0, 1\}$	binary variable to impose that the position of the $\ell$ th agent at the $k + 1$ th instant is within the $m$ th target set
$R \in \mathbb{R}_+$	Reward for visiting a target set
$N_{\text{sc}} \in \mathbb{N}$	Number of facets (half-planes) for interagent communication range
$P^{\text{con}} \in \mathbb{R}^{N_{\text{sc}} \times n_p}$	Constant matrix in the left-hand side of the half-plane inequalities for connectivity constraints
$\mathbf{q}^{\text{con}} \in \mathbb{R}^{N_{\text{sc}}}$	Constant vector in the right-hand side of the half-plane inequalities for connectivity constraints
$b_{i,\ell,k}^{\text{con}} \in \{0, 1\}$	Binary variable to impose that the position of the $\ell$ th agent at the $k$ th instant is within the communication range polyhedron around the $i$ th agent

$x_{c,\max} \in \mathbb{R}$	Bound on the maximal distance between agents in the $x$ direction for communication range
$y_{c,\max} \in \mathbb{R}$	Bound on the maximal distance between agents in the $y$ direction for communication range
$O(\bullet)$	Notation to indicate that the order (either memory usage or computation time) of an algorithm is dominated by the quantity $\bullet$
$T_G$	A tree
$r$	Root node of a tree
$L$	A list of the vertices obtained by traversing a tree in postorder and their indices obtained during the traversal
$C_i$	Set of children of the $i$ th node of a tree, where $i$ is the index of this node in postorder

## 2 | MATHEMATICAL FORMULATION

In this section, the problem of maneuvering a connected MAS through an obstacle field populated with target sets is formulated in mathematical terms using graph theory. Two encodings of the connectivity constraints are presented. Both ensure connectivity maintenance throughout the maneuver, the difference being that the first one models exactly the connectivity problem, at the cost of an exponential growth in the number of constraints with the number of agents, whereas the second is more conservative, with the advantage of a polynomial (linear) dependence of the number of constraints upon the number of agents.

### 2.1 | Multiagent communication graph

When global connectivity is of concern, we may encode the MAS from a communication standpoint as a time-varying graph  $G(t) = \{V, E(t)\}$ ,<sup>6</sup> where the set of vertices  $V = \{v_1, v_2, \dots, v_{N_a}\}$  represents the agents and the set of edges  $E(t) = \{(v_i, v_j) | v_i \in V, v_j \in V\}$  represents the active communication links between agents at time  $t$ . In this work, we consider that a communication link is established between two agents if they are within communication range of each other. Since we are not interested in properties of the communication link, but merely in its existence, no weights are considered for the edges. Moreover, communication is considered to be bidirectional, thus the graph  $G(t)$  is undirected. Finally, self-links are considered to be of no interest, so no edge  $(v_i, v_i)$  is ever present in  $G(t)$ .

In this graph framework, agent  $i$  may send information to agent  $j$  if there is a path from  $v_i$  to  $v_j$  in  $G(t)$ . The graph  $G(t)$  may be represented by an adjacency matrix  $A_{G(t)} \in \mathbb{R}^{N_a \times N_a}$ . Let us denote  $a_{ij}(t)$  as the element at row  $i$  and column  $j$  of  $A_{G(t)}$ ,  $a_{ij}(t) = 1$  if the edge  $(v_i, v_j)$  exists in  $G(t)$ , while  $a_{ij}(t) = 0$  otherwise. Given that  $G(t)$  is undirected, we also have  $a_{ij}(t) = a_{ji}(t), \forall i, \forall j$ . Therefore, formally, the MAS has global connectivity at time  $t$  if for each  $v_i \in V$  and  $v_j \in V$ , there is a path between  $v_i$  and  $v_j$ .

For the sake of providing background for the following discussions, we will introduce some additional concepts from graph theory. For a more in-depth discussion, the interested reader may refer to textbooks on graph theory, such as Cormen et al (2009).<sup>22</sup>

**Definition 1.** The graph  $G(S) = (S, E(S))$  is called an induced subgraph of  $G = (V, E)$  by  $S$  if  $S \subseteq V$  and  $E(S) = \{(v_i, v_j) \in E | v_i \in S, v_j \in S\}$ .

**Definition 2.** A path is a nonempty graph  $P = (V_P, E_P)$  such that  $V_P = \{v_{P,1}, v_{P,2}, \dots, v_{P,n}\}$  and  $E_P = \{(v_{P,1}, v_{P,2}), (v_{P,2}, v_{P,3}), \dots, (v_{P,n-1}, v_{P,n})\}$ , where  $v_{P,i} \neq v_{P,j}$ , for  $i = 1, 2, \dots, n$  and  $i \neq j$ .

**Definition 3.** A cycle is a nonempty graph  $P = (V_P, E_P)$  such that  $V_P = \{v_{P,1}, v_{P,2}, \dots, v_{P,n}\}$  and  $E_P = \{(v_{P,1}, v_{P,2}), (v_{P,2}, v_{P,3}), \dots, (v_{P,n-1}, v_{P,n}), (v_{P,n}, v_{P,1})\}$ , where  $v_{P,i} \neq v_{P,j}$ , for  $i = 1, 2, \dots, n$  and  $i \neq j$ .

**Definition 4.** A tree is a connected undirected graph from which none of the induced subgraphs are cycles.

**Definition 5.** A spanning tree  $T_G = (V, E_T)$  of an undirected graph  $G = (V, E)$  is a subgraph of  $G$  that is a tree.

*Remark 1.* A connected graph  $G$  has at least one spanning tree, and may have several if there are cycles in  $G$ . A graph that is not connected does not have a spanning tree.

*Remark 2.* A spanning tree is a subgraph with the minimum possible number of edges while maintaining connectivity, since the inexistence of cycles means that removing any edge would split the graph into two connected components.

**Theorem 1.** If an undirected graph  $G = (V, E)$  has  $|E| = |V| - 1$  and does not have cycles, then  $G$  is a tree.

The proof of Theorem 1 is omitted for brevity, as it is standard in graph theory.

**Corollary 1.** *A subgraph  $G' = (V', E')$  of an undirected graph  $G = (V, E)$  with  $|V'| = |V|$ ,  $|E'| = |V'| - 1$ , and no cycles is a spanning tree.*

## 2.2 | Mixed-integer linear programming for trajectory planning

In the present section, the employed model and notation are introduced alongside a discussion on how the maneuvering problem of an agent avoiding obstacles may be reformulated as a MPC-MILP optimization problem. The adopted formulation is that of Richards and How (2006),<sup>15</sup> but a simplification is made: only the nominal (undisturbed case) is considered. This simplification allows us to focus on the contributions of the present article, namely, coping with trajectory planning for MAS encompassing obstacle and interagent collision avoidance, flexible assignment between agents and visited target sets, and connectivity constraints within a single MILP. However, the constraint-tightening approach in Richards and How (2006)<sup>15</sup> can be seamlessly incorporated to the proposal within the present work to enable robustness to unknown disturbances.

A discrete-time linear model involving positions, velocities, and accelerations of the agent in a plane with axes  $x$  and  $y$  is considered. The inputs are  $a_x$  and  $a_y$  (accelerations along each axis). The velocities  $v_x$  and  $v_y$  and positions  $x$  and  $y$  are components of the state vector. The discrete-time model can be written as  $\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k$ , with the state and control vectors  $\mathbf{x}^T = [x \ v_x \ y \ v_y] \in \mathbb{R}^{n_x}$  and  $\mathbf{u}^T = [a_x \ a_y] \in \mathbb{R}^{n_u}$ , respectively. For a unit sampling time, the model matrices are

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0 \\ 1 & 0 \\ 0 & 0.5 \\ 0 & 1 \end{bmatrix}, \quad (1)$$

which correspond to double integrator dynamics in each axis.

The maneuver in Richards and How (2006)<sup>15</sup> consists of moving the agent starting at the initial state  $\mathbf{x}_{\text{initial}}$  to a polyhedral target set  $Q = \{P^{\text{target}} C\mathbf{x}_k \leq \mathbf{q}^{\text{target}}\}$ , where  $C$  is a matrix that extracts position information from the state vector, while avoiding  $N_o$  polyhedral obstacles  $\mathcal{O}_j$ ,  $1 \leq j \leq N_o$ , and minimizing a weighted time-fuel cost function. The state and control are constrained to admissible sets  $\mathcal{X}$  and  $\mathcal{U}$ , respectively, such that  $0 \in \mathcal{U}$ , with  $\mathcal{X}$  and  $\mathcal{U}$  also polyhedral. The optimization problem results in a sequence of inputs  $\mathbf{u}_k$  that steer the state from  $\mathbf{x}_0 = \mathbf{x}_{\text{initial}}$  to a point in  $Q$ . In this article, the notation for predicted values will simply be an index  $k$  referring to the prediction at the  $k$ th instant from the current instant ( $k=0$ ), as in Fleming et al (2015).<sup>23</sup> The maneuvering optimization problem is posed as:

**Problem 1.**

$$\min_{\mathbf{x}_k, \mathbf{u}_k, N} N + \gamma \sum_{k=0}^N \|\mathbf{u}_k\|_1, \quad (2)$$

subject to

$$\mathbf{x}_0 = \mathbf{x}_{\text{initial}}, \quad (3a)$$

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k, \quad 0 \leq k \leq N, \quad (3b)$$

$$\mathbf{x}_{k+1} \in \mathcal{X}, \quad 0 \leq k \leq N, \quad (3c)$$

$$\mathbf{u}_k \in \mathcal{U}, \quad 0 \leq k \leq N, \quad (3d)$$

$$\mathbf{x}_{k+1} \notin \mathcal{O}_j, \quad 0 \leq k \leq N, \quad 1 \leq j \leq N_o. \quad (3e)$$

$$\mathbf{x}_{N+1} \in \mathcal{Q}. \quad (3f)$$

The choice of the weight  $\gamma \in \mathbb{R}_+$  balances between optimization of the cost of fuel during the maneuver and the time to reach the terminal set, represented by the optimization variable  $N$ . The maneuver is concluded at time  $N + 1$ , as seen in (3f).

This problem involves two major difficulties for using numerical optimization algorithms, namely: the variable horizon  $N$  and the loss of convexity of the feasible set due to the obstacle avoidance constraints in (3e). These issues are both dealt with by the introduction of binary optimization variables in the problem, along with the so-called “big-M” approach.<sup>24</sup> The interested reader is referred to some of the following references for a thorough presentation of the schemes employed in writing a MILP whose solution is equivalent to the solution of Problem 1: Schouwenaars et al (2001),<sup>16</sup> Richards et al (2002),<sup>25</sup> and Richards and How (2006).<sup>15</sup> For general discussions about mixed-integer programming and its application in control we refer the reader to Jünger et al (2009),<sup>26</sup> Vielma (2015),<sup>27</sup> and Prodan et al (2016).<sup>28</sup>

The variable horizon issue is dealt with by using one binary variable  $b_k^{\text{hor}}$  for each time instant and imposing the terminal set constraints at every time instant along a fixed horizon  $T$ , using the “big-M” method to relax the constraints up until the target is reached. Moreover, the optimal solution of Problem 1 does not impose the state, control, and obstacle avoidance constraints after the variable horizon  $N$  (reaching of the target set). Therefore, the state, control, and obstacle avoidance constraints written up to the fixed horizon  $T$  are relaxed after the target set is reached. Additionally, the equality constraints in (3b) can be split into two inequalities and relaxed after the horizon. Now, with each constraint relaxed after the horizon, since  $0 \in \mathcal{U}$ , the optimal solution is to make  $\mathbf{u}_k = 0, k > N$ . Notice that for this to hold one must have  $T$  greater than or equal to the optimal  $N$ . If that is the case, the optimal solution of the MILP problem with the presented “big-M” constraints is the same as the one of the original Problem 1, but can now be obtained via faster and more reliable numerical methods.

On the other hand, the nonconvexity of the exclusion constraints in (3e), that is, the obstacle avoidance constraints, requires additional use of binary variables. If the sets  $\mathcal{O}_j, 1 \leq j \leq N_o$  are polyhedra with  $N_{os}$  faces each, then the obstacles can be represented as the intersection of  $N_{os}$  half-planes  $\mathcal{O}_j = \{P_j^{\text{obs}} \mathbf{C} \mathbf{x}_k \leq \mathbf{q}_j^{\text{obs}}\}, 1 \leq j \leq N_o$ , where the inequality is elementwise. Generalization to polyhedra with different number of faces is straightforward. For a point to be outside the obstacle  $\mathcal{O}_j$ , it suffices that one of the  $N_{os}$  inequalities is not respected, therefore the avoidance problem can be written by using “or” constraints, with the choice being that one of the  $N_{os}$  inequalities does not hold, that is,  $\exists n, 1 \leq n \leq N_{os} \mid (\mathbf{p}_{j,n}^{\text{obs}})^T \mathbf{C} \mathbf{x}_k > q_{j,n}^{\text{obs}}$ , where  $(\mathbf{p}_{j,n}^{\text{obs}})^T$  is the  $n$ th row of  $P_j^{\text{obs}}$  and  $q_{j,n}^{\text{obs}}$  the  $n$ th element of  $\mathbf{q}_j^{\text{obs}}$ . These “or” constraints can be imposed using the “big-M” method.<sup>24</sup>

One interesting enhancement to the formulation in Richards and How (2006)<sup>15</sup> was proposed in Maia and Galvão (2009)<sup>18</sup> involving the inclusion of additional constraints on the obstacle avoidance binary variables. The aim was to ensure that the consecutive positions of the agent at times  $k$  and  $k + 1$  did not produce a continuous-time trajectory that intersected the obstacle. This procedure was later made more economic in Richards and Turnbull (2015).<sup>19</sup> The elegant result in Richards and Turnbull (2015)<sup>19</sup> is that, to avoid intersample collision with the obstacles, it suffices to impose the same avoidance collisions at consecutive time instants. A later advancement was reported in Stoican et al (2018).<sup>20</sup> Using the so-called “shadow regions” the authors characterized the regions that may be reached by the agent in one time step without colliding with the obstacles and merged them in a collection of convex polyhedra, which may require less binary variables. However, the ensuing optimization problem is a mixed-integer nonlinear program (MILNP), which is known to be more cumbersome. Simplifications could be introduced to recast the problem into a MILP form, but at the cost of losing some of the benefits of the shadow region method. Therefore, in the present article, the approach in Richards and Turnbull (2015)<sup>19</sup> was adopted for its simplicity.

Finally, to cope with the appearance of the norm one of the control variables in (2), while keeping a linear cost function, one may use additional real-valued variables  $v_{k,i}, k = 0, 1, \dots, T, i = 1, 2, \dots, n_u$  and constraints as in Section 6.1.1 of Boyd and Vandenberghe (2009).<sup>29</sup> Thus,  $(T + 1)n_u$  additional continuous variables and  $2(T + 1)n_u$  inequalities are required.

### 2.3 | Multiagent trajectory planning with interagent collision avoidance

A formulation of the maneuvering problem using MPC-MILP for multiple agents is presented in Schouwenaars et al (2001).<sup>16</sup> This multiagent planner is not a direct extension of the one presented in Subsection 2.2, since it does not

implement some of the features of the latter one, such as the variable horizon and the intersample avoidance constraints. However, two additional characteristics are interesting to present: extension to the multiple agent case and collision avoidance between agents. Such is also the case in the planner formulation in Richards et al (2002),<sup>25</sup> in which the terminal states can be chosen by the agents, but the horizon is still fixed.

In this case, the MAS consists of  $N_a$  agents. Although in Schouwenaars et al (2001)<sup>16</sup> and Richards et al (2002)<sup>25</sup> the possibility of each agent having different dynamics is considered, this will be not addressed in the present work, but the reader can see that extension to this case is easily done. Thus, each agent will be supposed to have the dynamics given as

$$\mathbf{x}_{\ell,k+1} = \mathbf{A}\mathbf{x}_{\ell,k} + \mathbf{B}\mathbf{u}_{\ell,k}, \quad 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T. \quad (4)$$

For each agent, the state vector is defined similarly as in the single agent case as  $\mathbf{x}_{\ell,k} = [x_{\ell,k} \ v_{x,\ell,k} \ y_{\ell,k} \ v_{y,\ell,k}]^T$  and the control vector is  $\mathbf{u}_{\ell,k} = [a_{x,\ell,k} \ a_{y,\ell,k}]^T$ . The formulation also contemplates state and command limits and obstacle avoidance by employing the following constraints (once again different limits for each agent could be assumed, but this is not done so in the present article to simplify the exposition):

$$\left. \begin{aligned} -\mathbf{x}_{\ell,k+1} &\leq -\mathbf{x}_{\min} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \\ \mathbf{x}_{\ell,k+1} &\leq \mathbf{x}_{\max} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \end{aligned} \right\} 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T. \quad (5)$$

$$\left. \begin{aligned} -\mathbf{u}_{\ell,k} &\leq -\mathbf{u}_{\min} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \\ \mathbf{u}_{\ell,k} &\leq \mathbf{u}_{\max} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \end{aligned} \right\} 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T. \quad (6)$$

$$\left\{ \begin{aligned} -(\mathbf{p}_{j,n}^{\text{obs}})^T \begin{bmatrix} x_{\ell,k+1} \\ y_{\ell,k+1} \end{bmatrix} &\leq -q_{j,n}^{\text{obs}} + M(1 - b_{j,n,k}^{\text{obs}}) - \epsilon + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, & 1 \leq j \leq N_o, & 1 \leq n \leq N_{\text{os}}, & 0 \leq k \leq T, & 1 \leq \ell \leq N_a, \\ -(\mathbf{p}_{j,n}^{\text{obs}})^T \begin{bmatrix} x_{\ell,k} \\ y_{\ell,k} \end{bmatrix} &\leq -q_{j,n}^{\text{obs}} + M(1 - b_{j,n,k}^{\text{obs}}) - \epsilon + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, & 1 \leq j \leq N_o, & 1 \leq n \leq N_{\text{os}}, & 0 \leq k \leq T, & 1 \leq \ell \leq N_a, \\ \sum_{n=1}^{N_{\text{os}}} b_{\ell,j,n,k}^{\text{obs}} &\geq 1, & 1 \leq \ell \leq N_a, & 1 \leq j \leq N_o, & 0 \leq k \leq T. \end{aligned} \right. \quad (7)$$

Constraints (5), (6), and (7) refer to state limits, command limits, and obstacle avoidance, respectively. In Schouwenaars et al (2001)<sup>16</sup> and Richards et al (2002)<sup>25</sup> rectangular obstacles are considered as defined by their lower left and upper right extremes  $[x_{j,\min}, y_{j,\min}]^T$  and  $[x_{j,\max}, y_{j,\max}]^T$ , respectively, for  $j = 1, 2, \dots, N_o$ . Notice that this is a particular case of the obstacle constraints presented in Subsection 2.2. The constant  $\epsilon$  is subtracted from the right-hand side of the nonstrict inequalities “ $\leq$ ” to make the left-hand side obey a strict inequality “ $<$ ” with respect to the remaining terms on the right-hand side, thus removing the possibility that the agent occupies a position at the boundary of the obstacles.

In this multiagent setting, agents may collide with each other, so new constraints must be added for interagent collision avoidance. As in the case of obstacle avoidance, assuming that the safe region around an agent is represented by a polyhedron  $\{\mathbf{p}^{\text{col}} \mathbf{C}\mathbf{x} \leq \mathbf{q}^{\text{col}}\}$ , to impose that the agents do not collide it is sufficient to impose the following constraints (similar to the obstacle avoidance ones):

$$\begin{aligned} & -(\mathbf{p}_n^{\text{col}})^T \begin{bmatrix} x_{i,k+1} - x_{\ell,k+1} \\ y_{i,k+1} - y_{\ell,k+1} \end{bmatrix} \\ & \leq -q_n^{\text{col}} + M(1 - b_{i,\ell,n,k}^{\text{col}}) + M \sum_{i=0}^{k-1} b_i^{\text{hor}} - \epsilon, \quad 1 \leq i \leq N_a - 1, \quad i + 1 \leq \ell \leq N_a, \quad 1 \leq n \leq N_{\text{cs}}, \quad 0 \leq k \leq T, \end{aligned}$$



$$\begin{aligned} \sum_{n=1}^{N_{cs}} b_{i,\ell,n,k}^{\text{col}} &\geq 1, \quad 1 \leq i \leq N_a - 1, \quad i + 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \\ b_{i,\ell,n,k}^{\text{col}} &= 0, \quad 1 \leq i \leq N_a, \quad 1 \leq \ell \leq i, \quad 1 \leq n \leq N_{cs}, \quad 0 \leq k \leq T, \end{aligned} \quad (8)$$

where each  $b_{i,\ell,n,k}^{\text{col}}$  is a binary variable employed for the interagent collision avoidance mechanism,  $(\mathbf{p}_n^{\text{col}})^T$  is the  $n$ th row of  $P^{\text{col}}$ ,  $q_n^{\text{col}}$  is the  $n$ th element of  $\mathbf{q}^{\text{col}}$ , and  $\epsilon$  has the same role as in (7). The idea here is very similar to the one used for obstacle avoidance, with the remark that, since the safe region around each agent is assumed to be the same, if agent  $i$  does not collide with agent  $\ell$ , this pair does not need to be considered twice, which allows the reduction in the number of binary variables expressed by the last equality in (8).

## 2.4 | Multiagent trajectory planning with connectivity assurance

The problem to be solved in the present article is that of guiding one of  $N_a$  agents (which one of the agents is also an optimization variable) to a fixed terminal target set, collecting rewards when one of the agents passes through one of the other  $N_t - 1$  intermediate targets (if possible and profitable, that is, if it is feasible and if the time and fuel consumption to pass through an intermediate set is compensated by the reward it offers), while maintaining connectivity of the graph representing the communication network of the agents, avoiding collisions between agents and the  $N_o$  obstacles and between pairs of agents. As such, the proposal within this article allows complete flexibility, since there is no preassignment of an agent to a target set nor a leader is chosen beforehand, rather these choices are made during the optimization to cope with the constraints and achieve the minimal cost. Therefore, this formulation can be regarded as a strategy with no leader, the agents being bound only by collision avoidance and communication ranges around each of them. The cost function that encompasses these objectives is:

$$J = \sum_{k=0}^T k b_k^{\text{hor}} + \gamma \sum_{\ell=1}^{N_a} \sum_{k=0}^T \|\mathbf{u}_{\ell,k}\|_1 - R \sum_{m=1}^{N_t} \sum_{\ell=1}^{N_a} \sum_{k=0}^T b_{\ell,m,k}^{\text{target}} \quad (9)$$

The agents are assumed to have a certain communication range, which is common for all agents, represented as a polyhedron with  $N_{sc}$  faces:  $\{P^{\text{con}}\mathbf{C}\mathbf{x} \leq \mathbf{q}^{\text{con}}\}$ . Therefore, the pair of agents  $i$  and  $\ell$  is considered connected at instant  $k$  if  $\{P^{\text{con}}\mathbf{C}(\mathbf{x}_{i,k} - \mathbf{x}_{\ell,k}) \leq \mathbf{q}^{\text{con}}\}$ . Considering symmetric polygons with respect to the origin allows the conclusion  $\{P^{\text{con}}\mathbf{C}(\mathbf{x}_{i,k} - \mathbf{x}_{\ell,k}) \leq \mathbf{q}^{\text{con}}\} \Leftrightarrow \{P^{\text{con}}\mathbf{C}(\mathbf{x}_{\ell,k} - \mathbf{x}_{i,k}) \leq \mathbf{q}^{\text{con}}\}$ . Minor changes must be made to deal with the case where the polyhedron is not symmetric. In particular, in the present article the communication ranges will be defined by the bounds on the absolute values of the distances in each axis, namely,  $x_{c,\max}$  and  $y_{c,\max}$ , resulting in a rectangular communication range with sides parallel to the  $x$  and  $y$  axes. Any other form that results in a polyhedron could be used, therefore this presents no restriction of the proposal. In fact, even scenarios where the communication range may be better described as a circle can be coped with by inscribing a polygon in the circle. The enforcement of the polygonal constraints would ensure that the circle communication connectivity constraints are also respected. The amount of conservatism introduced by the approximation of the circle by a polygon can be reduced by increasing the number of sides of the polygon, at the cost of an increase in the number of constraints.

We remark that the adjacency matrix does not penalize the distances between the agents in our formulation. Penalizing the distances between agents might steer the optimal solution toward keeping the agents within communication range of one another, although no guarantee of connectivity arises by solely doing this without the connectivity constraints. On the other hand, weights depending on the distance between agents might lead to a more cumbersome optimization problem, as the weights in the adjacency matrix themselves would be optimization variables. For example, if one intends to use the euclidean distance, the relation between the weights and the positions of the agents would be nonlinear.

In order to impose the constraints over a horizon of at most  $T + 1$  time steps,  $T + 1$  binary variables are necessary. Moreover, constraints must be added to the classical maneuvering optimization Problem 1, resulting in Problem 2.

### Problem 2.

$$\min_{\mathbf{x}_{\ell,k}, \mathbf{u}_{\ell,k}, b_k^{\text{hor}}, b_k^{\text{obs}}, b_{\ell,j,n,k}^{\text{col}}, b_{i,\ell,n,k}^{\text{col}}, b_{\ell,m,k}^{\text{target}}, b_{i,\ell,k}^{\text{con}}} J, \quad (10)$$

subject to

$$\mathbf{x}_{\ell,0} = \mathbf{x}_{\ell,\text{initial}}, \quad 1 \leq \ell \leq N_a, \quad (11a)$$

$$\mathbf{x}_{\ell,k+1} \leq \mathbf{A}\mathbf{x}_{\ell,k} + \mathbf{B}\mathbf{u}_{\ell,k} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \quad 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11b)$$

$$-\mathbf{x}_{\ell,k+1} \leq -\mathbf{A}\mathbf{x}_{\ell,k} - \mathbf{B}\mathbf{u}_{\ell,k} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \quad 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11c)$$

$$-\mathbf{x}_{\ell,k+1} \leq -\mathbf{x}_{\min} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \quad 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11d)$$

$$\mathbf{x}_{\ell,k+1} \leq \mathbf{x}_{\max} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \quad 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11e)$$

$$-\mathbf{u}_{\ell,k} \leq -\mathbf{u}_{\min} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \quad 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11f)$$

$$\mathbf{u}_{\ell,k} \leq \mathbf{u}_{\max} + M \sum_{i=0}^{k-1} b_i^{\text{hor}}, \quad 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11g)$$

$$-(\mathbf{p}_{j,n}^{\text{obs}})^T \begin{bmatrix} \mathbf{x}_{\ell,k+1} \\ Y_{\ell,k+1} \end{bmatrix} \leq -q_{j,n}^{\text{obs}} + M(1 - b_{\ell,j,n,k}^{\text{obs}}) + M \sum_{i=0}^{k-1} b_i^{\text{hor}} - \epsilon, \quad 1 \leq \ell \leq N_a, \quad 1 \leq j \leq N_o, \quad 1 \leq n \leq N_{\text{os}}, \quad 0 \leq k \leq T, \quad (11h)$$

$$-(\mathbf{p}_{j,n}^{\text{obs}})^T \begin{bmatrix} \mathbf{x}_{\ell,k} \\ Y_{\ell,k} \end{bmatrix} \leq -q_{j,n}^{\text{obs}} + M(1 - b_{\ell,j,n,k}^{\text{obs}}) + M \sum_{i=0}^{k-1} b_i^{\text{hor}} - \epsilon, \quad 1 \leq \ell \leq N_a, \quad 1 \leq j \leq N_o, \quad 1 \leq n \leq N_{\text{os}}, \quad 0 \leq k \leq T, \quad (11i)$$

$$-(\mathbf{p}_n^{\text{col}})^T \begin{bmatrix} \mathbf{x}_{i,k+1} - \mathbf{x}_{\ell,k+1} \\ Y_{i,k+1} - Y_{\ell,k+1} \end{bmatrix} \leq -q_n^{\text{col}} + M(1 - b_{i,\ell,n,k}^{\text{col}}) + M \sum_{i=0}^{k-1} b_i^{\text{hor}} - \epsilon, \quad 1 \leq i \leq N_a - 1, \quad i + 1 \leq \ell \leq N_a, \quad 1 \leq n \leq N_{\text{cs}}, \quad 0 \leq k \leq T, \quad (11j)$$

$$P_m^{\text{target}} \begin{bmatrix} \mathbf{x}_{\ell,k+1} \\ Y_{\ell,k+1} \end{bmatrix} \leq \mathbf{q}_m^{\text{target}} + M(1 - b_{\ell,m,k}^{\text{target}}) \mathbf{1}_{N_s}, \quad 1 \leq \ell \leq N_a, \quad 1 \leq m \leq N_t, \quad 0 \leq k \leq T, \quad (11k)$$

$$P^{\text{con}} \begin{bmatrix} \mathbf{x}_{i,k+1} - \mathbf{x}_{\ell,k+1} \\ Y_{i,k+1} - Y_{\ell,k+1} \end{bmatrix} \leq \mathbf{q}^{\text{con}} + M(1 - b_{i,\ell,k}^{\text{con}}) \mathbf{1}_{N_{\text{sc}}} + M \sum_{j=0}^{k-1} b_j^{\text{hor}}, \quad 1 \leq i \leq N_a - 1, \quad i + 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11l)$$

$$\sum_{k=0}^T b_k^{\text{hor}} = 1, \quad (11m)$$

$$\sum_{\ell=1}^{N_a} \sum_{k=0}^T b_{\ell,m,k}^{\text{target}} \leq 1, \quad 1 \leq m \leq N_t, \quad (11n)$$

$$\sum_{k=0}^T k(b_{\ell,m,k}^{\text{target}} - b_k^{\text{hor}}) \leq 0, \quad 1 \leq \ell \leq N_a, \quad 1 \leq m \leq N_t, \quad (11o)$$

$$\sum_{\ell=1}^{N_a} b_{\ell,N_t,k}^{\text{target}} - b_k^{\text{hor}} = 0, \quad 0 \leq k \leq T, \quad (11p)$$

$$\sum_{n=1}^{N_{\text{os}}} b_{\ell,j,n,k}^{\text{obs}} \geq 1, \quad 1 \leq \ell \leq N_a, \quad 1 \leq j \leq N_o, \quad 0 \leq k \leq T, \quad (11q)$$

$$\sum_{n=1}^{N_{\text{cs}}} b_{i,\ell,n,k}^{\text{col}} \geq 1, \quad 1 \leq i \leq N_a - 1, \quad i + 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (11r)$$

$$b_{i,\ell,n,k}^{\text{col}} = 0, \quad 1 \leq i \leq N_a, \quad 1 \leq \ell \leq i, \quad 1 \leq n \leq N_{\text{cs}}, \quad 0 \leq k \leq T, \quad (11s)$$

$$\sum_{i=1}^{N_a-1} \sum_{\ell=i+1}^{N_a} b_{i,\ell,k}^{\text{con}} = N_a - 1, \quad 0 \leq k \leq T, \quad (11t)$$

$$\sum_{v_i, v_j \in S} b_{i,\ell,k}^{\text{con}} \leq |S| - 1, \quad 0 \leq k \leq T, \quad \forall S \subset V, \quad (11u)$$

$$b_{i,\ell,k}^{\text{con}} = 0, \quad 1 \leq i \leq N_a, \quad 1 \leq \ell \leq i, \quad 0 \leq k \leq T. \quad (11v)$$

For the sake of completeness, all the constraints involved in the proposed formulation are presented in Problem 2; however, we will focus our discussion here on the ones that were added to the problem in the present article. Constraints (11h) impose obstacle avoidance, while (11i) enforce the corner cutting avoidance, as described in Richards and Turnbull (2015).<sup>19</sup> One of the complements of the half-planes defining an obstacle side must be occupied by the agents at every sample time. This is ensured by (11q), in which a small difference regarding the encoding in Schouwenaars et al (2001)<sup>16</sup> and Richards and How (2006)<sup>15</sup> can be noticed: in the former papers, the choice is to consider the binary variable equal to one when the avoidance constraint is relaxed by the “big-M” procedure and zero otherwise, as opposed to the present formulation, where the binary variable is one when the corresponding constraint is imposed and zero in case it is relaxed. This small difference does not imply any negative consequence, if constraints (11h), (11i), and (11q) are modified accordingly, as was done.

Collision between each pair of agents is prevented by constraints (11j), which are similar to obstacle avoidance constraints. Therefore, at least one half-plane associated with the faces of the agent polyhedron must be imposed, which is done by (11r). Again, the same remark highlighted in the previous paragraph is valid: the choice for the encoding is opposed to the one in Schouwenaars et al (2001),<sup>16</sup> but the constraints are accordingly modified and the change is immaterial to the results.

A certain number of target sets may be visited; therefore, there are decision variables related to the choice of visiting a target set or not. These are associated with rewards that diminish the cost function and may only be collected if one of the agents occupies a position within the associated target set at one instant, as imposed in (11k). The decision binary variables are constrained as in (11n) to allow collection of the rewards only once per target. Moreover, the optimization must be considered up to the instant when the last target set is visited. This is ensured by (11o), which imposes that the optimization horizon is greater or equal than all the instants of visitation of the target sets. Moreover, (11p) imposes that the last target set in the list of possible target sets should be the last visited. The maneuver is guaranteed to finish within the maximal horizon  $T + 1$  by the constraint in (11m). This last feature allows a maneuver with a goal end position and consideration of additional opportunities of visiting other intermediate targets if convenient.

The main contribution of this work is related to (11l). This constraint, associated with (11t), (11u), and (11v), imposes that each agent must be within the communication range of at least one other agent and that no pair needs to be considered twice (meaning that, if it was imposed that agent  $i$  is already considered in the communication range of agent  $j$ , than agent

$j$  is not again imposed to be within the communication range of agent  $i$ ). Note that a polyhedral communication area must be considered. Despite a circular communication area being a more natural model, this is avoided as there would be a need to include quadratic constraints, which would make the optimization problem more challenging. In practice, the communication polyhedron may have as many faces as desired, therefore an arbitrarily precise approximation to a circle may be considered at the cost of adding inequalities to the optimization problem. For illustration,  $N_{os} = 4$  was considered in the simulations in the present article.

These constraints also impose that the graph must be connected. The equalities in (11v) encode the fact that the connectivity is bidirectional; therefore, there is no need to reconsider the terms  $b_{i,j,k}^{con}$  in the lower left triangles of the adjacency matrices (recall that the communication graph is time-varying, so we must have one adjacency matrix per time step), actually reducing the number of necessary binary variables.

For each time step  $k$ , Theorem 2 formally attests that the constraints (11t) and (11u) only permit generation of connected graphs. Moreover, all possible connected graphs are taken into account, as discussed later. First of all, we may say that these constraints are selecting a spanning tree for the communication graph.

**Theorem 2.** *At time step  $k$ , satisfaction of constraints (11t) and (11u) guarantees that the communication graph is connected.*

*Proof.* Theorem 1 imposes two conditions for a graph  $G = (V, E)$  to be a tree:  $|E| = |V| - 1$  and no cycles must be present. Constraints (11t) directly impose the first condition. The second one is guaranteed by constraints (11u), which are called *subtour elimination constraints* in the literature.<sup>30</sup> These constraints prevent the existence of cycles for each induced subgraph, therefore inexistence of cycles for the graph follows.

In fact, assume, for the sake of contradiction, that a cycle exists, then one could consider the induced subgraph  $G(S) = (S, E(S))$ , where  $S$  is the set of the vertices in this cycle. For this cycle, we have  $|S|$  vertices, so counting the edges due to the cycle entails at least  $|S|$  edges, that is,  $|E(S)| \geq |S|$  (edges other than the ones in the cycle may also be present) and constraints (11u) do not hold, which is a contradiction. Therefore, the communication graph imposed by constraints (11t) and (11u) must be a tree. Since a tree is connected by definition, the connectedness of the communication graph follows. ■

To verify that the tree imposed by these constraints is really a spanning tree of a less restricted connected communication graph which may be chosen by the optimization, observe that the “big-M” technique used in (11l) works by imposing the constraint if the edge is present in the adjacency matrix, but simply relaxing it otherwise. In other words, the existence of an edge constrains the two agents to be within communication range; however, the inexistence of it leaves the optimization free to decide whether these agents should be placed within communication range or not. Hence, the actual communication graph may have more links than the ones seen by constraints (11t) and (11u), which is actually choosing a spanning tree. Since these constraints are able to generate any spanning tree and no constraints are imposed to the other (redundant) edges of the graph, the optimization is free to select the most convenient connected graph.

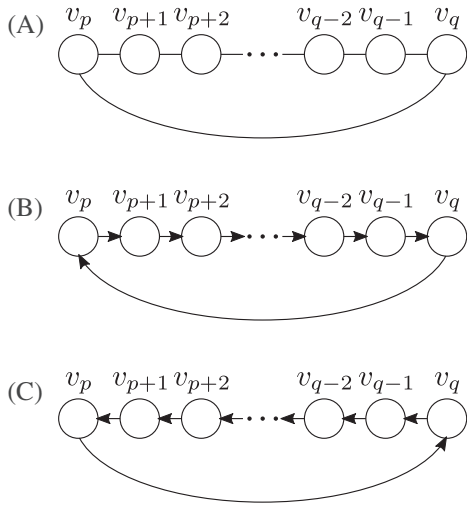
On the other hand, the amount of constraints introduced by (11u) may limit the applicability of the proposal in this article due to scalability issues. Since each induced subgraph must be considered, a priori we would need  $2^{N_a}$  of these constraints. In practice, some of these cases do not make sense or trivially hold. For  $|S| = 0$  and  $|S| = 1$ , no edges are present. For  $|S| = N_a$ , constraint (11t) is already a more restricted version. For  $|S| = 2$ , the constraints reduce to:

$$b_{i,\ell,k}^{con} \leq 1, \quad 1 \leq i \leq N_a - 1, \quad i + 1 \leq \ell \leq N_a, \quad 0 \leq k \leq T, \quad (12)$$

which are trivially true given that  $b_{i,\ell,k}^{con}$  are binary variables. Thence, (11u) actually introduces  $(T + 1)(2^{N_a} - N_a - 2 - N_a(N_a - 1)/2)$  constraints, that is still  $O(T2^{N_a})$ , that is, the number of constraints increases exponentially with the number of agents, which is theoretically undesirable for scalability. In practice, in terms of optimization time, this is not much of a downsize, as our simulation results support. On the other hand, if one considers memory usage, a larger number of inequalities may pose stringent requirements on hardware.

## 2.5 | More restrictive formulation using fewer constraints

To cope with scalability issues in the number of constraints, an alternative heuristic formulation is presented, which requires less constraints and is shown to always produce a connected graph (though possibly suboptimal). Consider



**FIGURE 1** The loop considered in the Proof of Theorem 3: A, loop with undirected edges, B, loop representing the assignment where  $b_{q,p}^{\text{con}} = 1$ , and C, loop representing the assignment where  $b_{p,q}^{\text{con}} = 1$

replacing the constraints (11t) and (11u) with the following alternative constraints:

$$\sum_{\ell=i+1}^{N_a} b_{i,\ell,k}^{\text{con}} = 1, \quad 1 \leq i \leq N_a - 1, \quad 0 \leq k \leq T, \quad (13)$$

which impose that a single edge exists in each row of the adjacency matrix, except for the last one, which must have none. In this case, we have merely  $(T + 1)(N_a - 1) = O(TN_a)$  constraints, thus the optimization problem scales much better. The graphs generated by this constraint are also connected, in fact they are also spanning trees as before. This fact is formally stated as Theorem 3.

To better grasp Theorem 3 and other following discussions, let us introduce an intuitive representation of the possible assignments for binary variables  $b_{i,\ell,k}^{\text{con}}$ . For a given time step  $k$ , we may associate an adjacency matrix to these binary variables. Due to constraint (11v), no self-loops exist and all elements below the main diagonal are zero. Therefore, we may think of possible assignments for  $b_{i,\ell,k}^{\text{con}}$  as building a directed graph where  $b_{i,\ell,k}^{\text{con}} = 1$  means the directed edge  $(v_i, v_\ell)$  exists.

**Theorem 3.** *The graphs whose adjacency matrices respect constraints (11v) and (13) are trees.*

*Proof.* Consider any of these such graphs  $G = (V, E)$  with  $|V| = N_a$ . First, since the constraint guarantees that each row of the adjacency matrix has exactly one active column, except for the last none, which has none, and we have  $N_a - 1$  rows, the graph has  $N_a - 1$  edges. Let us assume, by absurd, that cycles exist in the graph. Without loss of generality, let us pick one of these cycles such that its set of edges is:  $\{(v_p, v_{p+1}), (v_{p+1}, v_{p+2}), \dots, (v_{q-2}, v_{q-1}), (v_{q-1}, v_q), (v_q, v_p)\}$ . Figure 1A shows this loop. Considering the edge  $(v_q, v_p)$  of this undirected graph, there are two possible assignments for the adjacency matrix:  $b_{q,p}^{\text{con}} = 1$  or  $b_{p,q}^{\text{con}} = 1$  (where we dropped the time step index for conciseness). Due to (11v),  $b_{i,j}^{\text{con}} = 0, \forall j \leq i$ , thus  $b_{q,p}^{\text{con}} = 1$  and  $b_{p,q}^{\text{con}} = 1$  imply  $q < p$  and  $p < q$ , respectively. Let us consider these two cases separately.

If  $b_{q,p}^{\text{con}} = 1$ , we must have  $q < p$ . Since no other column in row  $q$  may be active due to (13), we necessarily have  $b_{q-1,q}^{\text{con}} = 1$  for edge  $(v_{q-1}, v_q)$  to exist. Analogously, we have  $b_{q-2,q-1}^{\text{con}} = 1, \dots, b_{p+1,p+2}^{\text{con}} = 1, b_{p,p+1}^{\text{con}} = 1$ , which impose the following numbering between vertices (due to (11v)):  $p < p + 1 < \dots < q - 1 < q$ . This situation is illustrated in Figure 1B. Thence,  $q < p$  and  $p < q$  creates a contradiction.

The case where  $b_{p,q}^{\text{con}} = 1$  and  $p < q$  is analogous. We must have  $b_{p+1,p}^{\text{con}} = 1$  for edge  $(v_p, v_{p+1})$  to exist. Propagating this, we obtain  $b_{p+2,p+1}^{\text{con}} = 1, \dots, b_{q-1,q-2}^{\text{con}} = 1, b_{q,q-1}^{\text{con}} = 1$  (see Figure 1C), then  $q < q - 1 < \dots < p + 1 < p$  and the same contradiction arises.

Therefore, the graph cannot have any cycle. Since the graph has  $|E| = |V| - 1$  edges and no cycles, by applying Theorem 1, we prove that this graph is a tree. ■

Still, the not obvious downside is that this new formulation is suboptimal since the alternative set of constraints does not generate some spanning trees. To verify this, we will show a counterexample. If  $N_a = 3, V = \{v_1, v_2, v_3\}$  and we have three possible spanning trees:

1.  $G_1 = (V, E_1)$  with  $E_1 = \{(v_1, v_2), (v_2, v_3)\}$ .
2.  $G_2 = (V, E_2)$  with  $E_2 = \{(v_1, v_3), (v_2, v_3)\}$ .
3.  $G_3 = (V, E_3)$  with  $E_3 = \{(v_1, v_2), (v_1, v_3)\}$ .

However, the spanning tree  $G_3$  violates constraint (13) and will not be taken into account by the optimization. To see this, consider the adjacency matrix of this graph:

$$(b_{i,\ell,*}^{\text{con}}) = \begin{bmatrix} * & 1 & 1 \\ * & * & 0 \\ * & * & * \end{bmatrix}, \quad (14)$$

where the symbol \* was used to denote values that do not matter. The sum of the first row yields 2 in this example, then constraint (13) does not hold. In general, this comes from the fact that agents with a higher index have the privilege of being able to have a higher degree in this formulation. Consider the first agent, the only edge this agent may be in the chosen spanning tree comes from the first row, so the maximum degree for this agent is 1. For agent 2, the edges may come from rows 1 and 2, then the maximum degree is 2 in this case, and so on. Therefore, the  $i$ th agent has a maximum degree of  $i$ , except for the last one, which has the same maximum degree as the penultimate, that is,  $N_a - 1$ .

Despite this theoretical limitation, in simulation experiments, if the agents' numbering is initially selected so an initial spanning tree respecting constraint (13) exists, we have observed that this heuristic formulation is able to generate multiagent trajectories very close in terms of optimality to the ones provided by the more complex optimization problem resulting from constraints (11t) and (11u). Indeed, if the communication graph is initially connected, it is always possible to enumerate the agents respecting constraint (13). In the following, Algorithm 1 is proposed to do this numbering and Theorem 4 states that the resulting numbering yields feasibility of constraints (11v) and (13) whenever a connected graph exists.

First, consider the initial connected communication graph. If this graph is not known beforehand, we may construct it by checking if each pair of agents are within communication range of each other, which takes  $O(N_a^2)$  time. As a connected graph, it admits at least one spanning tree. Let us pick any of these spanning trees. A spanning tree may be easily obtained by executing a breadth-first search (BFS) or a depth-first search (DFS), which are known to construct a tree<sup>22</sup> and have computational complexity of  $O(N_a^2)$  if an adjacency matrix is used to represent the graph.

Then, recall that in a tree each node has exactly one parent. Therefore, directing each edge of this tree from node to parent, it appears that an assignment respecting constraint (13) exists. We still need to take into account the fact that agents with lower numbers have a lower degree limit. Since the first agent has a degree limit of 1, we must select a leaf for it. In this sense, our intuition says that we should enumerate leaves first, then go on enumerating the parents of these leaves, and so on. In fact, enumerating the agents in postorder<sup>31</sup> results in the required numbering, which needs  $O(N_a)$  time. The pseudocode for a postordering of a tree  $T_G$  with initial index  $i$  is presented in Algorithm 1.

---

#### Algorithm 1. Postorder tree vertex ordering

---

```

function POSTORDER( $T_G, L, i$ )
   $r \leftarrow T_G.$ RootNode ▷ root node of the tree
  for Child  $\in$  Children( $r$ ) do
     $T'_G \leftarrow$  SubTree( $T_G, \text{Child}$ ) ▷ returns the subtree with root node Child
    ( $L, i$ )  $\leftarrow$  PostOrder( $T'_G, L, i$ )
  end for
  if  $i = 1$  then
     $L \leftarrow (r, i)$ 
  else
     $L \leftarrow$  append( $L, (r, i)$ ) ▷ returns the ordered list  $L$  with the pair  $(r, i)$  added at the end
  end if
   $i \leftarrow i + 1$ 
  return ( $L, i$ )
end function

```

---

Moreover, the following theorem states that the resulting numbering from Algorithm 1 ensures feasibility of constraints (11v) and (13).

**Theorem 4.** *If an initially connected graph exists, then labeling according to Algorithm 1 results in feasibility of constraints (11v) and (13).*

*Proof.* If the graph is initially connected, then in view of Remark 1 it has a spanning tree; therefore, its nodes can be labeled using Algorithm 1, and it is assumed in the following that such a labeling has been done. Let us consider a node labeled with  $i$  in the spanning tree and its  $c_i$  children labeled as  $\ell_{i,1}, \ell_{i,2}, \dots, \ell_{i,c_i}$ .

A possible choice of representation of the adjacency matrix at instant  $k$  is to set the binary variables representing the connection between each node  $\ell_{ij}$  and its parent  $i$  to one in row  $\ell_{ij}$ , that is,

$$b_{\ell_{ij},i,k}^{\text{con}} = 1, \quad (15)$$

and set the remaining as

$$b_{\ell_{ij},m,k}^{\text{con}} = 0, \quad m \neq i. \quad (16)$$

By applying Algorithm 1 to label the nodes of this tree, it is clear that the label of a parent is greater than the label of all of its children, then it is true that:

$$i \geq \ell_{ij}, \quad j = 1, 2, \dots, c_i. \quad (17)$$

In view of (15), (16), and (17), both (11v) and (13) are trivially satisfied for rows  $\ell_{ij}$  of the adjacency matrix.

This constructive procedure can be applied to each level of the tree, entailing that constraints (11v) and (13) are respected, filling the adjacency matrix up until the level of the root of the tree. The label attributed by Algorithm 1 to the root is  $N_a$  and since the root has no parent

$$b_{N_a,m,k}^{\text{con}} = 0, \quad m = 1, 2, \dots, N_a, \quad (18)$$

which enforces (11v). ■

Algorithm 2 summarizes the procedure presented here to enumerate the agents in a way that constraint (13) is true (the algorithm assumes DFS, but BFS may be used as well). Considering the computational complexity of each of its parts, the overall procedure takes  $O(N_a^2)$  time, which is already the lowest time complexity we may achieve, given that we have to first build the connected graph by checking each pair of agents.

---

**Algorithm 2.** Enumerates agents so the heuristic formulation may be used

---

**function** ENUMERATEAGENTS( $\{\mathbf{x}_{i,0}\}_{i \in \{1, \dots, N_a\}}$ )

$V \leftarrow \{1, \dots, N_a\}$

$E \leftarrow \emptyset$

**for**  $i = 1, \dots, N_a - 1$  **do**

**for**  $\ell = i + 1, \dots, N_a$  **do**

**if**  $|\mathbf{x}_{i,0} - \mathbf{x}_{\ell,0}| \leq \mathbf{x}_{c,\max}$  and  $|Y_{i,0} - Y_{\ell,0}| \leq Y_{c,\max}$  **then**

$E \leftarrow E \cup \{(i, \ell)\}$

**end if**

**end for**

**end for**

$G \leftarrow (V, E)$

$T_G \leftarrow \mathbf{DFS}(G)$

$L \leftarrow \mathbf{PostOrder}(T_G, \emptyset, 1)$

$\triangleright T_G$  is the tree built by DFS

$\triangleright L$  maps nodes to their postorder numbering and output \* is not used

**end function**

---

### 3 | SIMULATION RESULTS

Three scenarios are compared in the following subsection to illustrate the use of the proposed approaches: (i) the proposal in Problem 2, (ii) the simplification to Problem 2 proposed in Subsection 2.5, and (iii) the imposition of a totally connected graph in the multiagent problem without using the proposed connectivity assurance method based on writing the adjacency matrix with binary optimization variables. In this last approach, to ensure connectivity, each agent is constrained to be within the polyhedron that represents the communication range of each of the  $N_a - 1$  other agents.

The adopted conditions for the simulations are presented in Table 1. In this table, the obstacles have  $N_{os} = 4$  facets, the safe regions around the agents have  $N_{cs} = 4$  facets, and the polyhedra representing the communication ranges have  $N_{sc} = 4$  facets, each with the projection of each facet parallel either to the  $x$ - or  $y$ -axis; therefore, they are represented by the upper and lower limits on  $x$  and  $y$ . We stress that the representation of the safety regions and communication ranges as squares in these examples do not represent a limitation of the proposed approach, as other polygons with more sides could be used. It is rather an arbitrary choice for illustration. For other more general nonpolygonal safety regions or communication ranges, for example, circles, inner polygonal approximations of arbitrary accuracy are enabled by our proposal.

The software Matlab was used for the simulations, with CPLEX as the optimizer in a computer with an Intel Core i7 6700K processor (4.0 GHz clock) and 24GB RAM.

In all figures depicting trajectories and the associated connectivity graphs in the illustrative examples the following convention is used: the obstacles are depicted as black-filled polygons, the target sets are white-filled polygons with black borders and contain a roman number inside corresponding to the order in Table 1, the trajectories of each agent are depicted with a different color (with the corresponding agent id depicted in the legend), with their positions depicted as crosses ( $\times$ ). The communication range of each agent is depicted as a semitransparent square with sides of 2 length units compatible with the communication range presented in Table 1 and of the same color as the respective agent; however, these are just depicted at the instants of visitation of the target sets, since drawing them throughout the whole trajectory would clutter the figures excessively. Moreover, the connectivity graphs are not depicted at every instant, rather only at instants in which at least one target set is visited, as to avoid showing excessive information. In these cases, dark gray lines are used to represent a connection between agents at that instant.

#### 3.1 | Comparison of the results in the three simulation scenarios

In employing a numerical optimizer, one may define a maximal time for returning the feasible solution with the least cost at the moment. Figure 2A shows the results with a maximal time of 20 seconds for the proposed Problem 2. As can be seen, the agents move toward the fixed ultimate goal (target set 5). In their trajectory, two other target sets are visited. The connectivity graph of the agents is depicted in Figure 2B,C only at the instants of visitation of the targets, although the graph always remains connected, since showing it at every instant would demand excessive space. It can be noticed that the connectivity is maintained and, in fact, the flexibility of the positions of the agents is used to visit two target sets at once, while maintaining the connectivity.

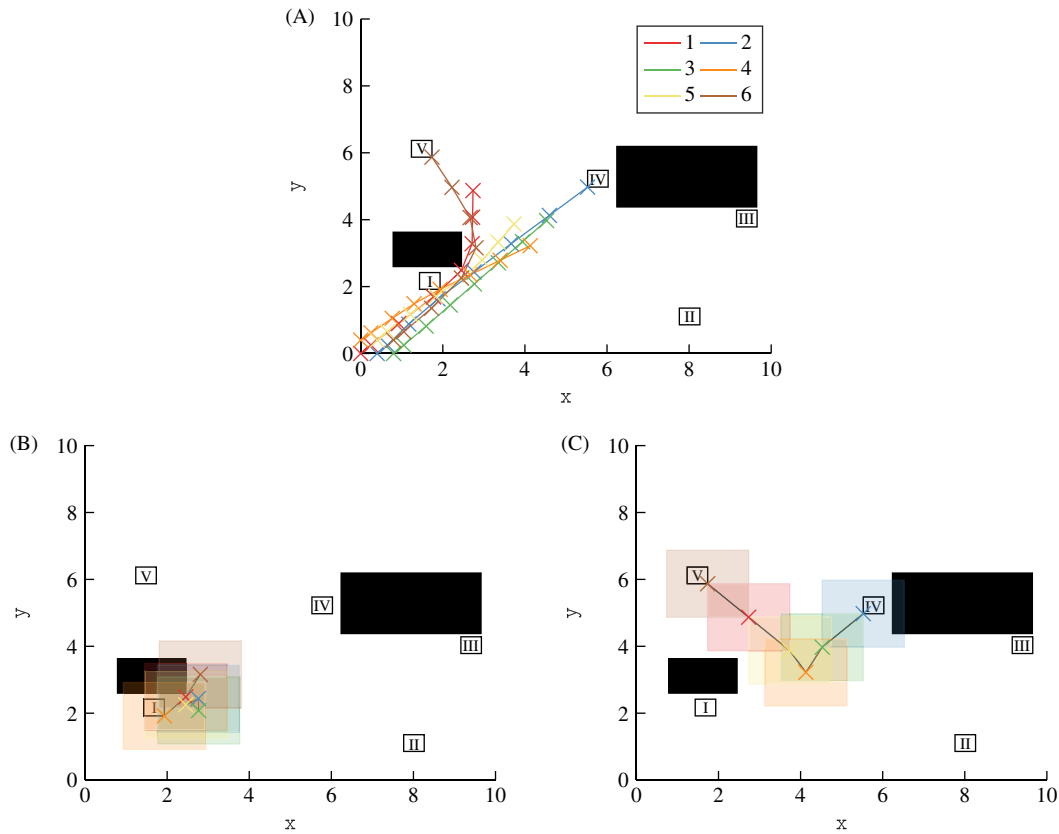
To provide insight into the proposed formulation, the connectivity matrices at the instants of visitation of the target sets are presented:

$$(b_{i,\ell,A}^{\text{con}}) = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (19)$$



TABLE 1 Simulation parameters

Parameter name	Symbol	Values	
Maximal horizon	$T$	20	
Number of agents	$N_a$	6	
Number of targets	$N_t$	5	
Number of obstacles	$N_o$	2	
Fuel weight	$\gamma$	0.1	
Reward per target	$R$	10	
“Big-M”	$M$	100	
Initial states	$\mathbf{x}_{1,\text{initial}}$	$[0\ 0\ 0\ 0]^T$	
	$\mathbf{x}_{2,\text{initial}}$	$[0.4\ 0\ 0\ 0]^T$	
	$\mathbf{x}_{3,\text{initial}}$	$[0.8\ 0\ 0\ 0]^T$	
	$\mathbf{x}_{4,\text{initial}}$	$[0\ 0\ 0.4\ 0]^T$	
	$\mathbf{x}_{5,\text{initial}}$	$[0.4\ 0\ 0.4\ 0]^T$	
	$\mathbf{x}_{6,\text{initial}}$	$[0.8\ 0\ 0.4\ 0]^T$	
State constraints	$\mathbf{x}_{\text{max}}$	$[10\ 1\ 10\ 1]^T$	
	$\mathbf{x}_{\text{min}}$	$[0\ -1\ 0\ -1]^T$	
Control constraints	$\mathbf{u}_{\text{max}}$	$[0.5\ 0.5]^T$	
	$\mathbf{u}_{\text{min}}$	$[-0.5\ -0.5]^T$	
Target sets	$[\mathbf{x}_{tI,\text{min}}\ \mathbf{x}_{tI,\text{max}}]$	$[1.4327\ 1.9327]$	
	$[\mathbf{y}_{tI,\text{min}}\ \mathbf{y}_{tI,\text{max}}]$	$[1.9192\ 2.4192]$	
	$[\mathbf{x}_{tII,\text{min}}\ \mathbf{x}_{tII,\text{max}}]$	$[7.7625\ 8.2625]$	
	$[\mathbf{y}_{tII,\text{min}}\ \mathbf{y}_{tII,\text{max}}]$	$[0.8571\ 1.3571]$	
	$[\mathbf{x}_{tIII,\text{min}}\ \mathbf{x}_{tIII,\text{max}}]$	$[9.1561\ 9.6561]$	
	$[\mathbf{y}_{tIII,\text{min}}\ \mathbf{y}_{tIII,\text{max}}]$	$[3.7841\ 4.2841]$	
	$[\mathbf{x}_{tIV,\text{min}}\ \mathbf{x}_{tIV,\text{max}}]$	$[5.5272\ 6.0272]$	
	$[\mathbf{y}_{tIV,\text{min}}\ \mathbf{y}_{tIV,\text{max}}]$	$[4.9745\ 5.4745]$	
	$[\mathbf{x}_{tV,\text{min}}\ \mathbf{x}_{tV,\text{max}}]$	$[1.2356\ 1.7356]$	
	$[\mathbf{y}_{tV,\text{min}}\ \mathbf{y}_{tV,\text{max}}]$	$[5.8694\ 6.3694]$	
	Obstacles	$[\mathbf{x}_{o1,\text{min}}\ \mathbf{x}_{o1,\text{max}}]$	$[0.7905\ 2.4531]$
		$[\mathbf{y}_{o1,\text{min}}\ \mathbf{y}_{o1,\text{max}}]$	$[2.5992\ 3.6251]$
$[\mathbf{x}_{o2,\text{min}}\ \mathbf{x}_{o2,\text{max}}]$		$[6.2389\ 9.6468]$	
$[\mathbf{y}_{o2,\text{min}}\ \mathbf{y}_{o2,\text{max}}]$		$[4.3813\ 6.1894]$	
Communication range	$\mathbf{x}_{c,\text{max}}$	1	
	$\mathbf{y}_{c,\text{max}}$	1	
Agent safe region	$[\mathbf{x}_{a,\text{min}}\ \mathbf{x}_{a,\text{max}}]$	$[-0.05\ 0.05]$	
	$[\mathbf{y}_{a,\text{min}}\ \mathbf{y}_{a,\text{max}}]$	$[-0.05\ 0.05]$	



**FIGURE 2** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting three target sets (white-filled squares). B, Connectivity graph at the instant target sets I and, C, IV and V are visited. Maximal time for obtaining this solution is 20 seconds with Problem 2. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

and

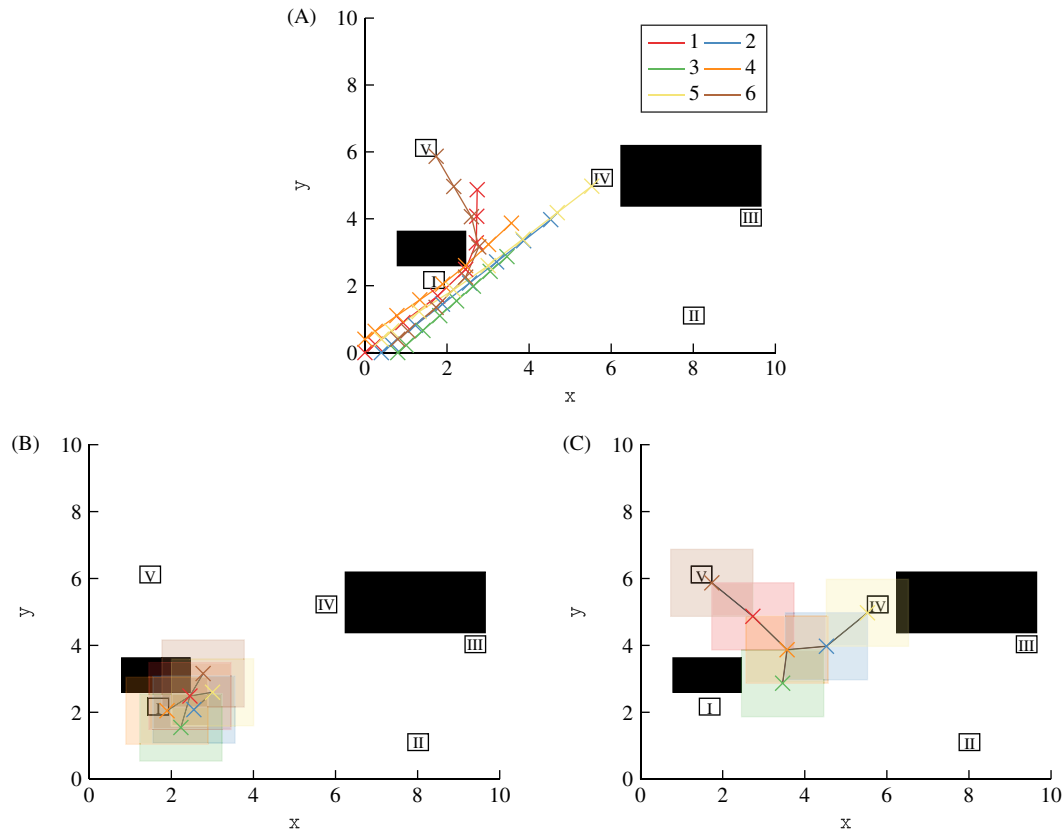
$$(b_{i,\ell,7}^{\text{con}}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (20)$$

The configurations in the connectivity matrices reflect the graphs in Figure 2B,C. Notice that the matrix in (19) corresponds to a graph in which agent number 1 (red trajectory) is connected to three others, since the terms  $b_{1,3,4}^{\text{con}} = b_{1,4,4}^{\text{con}} = b_{1,6,4}^{\text{con}} = 1$ , a situation that could not be achieved with the modified version of Problem 2. On the other hand,  $(b_{i,\ell,7}^{\text{con}})$  corresponds to a graph in which no agent has degree greater than two.

As the optimization time is increased to 40 seconds, Figure 3A depicts the ensuing trajectory still visiting three target sets. It can be seen again in Figure 3C that the topology is explored to allow visiting two targets at once, namely, target sets IV and V.

In Figure 4A, it can be seen that with 60 seconds the trajectory with the least cost so far visits all five target sets. Figure 4B-F shows the connectivity graph at the instants each target set is visited.

Figures 5, 6, and 7 depict the same scenarios as Figures 2, 3, and 4, respectively, but employing the proposed modified version of Problem 2 in Subsection 2.5. It is most interesting to note that the results, given the same amount of time, are



**FIGURE 3** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting three target sets (white-filled squares). B, Connectivity graph at the instant target sets I and, C, IV and V are visited. Maximal time for obtaining this solution is 40 seconds with Problem 2. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

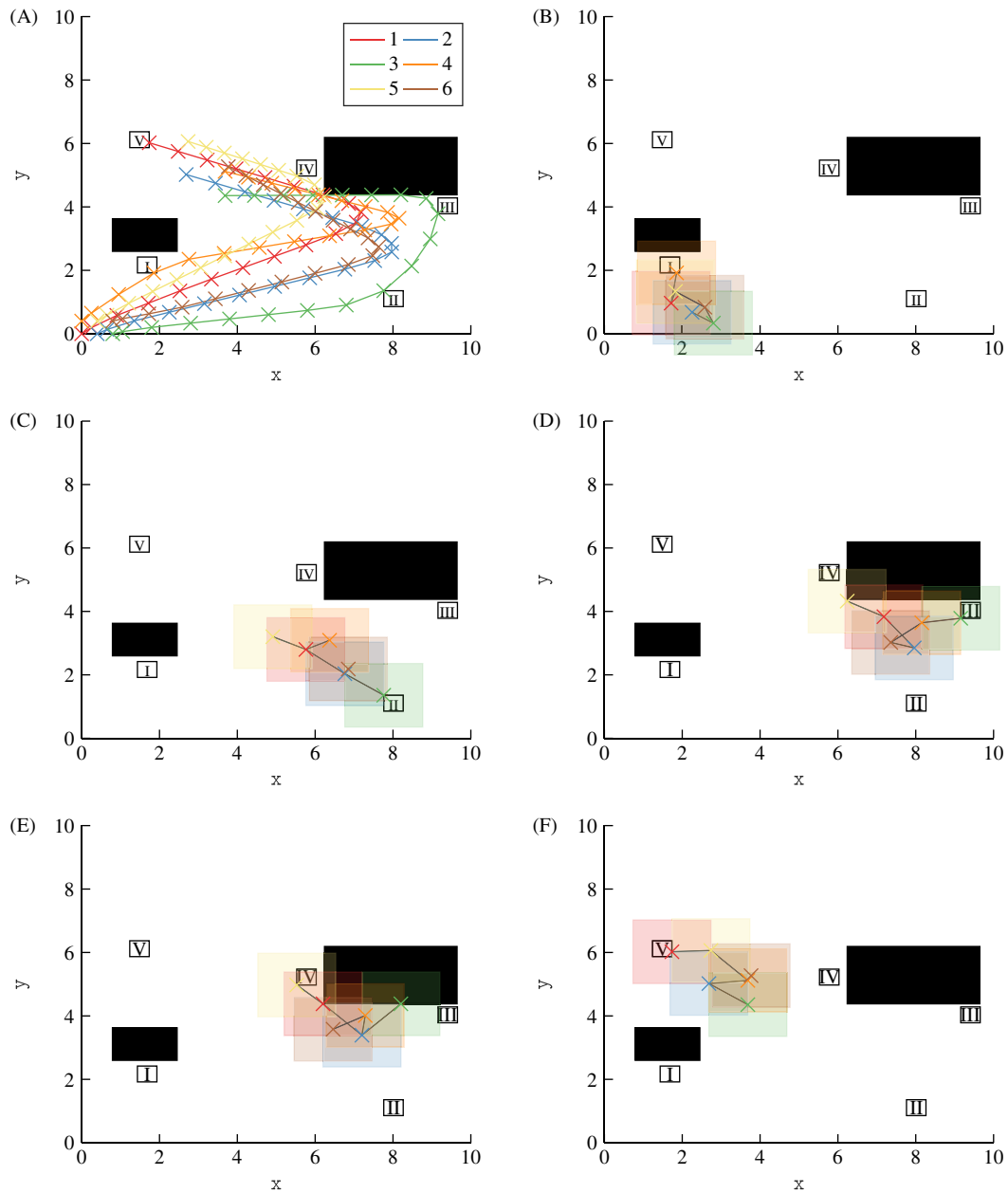
similar to the ones obtained with Problem 2, despite the fact that the modified version has more stringent constraints. As a matter of fact, by comparing the trajectories in Figures 6 and 3, one can see that the modified version was able to visit 4 targets given 40 seconds, whereas the best solution available so far to Problem 2 visits only three. After 60 seconds, both approaches find solutions that visit all five target sets.

For comparison, the trajectories when all the agents are required to remain within the communication range of each other are shown in Figure 8A. The optimal solution up until 20 seconds of computation time can be seen to visit three targets, as well as the case when the minimal connectivity is required in Figure 2A. Figure 8B-D shows the connectivity graph, which is totally connected, at the target visitation instants.

As the optimization time is increased to 40 seconds, the trajectories are displayed in Figure 9A. Similarly to the case of minimal connectivity (Figure 6), in this case four target sets are visited. The fully connected graph can be observed at the target visitation instants in Figure 9B-E.

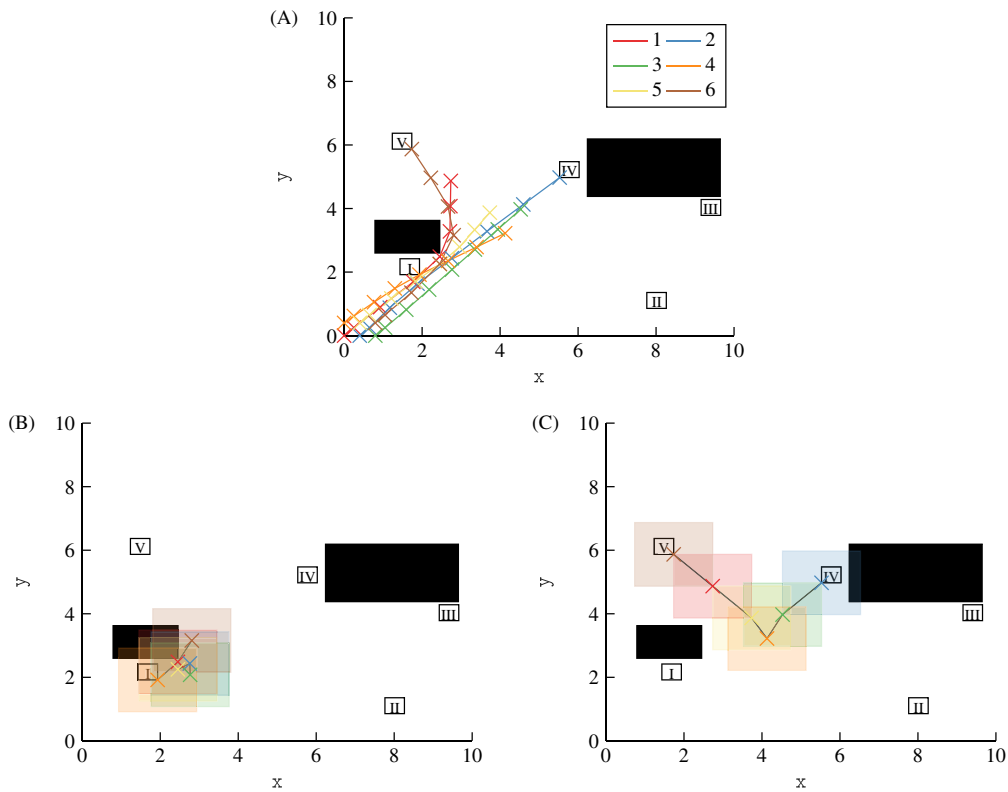
On the other hand, to find a solution that includes trajectories visiting all five targets, a computation time of 450 seconds is required with the full connectivity imposition. This is 7.5 times the computation time required by the proposed minimal connectivity algorithm. The trajectories are shown in Figure 10A, and the connectivity graphs at each target visitation instant are shown in Figure 10B-F.

As a matter of fact, the capacity of finding solutions with lesser cost than the full-connectivity imposition is inherent to the proposal in this work, since the solution with full connectivity is a particular one to the minimal connectivity algorithm. Table 2 shows the cost value obtained at each computation time with the three approaches, where (i) is Problem 2, (ii) is the modified Problem 2, and (iii) is the fully connected graph version. Three facts deserve commenting: (1) the cost obtained with approaches (i) and (ii) is smaller than the one obtained with the full connectivity given the same optimization time, (2) the best cost obtained after 450 seconds with the full connectivity is still greater than

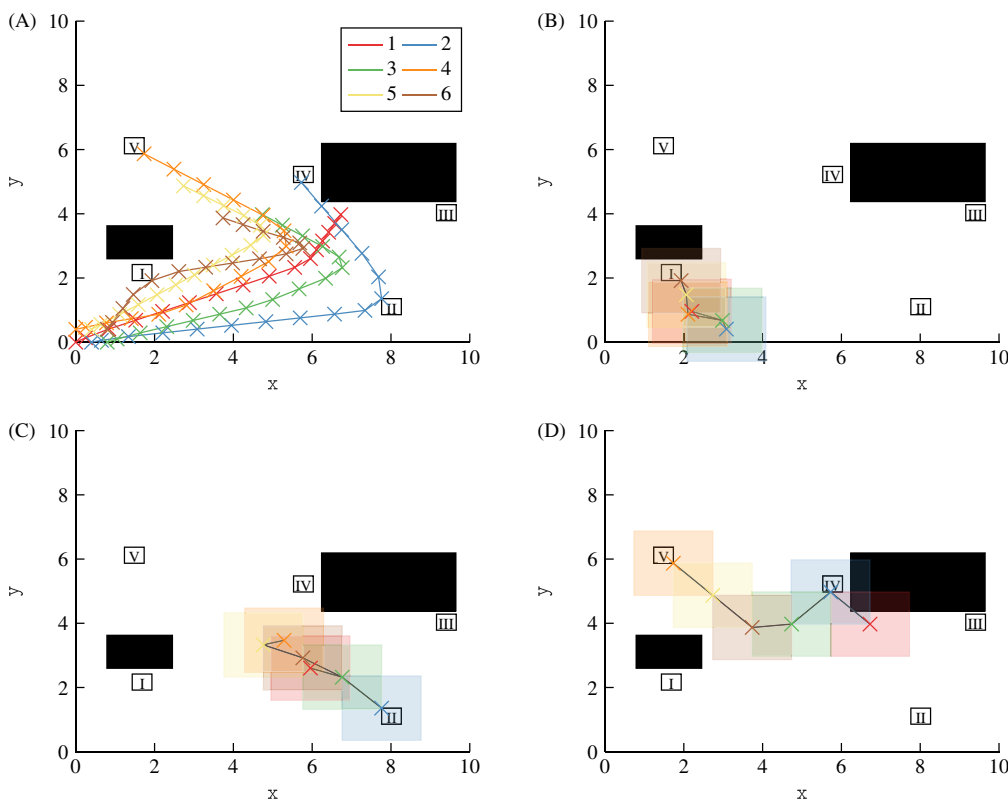


**FIGURE 4** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting five target sets (white-filled squares). B, Connectivity graph at the instant target sets I, C, II, D, III, E, IV, and F, V are visited. Maximal time for obtaining this solution is 60 seconds with Problem 2. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

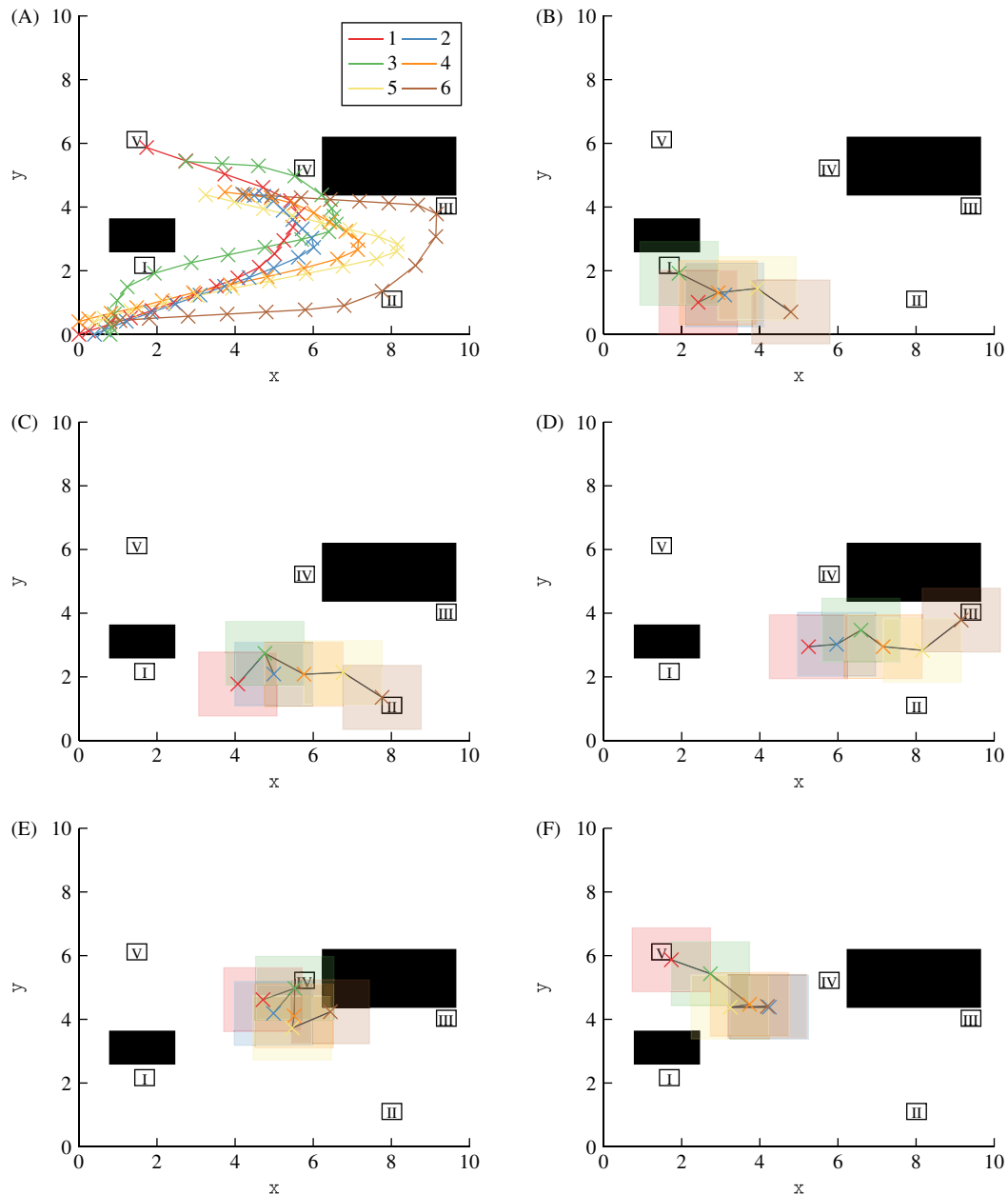
the one obtained after 60 seconds with the proposals (i) and (ii), and (3) the modified Problem 2 (approach (ii)) yields solutions with better cost compared with (i), given the same optimization time). This may be a counterintuitive result, as the approach (ii) is suboptimal when compared with (i), but the smaller number of constraints facilitates the obtainment of better solutions within the same optimization time. Of course, one might expect that, given enough time, the cost of approach (i) becomes smaller than or equal to the cost of approach (ii). Moreover, both approaches (i) and (ii) involve more binary variables than (iii), and still less computation time is required by (i) and (ii) to achieve better solutions, illustrating that the proposed algorithms, though in principle more computationally demanding, in practice show better results within the same amount of computation time.



**FIGURE 5** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting three target sets (white-filled squares). B, Connectivity graph at the instant target sets I and, C, IV and V are visited. Maximal time for obtaining this solution is 20 seconds with modified Problem 2. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]



**FIGURE 6** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting four target sets (white-filled squares). B, Connectivity graph at the instant target sets I, C, II, and D, IV and V are visited. Maximal time for obtaining this solution is 40 seconds with modified Problem 2. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

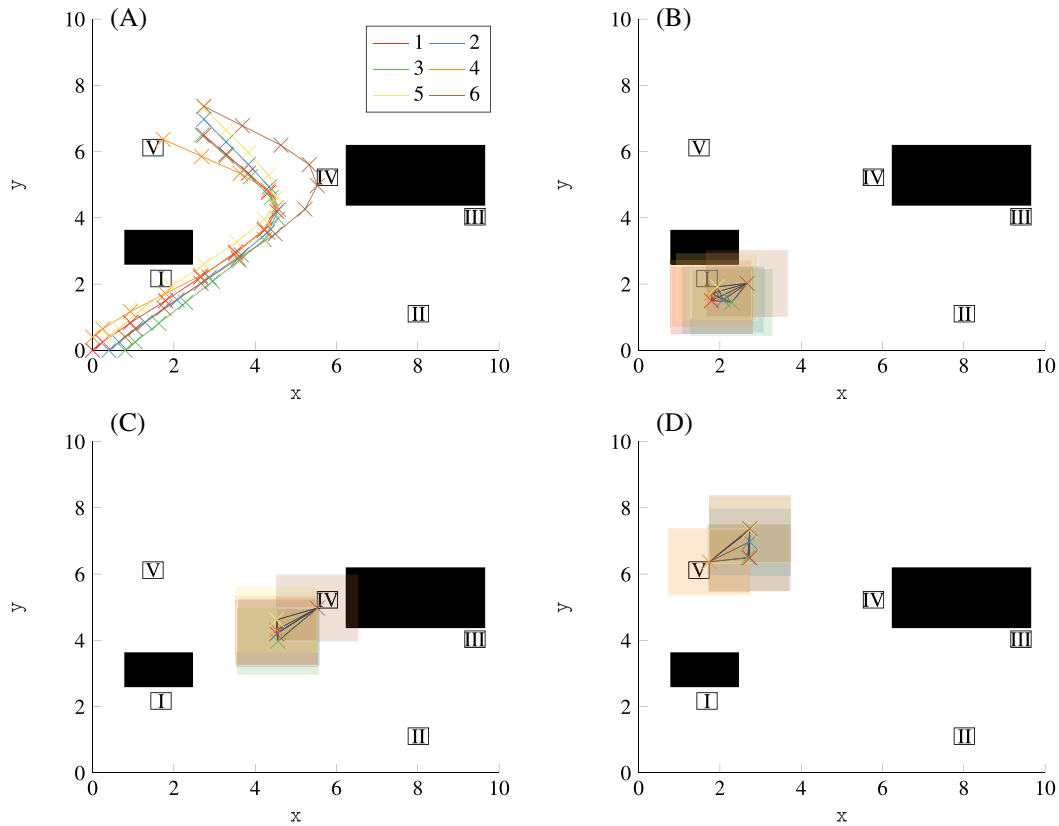


**FIGURE 7** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting five target sets (white-filled squares), B, connectivity graph at the instant target sets I, C, II, D, III, E, IV, and, F, V are visited. Maximal time for obtaining this solution is 60 seconds with modified Problem 2. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

### 3.2 | Results of Algorithm 2

In this subsection, we present results regarding Algorithm 2. We implemented this algorithm in C++ and executions with up to 100 agents were resulting in execution times on order of  $10^{-5}$  seconds, thus we may say that its execution time may be neglected when compared to the planning time.

In Figures 11A, 12A, and 13A, the initial communication graphs are shown, with agents assigned arbitrary id numbers, whereas Figures 11B, 12B, and 13B present the resulting numbering employing Algorithm 2. To verify its correctness, see Figure 11A, where the choice of numbering for the agents would not satisfy constraint (13), since the agent labeled with #1 must connect to two others, but this violates constraint (13). On the other hand, with the renumbering in Figure 11B,



**FIGURE 8** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting three target sets (white-filled squares). B, connectivity graph at the instant target sets I, C, IV, and D, V are visited. Maximal time for obtaining this solution is 20 seconds with all agents connected. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

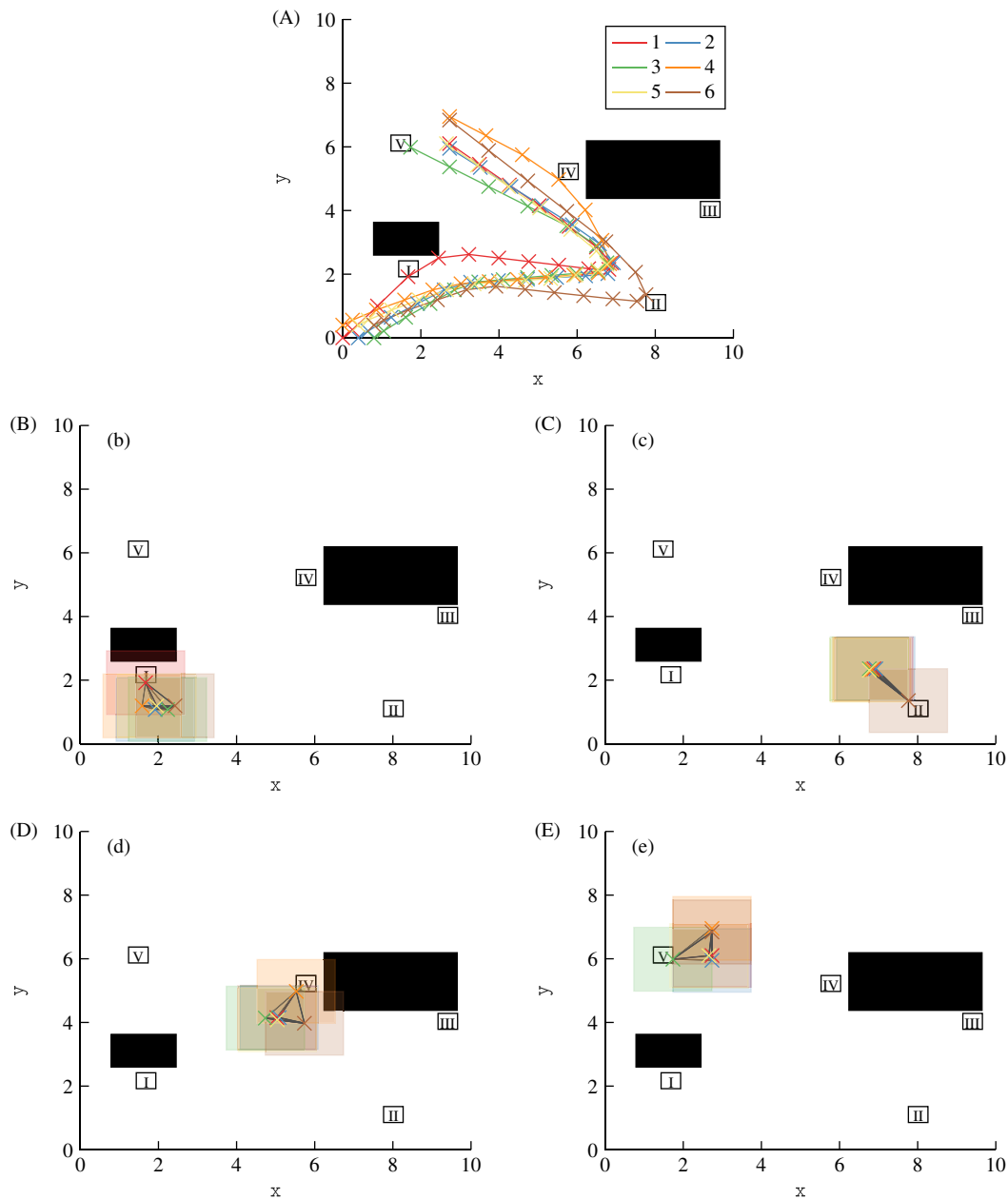
Approach	Maximal time (s)	Cost	Visited targets
(i)	20	-21.89	3
	40	-21.95	3
	60	-29.16	5
(ii)	20	-21.97	3
	40	-23.63	4
	60	-30.06	5
(iii)	20	-17.12	3
	40	-20.57	4
	450	-28.02	5

**TABLE 2** Cost evolution with time

the label #1 can be seen to have been assigned to the rightmost agent, which needs only to be connected to one neighbor (namely, the one labeled as #2 in Figure 11B), satisfying constraint (13).

## 4 | CONCLUSION

In this work, we developed mixed-integer linear programs for multiagent task allocation and trajectory planning with obstacle avoidance and global connectivity maintenance. In the considered task, the agents had to reach a final



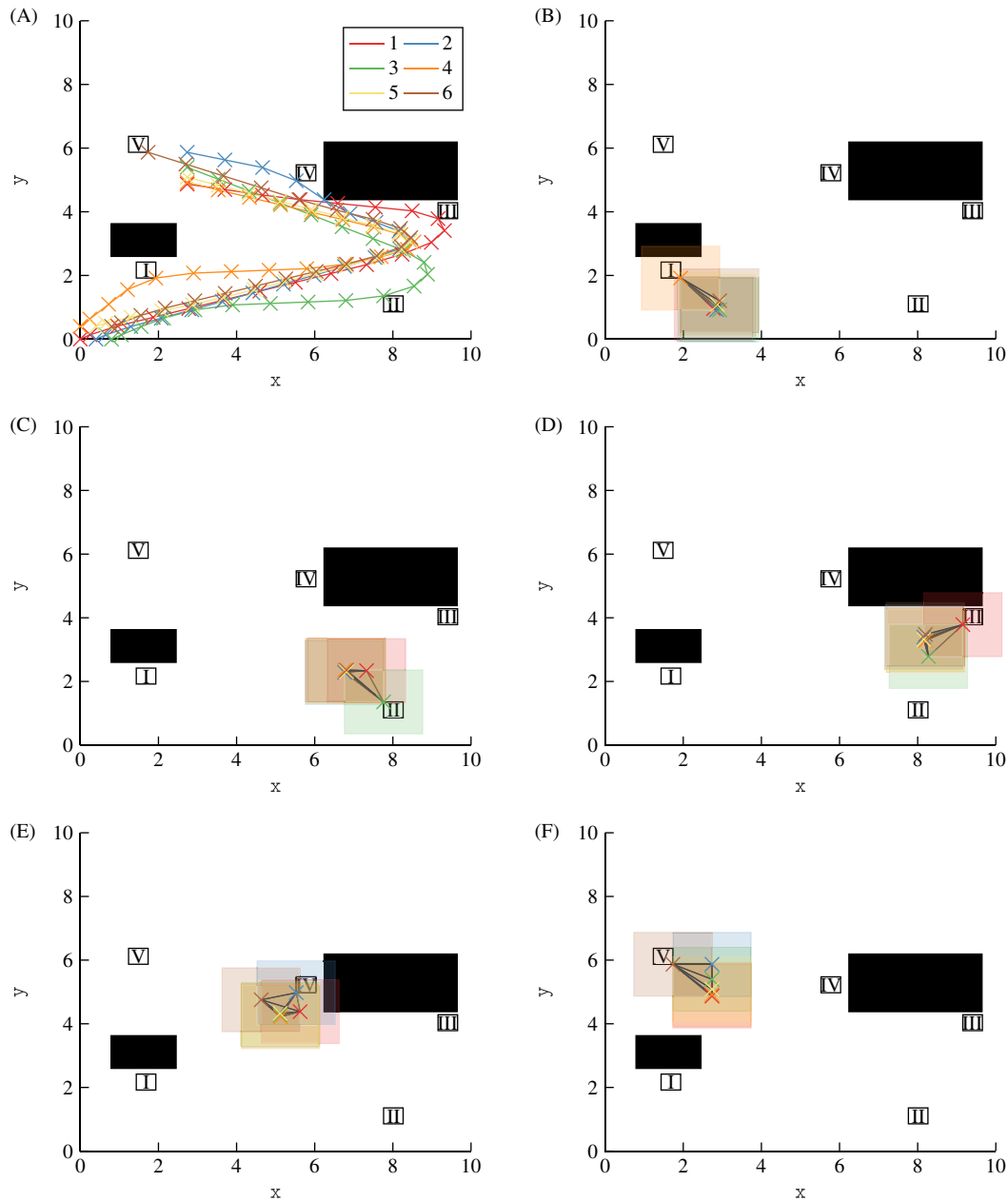
**FIGURE 9** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting four target sets (white-filled squares). B, Connectivity graph at the instant target sets I, C, II, D, IV, and E, V are visited. Maximal time for obtaining this solution is 40 seconds with all agents connected. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

destination, but were allowed to collect rewards during the task. The proposed formulations plan trajectories for the team of agents while deciding which agent should collect which reward.

Two optimization formulations were presented. The first one is optimal, considering the exact number of constraints that are necessary to model the connectivity problem. However, the number of connectivity maintenance constraints grows exponentially in the number of agents. Therefore, we also introduced an heuristic formulation where the connectivity maintenance increases linearly in the number of agents, which is a more scalable solution. We call this approach heuristic, because it does not consider all possible communication graphs the MAS may assume, so it is not ruled out that a lower cost solution using the exact connectivity constraints might be found.

Simulation results were used to evaluate the proposed approaches. The planner indeed found feasible trajectories for the MAS while respecting the constraints and was able to find solutions where the agents collect all rewards

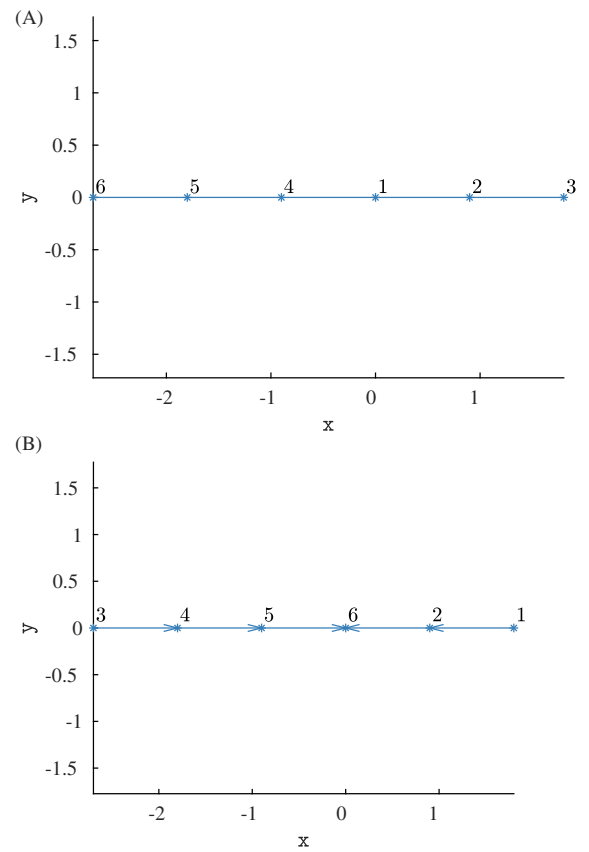




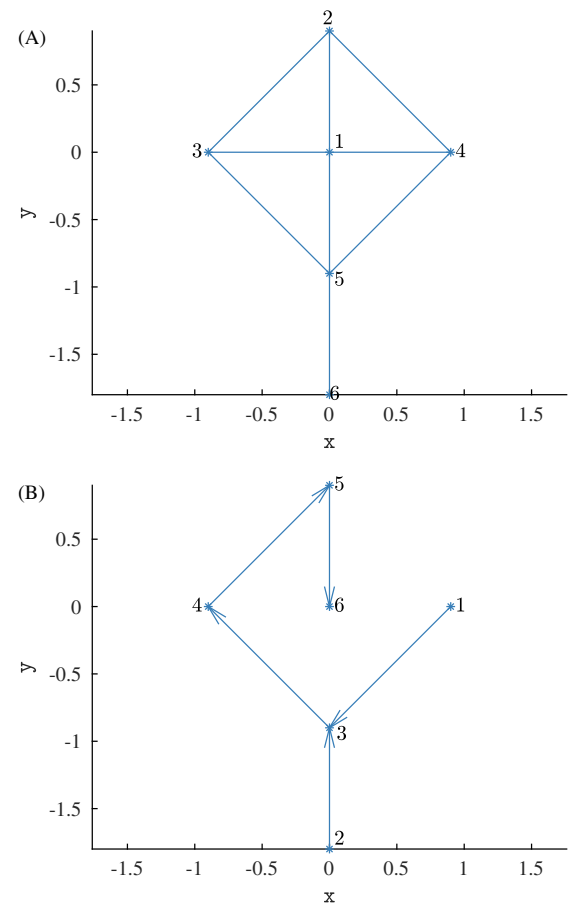
**FIGURE 10** A, Trajectories of each of the six agents avoiding the obstacles (black-filled rectangles) and visiting five target sets (white-filled squares). B, Connectivity graph at the instant target sets I, C, II, D, III, E, IV, and F, V are visited. Maximal time for obtaining this solution is 450 seconds with all agents connected. The communication range of each agent is depicted as a semitransparent square [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

when the optimization solver was allowed to run for enough time. Moreover, we showed that both formulations obtained better solutions than a planner, which obligates the communication graph to be fully connected during the task.

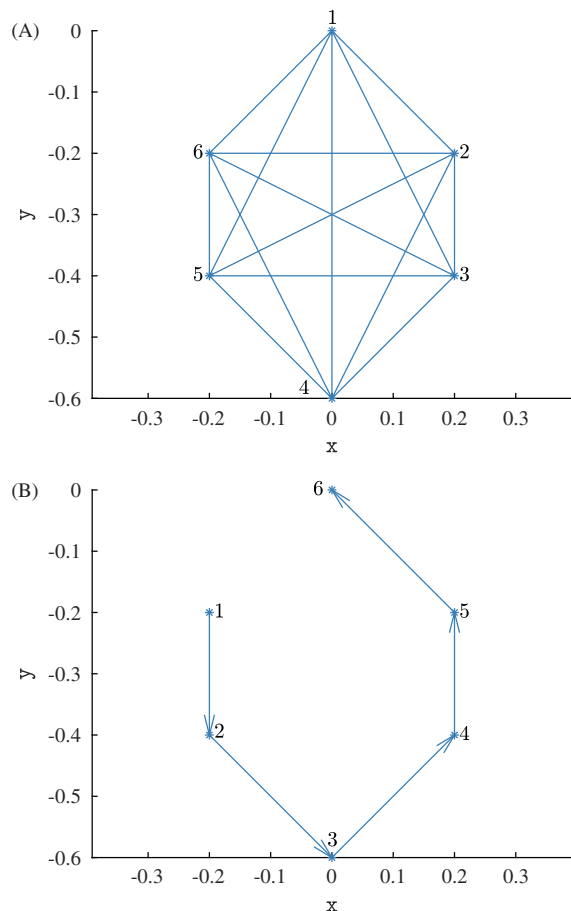
Further enhancements that could be done with small additional overhead include: (i) implementation of a constraint tightening-approach to ensure robustness to external disturbances; (ii) incorporation of penalties proportional to the projection of the vectors connecting pairs of agents over some predefined directions so as to approximate the interagent distances, thus prioritizing solutions with lower dispersion between the agents; (iii) introduction of bounds on the fuel expense (herein represented by the sum of the absolute values of the control vector) of each agent; and (iv) introduction of a capacity bound of each agent, which is consumed by visiting the target sets, so as to consider knapsack-like problems.



**FIGURE 11** An example of application of Algorithm 2 where the agents are initially disposed in a line: A, communication graph with numbers representing arbitrary agents' ids and, B, spanning tree and agents' numbers as computed by Algorithm 2 (the directed edges connect each node to its parent) [Colour figure can be viewed at wileyonlinelibrary.com]



**FIGURE 12** An example of application of Algorithm 2 where the agents are initially disposed in a shape that resembles a kite: A, communication graph with numbers representing arbitrary agents' ids and, B, spanning tree and agents' numbers as computed by Algorithm 2 (the directed edges connect each node to its parent) [Colour figure can be viewed at wileyonlinelibrary.com]



**FIGURE 13** An example of application of Algorithm 2 where the agents are initially disposed in an hexagon shape: A, communication graph with numbers representing arbitrary agents' ids and, B, spanning tree and agents' numbers as computed by Algorithm 2 (the directed edges connect each node to its parent) [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

Future work could also encompass consideration of more realistic agent models and validation of the proposal with laboratory experiments.

## ACKNOWLEDGEMENT

R.A. and M.M. acknowledge the support of grant 2016/03647-3, São Paulo Research Foundation (FAPESP); R.A. acknowledges CAPES (grant 88881.145490/2017-01) and the German Ministry of Education and Research through the Alexander von Humboldt Foundation for his fellowship; and R.G. acknowledges the support of grant 303393/2018-1, National Council for Scientific and Technological Development (CNPq).


## FOOTNOTES

Rubens J.M. Afonso is on leave from Instituto Tecnológico de Aeronáutica for a postdoctoral research stay at the Technical University of Munich.

## ORCID

Rubens J.M. Afonso  <https://orcid.org/0000-0001-9209-2253>

Marcos R.O.A. Maximo  <https://orcid.org/0000-0003-2944-4476>

Roberto K.H. Galvão  <https://orcid.org/0000-0001-9794-8815>

## REFERENCES

- Agmon N, Fok CL, Emaliah Y, Stone P, Julien C, Vishwanath S. On coordination in practical multi-robot patrol. Paper presented at: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA); 2012:650-656; St. Paul.
- Fazli P. On multi-robot area coverage. *6085 of Advances in Artificial Intelligence*. Berlin, Germany: Springer; 2010:384-387.
- Baxter JL, Burke EK, Garibaldi JM, Norman M. Multi-robot search and rescue: a potential field based approach. *Autonomous Robots and Agents*. Vol 76. Berlin, Germany: Springer; 2007:9-16.
- Burgard W, Moors M, Fox D, Simmons R, Thrun S. Collaborative multi-robot exploration. Paper presented at: Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065). Proceedings of the IEEE International Conference on Robotics and Automation (ICRA); 2000:476-481; San Francisco.
- Yan Z, Jouandeau N, Cherif AA. A survey and analysis of multi-robot coordination. *J Adv Robot Syst*. 2013;10(12):1-18. <https://doi.org/10.5772/57313>.

6. Gasparri A, Sabattini L, Ulivi G. Bounded control law for global connectivity maintenance in cooperative multirobot systems. *IEEE Trans Robot.* 2017;33(3):700-717. <https://doi.org/10.1109/TRO.2017.2664883>.
7. Kantaros Y, Zavlanos MM. Global planning for multi-robot communication networks in complex environments. *IEEE Trans Robot.* 2016;32(5):1045-1061. <https://doi.org/10.1109/TRO.2016.2593045>.
8. Stephan J, Fink J, Kumar V, Ribeiro A. Concurrent control of mobility and communication in multirobot systems. *IEEE Trans Robot.* 2017;33(5):1248-1254. <https://doi.org/10.1109/TRO.2017.2705119>.
9. Filotheou A, Nikou A, Dimarogonas DV. Robust decentralised navigation of multi-agent systems with collision avoidance and connectivity maintenance using model predictive controllers. *Int J Control.* 2018;93(6):1470-1484. <https://doi.org/10.1080/00207179.2018.1514129>.
10. Karkoub M, Atınc G, Stipanovic D, Voulgaris P, Hwang A. Trajectory tracking control of unicycle robots with collision avoidance and connectivity maintenance. *J Intell Robot Syst.* 2019;96:331-343. <https://doi.org/10.1007/s10846-019-00987-2>.
11. Khatib O. Real-time obstacle avoidance for manipulators and mobile robots. *Int J Robot Res.* 1986;5(1):90-98. <https://doi.org/10.1177/027836498600500106>.
12. Lavelle SM. *Planning Algorithms*. New York, NY: Cambridge University Press; 2006.
13. Kuffner JJ, LaValle SM. RRT-connect: an efficient approach to single-query path planning. Paper presented at: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA); 2000:995-1001; San Francisco.
14. Gustavi T, Dimarogonas DV, Egerstedt M, Hu X. Sufficient conditions for connectivity maintenance and rendezvous in leader-follower networks. *Automatica.* 2010;46(1):133-139. <https://doi.org/10.1016/j.automatica.2009.10.014>.
15. Richards A, How JP. Robust variable Horizon model predictive control for vehicle maneuvering. *Int J Robust Nonlinear Control.* 2006;16(7):333-351. <https://doi.org/10.1002/rnc.1059>.
16. Schouwenaars T, Moor DB, Feron E, How JP. Mixed integer programming for multi-vehicle path planning. Paper presented at: Proceedings of the 2001 European Control Conference (ECC). Proceedings of the European Control Conference (ECC); 2001:2603-2608; Porto, Portugal.
17. Prodan I, Olaru S, Stoican C, Niculescu SI. On the tight formation for multi-agent dynamical systems. *Agent and Multi-Agent Systems. Technologies and Applications*. Berlin, Heidelberg/Germany: Springer; 2012:554-565.
18. Maia MH, Galvão RKH. On the use of mixed-integer linear programming for predictive control with avoidance constraints. *Int J Robust Nonlinear Control.* 2009;19(7):822-828. <https://doi.org/10.1002/rnc.1341>.
19. Richards A, Turnbull O. Inter-sample avoidance in trajectory optimizers using mixed-integer linear programming. *Int J Robust Nonlinear Control.* 2015;25(4):521-526. <https://doi.org/10.1002/rnc.3101>.
20. Stoican F, Prodan I, Grøtli EI. Exact and overapproximated guarantees for corner cutting avoidance in a multiobstacle environment. *Int J Robust Nonlinear Control.* 2018;154(2):549. <https://doi.org/10.1002/rnc.4248>.
21. Sabattini L, Chopra N, Secchi C. Decentralized connectivity maintenance for cooperative control of mobile robotic systems. *Int J Robot Res.* 2013;32(12):1411-1423. <https://doi.org/10.1177/0278364913499085>.
22. Cormen T, Leiserson C, Rivest R, Stein C. *Introduction to Algorithms*. 3rd ed. Cambridge, MA: The MIT Press; 2009.
23. Fleming J, Kouvaritakis B, Cannon M. Robust tube MPC for linear systems with multiplicative uncertainty. *IEEE Trans Automat Control.* 2015;60(4):1087-1092. <https://doi.org/10.1109/TAC.2014.2336358>.
24. Agarwal A, Bhat S, Gray A, Grossmann IE. Automating mathematical program transformations. *Practical Aspects of Declarative Languages (PADL 2010) - Lecture Notes in Computer Science*. Vol 5937. Berlin, Germany: Springer; 2010:134-148.
25. Richards A, Schouwenaars T, How JP, Feron E. Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming. *J Guid Control Dyn.* 2002;25(4):755-764. <https://doi.org/10.2514/2.4943>.
26. Jünger M, Liebling TM, Naddef D, et al. *50 Years of Integer Programming 1958-2008*. Berlin, Heidelberg/Germany: Springer; 2009.
27. Vielma JP. Mixed integer linear programming formulation techniques. *SIAM Rev.* 2015;57(1):3-57. <https://doi.org/10.1137/130915303>.
28. Prodan I, Stoican F, Olaru S, Niculescu SI. *Mixed-Integer Representations in Control Design*. New York, NY: Springer International Publishing; 2016.
29. Boyd S, Vandenberghe L. *Convex Optimization*. New York, NY: Cambridge University Press; 2009.
30. Punnen AP. The traveling salesman problem: applications, formulations and variations. *The Traveling Salesman Problem and Its Variations. Combinatorial Optimization*. Vol 12. Boston, MA: Springer; 2007:1-28.
31. Morris JM. Traversing binary trees simply and cheaply. *InfProcess Lett.* 1979;9(5):197-200. [https://doi.org/10.1016/0020-0190\(79\)90068-1](https://doi.org/10.1016/0020-0190(79)90068-1).

## SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of this article.

**How to cite this article:** Afonso RJM, Maximo MROA, Galvão RKH. Task allocation and trajectory planning for multiple agents in the presence of obstacle and connectivity constraints with mixed-integer linear programming. *Int J Robust Nonlinear Control.* 2020;30:5464-5491. <https://doi.org/10.1002/rnc.5092>