Autonomous collision avoidance for wheeled mobile robots using a differential game approach

Thulasi Mylvaganam\textsuperscript{a,}\textsuperscript{b}, Mario Sassano\textsuperscript{b}

\textsuperscript{a} Department of Aeronautics, Imperial College London, London SW7 2AZ, UK
\textsuperscript{b} Dipartimento di Ingegneria Civile e Ingegneria Informatica, "Tor Vergata", Via del Politecnico 1, Rome 00133, Italy

1. Introduction

With several important areas of application, such as mobile sensor networks \cite{2,9,10,17,18,36}, air traffic management \cite{7,35}, power systems \cite{19} and distributed robotics, including collision avoidance and formation flying \cite{15,20,22}, multi-agent systems have gained much interest in recent years. Essentially a multi-agent system is a system consisting of several, interacting subsystems, which we refer to as agents. While a system of several simple agents can perform tasks more efficiently and reliably than a single (possibly more complex) agent, often such systems and their applications call for the development of novel control methodologies.

In the context of robot navigation many approaches available in the literature are based on a so-called navigation function (see, for instance \cite{3,6,31,33,34}) which is constructed from the geometric information on the considered topology and employed to define gradient descent control laws. Recognising that multi-agent systems are ubiquitous in the nature, other approaches are inspired by naturally occurring systems, such as schools of fish, migrating birds and swarms of bees \cite{8,12,25–27,29,32}. In the presence of communication constraints, the communication topology is often represented by a graph and control strategies are designed using notions from graph theory \cite{15,28}. The use of game theory in the context of multi-agent systems has also been explored (see, for instance, \cite{11,13,14,24}).

The multi-agent collision avoidance problem for a system of $N$ wheeled mobile robots (WMRs) is considered in this paper, i.e. assuming the WMRs satisfy unicycle dynamics we consider the problem of designing feedback controls such that the robots autonomously reach predefined targets while avoiding collisions with obstacles and other robots. A Lyapunov-based approach for avoidance control has been considered in \cite{11}, where the authors recognise that the main challenge associated with the approach lies in determining Lyapunov functions. In \cite{22,24} the multi-agent collision avoidance has been considered for agents satisfying single-integrator dynamics. Exploiting the single-integrator dynamics of the agents, feedback strategies which approximate the Nash equilibrium solution of the game are readily constructed, thus providing a systematic manner of constructing Lyapunov functions (see \cite{22,24} for details).

In this paper we extend the results of \cite{24} to WMRs. We also present several interesting and challenging case studies. For instance, we consider situations which include several sources of potential deadlocks which, to the best of our knowledge, are not readily solved using other approaches. The multi-agent collision avoidance problem is solved in two steps. First, virtual agents satisfying single-integrator dynamics are introduced and the multi-agent collision avoidance problem for the virtual agents is solved.
using a differential game formulation of the problem and exploiting the main results of [23,24]. Second, the solution of the differential game involving virtual agents is used to design feedback strategies for the WMRs which ensure that the robots reach their targets while avoiding collisions. The novel contributions presented in this paper can be roughly summarised as follows. The introduction of virtual agents allows to use the main result of [24] to generate a trajectory plan for the WMRs, which can then be tracked with zero error. Moreover, the case studies provide useful insights and validate the theoretical results obtained both in this paper and in [24] through simulations of complex and challenging scenarios as well as a sensitivity analysis with respect to certain control parameters.

The remainder of the paper is structured as follows. The multi-agent collision avoidance problem is introduced in Section 2. A method of generating collision-free trajectories exploiting the virtual agents is presented in Section 3 (see [24] for details). In Section 4 the virtual agents are used to solve the multi-agent collision avoidance problem for the WMRs. A series of illustrative numerical examples are then provided in Section 5 before some concluding remarks and directions for future research are given in Section 6.

Notation: Standard notation is used in the remainder of the paper. The set of real numbers is denoted by \( \mathbb{R} \), the set of complex numbers is denoted by \( \mathbb{C} \) and \( \mathbb{C}^n \) denotes the open left-half plane of \( \mathbb{C} \). Given a vector \( \mathbf{x} \in \mathbb{R}^n \), the gradient of a function \( V : \mathbb{R}^n \rightarrow \mathbb{R} \) is denoted by \( \nabla V \). The weighted Euclidean norm of a vector is denoted by \( \| \mathbf{v} \|_A = \sqrt{\mathbf{v}^T A \mathbf{v}} \), where \( A = A^T > 0 \) and \( A \in \mathbb{R}^{n \times n} \). The shorthand \( A = [A_{ij}] \) is used to denote the block matrix \(\begin{bmatrix} A_{11} & \ldots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{N1} & \ldots & A_{NN} \end{bmatrix} \). The empty set is denoted by \( \emptyset \).

2. Problem formulation

In this section the multi-agent collision avoidance problem is introduced, formulated and studied in a centralised setting, i.e. the positions of each agent are available to the remaining members of the group at all times. We consider a team of \( N \) WMRs moving on the ground (Euclidean plane), possibly characterised by the presence of (static) obstacles. In particular, each WMR is described by a unicycle-like model with a passive wheel, the dynamics of which are defined (see, for instance, [4,5]) by the equations

\[
\begin{align*}
\dot{X}_i &= \cos(\theta_i) v_i, \\
\dot{Y}_i &= \sin(\theta_i) v_i, \\
\dot{\theta}_i &= \omega_i,
\end{align*}
\]

with \( i = 1, \ldots, N \), where \((X_i, Y_i) \in \mathbb{R}^2\) denote the position on the Euclidean plane of the middle point between the two actuated wheels along the direction of the axle connecting the wheels, \(\theta_i\) denotes the orientation of the i-th robot and \(v_i\) and \(\omega_i\) represent the longitudinal and the angular velocity controls, respectively. The constant \( q_i \) describes the distance between the point \((X_i, Y_i)\) and the centre of mass of the i-th robot \((x_i, y_i) \in \mathbb{R}^2\), along the segment connecting \((X_i, Y_i)\) and the (forward) passive wheel (see Fig. 1). The control task may be informally stated as follows: given a target location \((X_c, Y_c) \in \mathbb{R}^2\) for the centre of mass of each agent, determine control inputs \(v_i\) and \(\omega_i\) such that the i-th agent is steered towards the desired point while avoiding collisions with other members of the team and with static obstacles.

**Problem 1.** Consider a multi-agent system consisting of \( N \) WMRs with dynamics (1), for \( i = 1, \ldots, N \). The multi-agent collision avoidance problem consists in determining feedback control laws \(v_i\) and \(\omega_i, i = 1, \ldots, N\), that steer each agent from its initial position to a predefined target while avoiding collisions between agents and among agents and static obstacles.

3. Collision-free trajectory planning

**Problem 1** is solved in two steps. First, \( N \) virtual agents (one for each robot) satisfying single-integrator dynamics are introduced and collision-free trajectories are planned for the virtual agents. This result is one of the main contributions of [24] and it is based on a formulation of the multi-agent collision avoidance problem in terms of a nonzero-sum differential game for which a closed-form (approximate) solution is provided. Second, control inputs \(v_i\) and \(\omega_i\) are determined to steer the i-th robot with dynamics (1) along the collision-free trajectories, thus solving Problem 1. The first step, namely collision-free trajectory planning, is considered in this section. The reader is referred to [24] for a more detailed discussion on the results presented in this section.

We consider the case in which the movement of each virtual agent is described by single-integrator dynamics, i.e.

\[
\dot{x}_i = u_i, \\
i = 1, \ldots, N,
\]

where \( u_i \in \mathbb{R}^2 \) is the control input of the i-th agent and the position of the i-th agent is denoted by \( x_i \in \mathbb{R}^2 \). Each virtual agent is associated with a desired goal, namely a target position \( x^*_i \in \mathbb{R}^2, i = 1, \ldots, N \). Let \( \bar{x}_i = x_i - x^*_i \), i.e. \( \bar{x}_i \) denotes the error variable between the current position of the i-th agent and its corresponding target position. Moreover, each agent i is associated with a parameter \( r_i > 0 \), which plays the role of safety radius, which may differ from one agent to another. These values take into account the fact that the agents are WMRs with dynamics defined by Eqs. (1), rather than point masses with the single-integrator dynamics (2), i.e. they account for the physical dimensions of the unicycle.

A static obstacle is represented by its centre of mass \( p_i^e \in \mathbb{R}^2 \) and the region of the Euclidean plane that it occupies \( P_j \subset \mathbb{R}^2 \), with \( j = 1, \ldots, m \), where \( m \geq 0 \) denotes the number of static obstacles present. The boundary of the region \( P_j \) is denoted by \( \partial P_j \) and in what follows elliptical obstacles are considered, i.e.

\[
\delta P_j = \{ x \in \mathbb{R}^2 : \| x - p_j \|^2 - \rho^2_j = 0 \},
\]

where \( \rho_j > 0 \) and \( E_j = E^T_j > 0 \). There is a one-to-one relation between the point \( p_j \), \( \rho_j \) and \( E_j \) and the physical parameters of the ellipse, i.e. the centre of mass \( p_j^e \) and the lengths of the semi-axes. The obstacle avoidance region and the agent avoidance region of the i-th agent are defined as follows.
Definition 1. Consider the open sets $S_j = \{x \in \mathbb{R}^2 : \|x - p_j\|_j^2 < \rho_j^2\}$. The obstacle avoidance region, denoted by $S$, is defined as $S = \bigcup_{j=1}^{m} S_j$.

Definition 2. Given a time instant $t \geq 0$, consider the sets $\mathcal{D}_i^j = \{x \in \mathbb{R}^2 : \|x - x_i(t)\|^2 \leq (r_i + \rho_i^2)\}$. The agent avoidance region of the $i$-th agent at $t$, denoted $\mathcal{D}_i^j$, is defined as $\mathcal{D}_i^j = \bigcup_{j=1,j \neq i}^{N} \mathcal{D}_i^j$.

Let $\mathcal{D}_i^j$ and $\mathcal{S}$ denote the complement of the sets $\mathcal{D}_i^j$ and $\mathcal{S}$, respectively. A collision-free trajectory for the $i$-th agent is defined in the following statement.

Definition 3. The $i$-th agent is said to be collision-free if $x_i(t) \not\in \mathcal{D}_i^j \cup \mathcal{S}$ for all $t \geq 0$, or equivalently $x_i(t) \in \mathcal{D}_i^j \cap \mathcal{S}$ for all $t \geq 0$.

Let $\bar{\rho}_j(\phi)$ denote the radius of the ellipse $\mathcal{E}_j$ in polar coordinates as a function of the angle $\phi$ of the line segment connecting $x_i(\phi)$ and $p_j$, relative to the polar description of $\mathcal{E}_j$, i.e., $(p_{0,j}, \rho_{0,j})$. We assume that sufficient time is provided to accomplish the task of steering each virtual agent from its initial position $x_i(0)$ to its corresponding target position $x_i^*$ and that the following conditions are satisfied.

**Assumption 1.**

i) **Obstacle collision-free initial deployment:** the initial positions of the agents satisfy

$$\|x_i(0) - p_j\|^2 > (r_i + \bar{\rho}_j(\phi(0)))$$

for all $i = 1, \ldots, N$, $j = 1, \ldots, m$.

ii) **Agent collision-free initial deployment:** the initial positions of the agents satisfy

$$\|x_i(0) - x_j(0)\| > r_i + r_j$$

for all $i = 1, \ldots, N$ and $j = 1, \ldots, N$, $j \neq i$.

iii) **Obstacle collision-free desired deployment:** the target positions of the agents satisfy

$$\|x_i^* - p_j\|^2 > (r_i + \bar{\rho}_j(\phi^*))$$

for all $i = 1, \ldots, N$, $j = 1, \ldots, m$.

iv) **Agent collision-free desired deployment:** the target positions for each agent satisfy

$$\|x_i^* - x_j^*\| > r_i + r_j$$

for all $i = 1, \ldots, N$ and $j = 1, \ldots, N$, $j \neq i$.

v) **Configuration feasibility:** the static obstacles do not form an impermeable boundary about targets of one or more of the agents. Namely, there exists a continuous path $I_i$, connecting the initial condition $x_i(0)$ and the target $x_i^*$ satisfying

$$I_i \cap \bigcup_{j=1, \ldots, m} \partial \mathcal{E}_j = \emptyset$$

for $i = 1, \ldots, N$.

The problem of planning a collision-free trajectory for the virtual agents is defined in the following statement.

**Problem 2.** Consider a multi-agent system consisting of $N > 1$ agents with dynamics (2), for $i = 1, \ldots, N$. The multi-agent collision avoidance problem consists in determining feedback control strategies $u_i$, $i = 1, \ldots, N$, that steer each agent from its initial position to a predefined target while avoiding collisions between agents and among agents and static obstacles.

As will be seen in Section 4, the solution of Problem 2, which involves the virtual agents, is instrumental to the systematic construction of a solution to Problem 1, which concerns the WMRs.

To streamline the presentation the following notation is introduced at this stage. Let $\xi = [\xi_1, \ldots, \xi_N] \in \mathbb{R}^{2N}$, where $\xi_i \in \mathbb{R}^2$, $i = 1, \ldots, N$. Denote a dynamic extension and consider $P_i(x)$, $i = 1, \ldots, N$, given by

$$P_i(\bar{x}) = \left[\bar{P}_i^1(\bar{x})^T + \gamma_1 I\right]$$

where $\bar{P}_i^1 \in \mathbb{R}^{2 \times 2}$, $k = 1, \ldots, N$, $j = 1, \ldots, N$, and $\gamma_1 > 0$ is a constant parameter,

$$\bar{P}_i^1(\bar{x}) = \sqrt{\alpha_i + \beta_i^1 g_i^1(\bar{x}) + \beta_i^2 g_i^2(\bar{x}) I},$$

with the so-called collision avoidance functions defined as

$$g_i^1(\bar{x}) = \sum_{j=1}^{m} \frac{1}{\| (\bar{x}_i + x_j^* - p_j) \|_j^2 - \rho_j^2 } ,$$

$$g_i^2(\bar{x}) = \sum_{j=1,j \neq i}^{N} \frac{1}{\| \bar{x}_i - x_j^* \|_j^2 - \rho_j^2} .$$

for $c \geq 1$, and $\bar{P}_i^1 = 0$ for $k \neq i$ and $j \neq i$. Let $\bar{R}_i = R_i^j \in \mathbb{R}^{2N \times 2N}$ and consider the partition $\bar{R}_i = \left[N_{ij}\right]$ where $N_{ij} \in \mathbb{R}^{2 \times 2}$, $k = 1, \ldots, N$, $j = 1, \ldots, N$. Consider the set $\mathcal{M} = [\xi \in \mathbb{R}^{2N} : g_i^1(\bar{x}) + g_i^2(\bar{x}) < \infty]$. In [22,24] it is demonstrated that the multi-agent collision avoidance problem for the virtual agents, i.e. Problem 2, can be formulated as a differential game for which (approximate) solutions can be constructed in closed-form.

In particular, it can be demonstrated that the functions

$$V_i = \frac{1}{2} \xi^T P_i(\bar{x}) \xi + \|\xi - x_i\|_2^2$$

allow us to systematically construct so-called $\epsilon_\alpha$-Nash equilibrium strategies (see [23] for details) of the differential game defined by the dynamics (2), $i = 1, \ldots, N$, and individual cost functionals

$$J_i(\bar{x}(0), u_1, \ldots, u_N) = \frac{1}{2} \int_0^1 \left( (\dot{\bar{x}}_i + \beta_i^1 g_i^1(\bar{x}) + \beta_i^2 g_i^2(\bar{x}) \right) \bar{x}_i^T \bar{x}_i + ||u_i||_2^2 \right) dr .$$

which each agent $i$, $i = 1, \ldots, N$, seeks to minimise. Namely, using only the matrix-valued mappings $P_i$, for $i = 1, \ldots, N$, we can define the functions (12) $i = 1, \ldots, N$, and thus construct dynamic strategies $(u_{ij}, u_{ij}^\alpha)$ so that for $i = 1, \ldots, N$, satisfy the inequalities

$$J_i(\bar{x}_0, u_i^*, u_{ij}^* < J_i(\bar{x}_0, u_i, u_{ij}^*) + \epsilon_{\alpha, i}]$$

for all strategies $u_{ij}, u_{ij}^\alpha = (u_{ij}^1, u_{ij}^2, \ldots, u_{ij}^{\alpha})$ such that the system (2), $i = 1, \ldots, N$, in closed-loop with $u_{ij}, u_{ij}^\alpha$ is (locally) asymptotically stable and $\sigma(A_{ij} + a\dot{\alpha}) C^\alpha$. When $\alpha > 0$, $A_{ij}$ is the matrix describing the linearisation of the closed-loop system around the origin, $(u_{ij}, u_{ij}^\alpha)$ is the set of strategies $(u_i)$ for $j = 1, \ldots, N$ and $j \neq i$ and $\epsilon_{\alpha, i} \geq 0$. This, in turn, provides a solution of Problem 2 as recalled in the following statement which summarises the main result of [24].

**Theorem 1** [24]. Consider the dynamics (2) and the matrix-valued mappings (9). Let $\xi(0) \in \mathcal{M}$ and suppose that Assumption 1 holds. Then there exist $k \geq 0$, $R_k$, $i = 1, \ldots, N$, and a neighbourhood

---

1 Given an ellipse $\mathcal{E}_j$, the function $\bar{\rho}_j(\phi)$ can be computed by straightforward computations, yielding $\bar{\rho}_j(\phi) = \frac{\bar{a} \bar{b}}{\sqrt{\bar{a}^2 \sin^2(\phi) + \bar{b}^2 \cos^2(\phi)}}$ with $\bar{b}_j(\phi) = (\bar{a}^2 - \bar{b}^2) \sin^2(\phi) + \bar{b}_0^2 \cos^2(\phi) + \bar{a}^2 \cos^2(\phi - \phi_0) + 2\bar{a}\bar{b} \sin(\phi - \phi_0)$, where $a$ and $b$ denote the major and minor semiaxis of the ellipse, respectively, and $\phi_0$ is the rotation of the major semiaxis relative to $\phi_0$.

2 A path is defined as a continuously differentiable locus of points on $\mathbb{R}^2$ connecting two points.

---

3 The superscripts “s” and “d” are used to denote “static” and “dynamic” obstacles, respectively.
\[ \Omega \subseteq \mathbb{R}^{2N} \times \mathbb{R}^{2N} \text{ of the origin such that the dynamic strategies} \]

\[
u_i = -\tilde{x}_i \left( \frac{\alpha_i + \beta_i \gamma i (\xi)}{\beta_i} + N_{ij}(\tilde{x}_j - \tilde{x}_i) \right), \]

\[
\dot{\xi} = -k \sum_{i=1}^{N} \left( \frac{\alpha_i + \beta_i \gamma i (\xi)}{\beta_i} + N_{ij}(\tilde{x}_j - \tilde{x}_i) \right) \left( \frac{\beta_i \partial g_i(\xi)}{\partial \xi} + \beta_i \partial g_i(\xi) \right) - R_i(\tilde{x} - \tilde{\xi}) \right). \tag{16} \]

**Proof.** To begin with, note that the coordinates of the centre of mass of each robot may be written as \( p_i = (X_i + a_i \cos(\theta_i), Y_i + a_i \sin(\theta_i)) \). Differentiating the above relation with respect to time, the dynamics of the centre of mass of each agent is described by the equations

\[
\dot{p}_i = \left[ \begin{array}{c}
\cos(\theta_i) \\
\sin(\theta_i)
\end{array} \right] - a_i \sin(\theta_i) \left[ \begin{array}{c}
\nu_i \\
\omega_i
\end{array} \right] \Delta T_i(\theta_i) \left[ \begin{array}{c}
\nu_i \\
\omega_i
\end{array} \right]. \tag{17}
\]

The selection \( \nu_i \) and \( \omega_i \) as in (16) yields \( \dot{p}_i = k (x_i - p_i) + \dot{x}_i \). Let \( \dot{e}_i(t) = p_i(t) - x_i(t), \) i.e. \( \dot{e}_i \) is the error between the centre of mass of the \( i \)-th robot and the corresponding virtual agent. Taking the time derivative of the error we obtain \( \dot{e}_i = -k \dot{e}_i \). Note that the initial conditions are such that \( \dot{e}_i(0) = 0 \). It follows that \( e_i(t) = 0 \) for all \( t \geq 0 \). Hence, by definition of \( u_i \), \( p_i \) evolve on a collision-free trajectory towards the desired target position.

**Remark 3.** The initial conditions, \( p_i(0) = x_i(0) \), for \( i = 1, \ldots, N \) are such that \( e_i(0) = 0 \) and as a consequence \( e_i(t) = 0 \) for all \( t \geq 0 \).

However, since \( \dot{e}_i = -k \dot{e}_i \), if a “buffer” \( \delta \tau_i \) is included in the safety radius, i.e. \( \tau_i = \tau_i + \delta \tau_i \), any initial conditions satisfying \( |e_i(0)| < \delta \tau_i \), \( i = 1, \ldots, N \), is such that the WMRs reach their targets without collisions occurring. This may be useful for scenarios in which the postural measurements are imperfect.

**Remark 4.** WMRs described by unicycle dynamics are considered in this paper. However, the virtual agents can be exploited to generate a collision-free trajectory plan which can be tracked with zero error through the design of controllers similar to those in Theorem 2 for any mobile robots with dynamics satisfying certain (possibly partial) feedback linearisability assumptions.

## 5. Simulations

A variety of illustrative examples are presented in this section: each example is considered separately in what follows.\(^4\) In all cases the differential game corresponding to the problem associated with the virtual agents is solved using Theorem 1. This is then used to construct the dynamic controllers for the WMRs following Theorem 2. The notation \( \tilde{\xi} = [\tilde{\xi}_1, \ldots, \tilde{\xi}_N] \), where \( \tilde{\xi}_i \in \mathbb{R}^2 \) for \( i = 1, \ldots, N \), is used in some examples. The parameters associated with the dynamic controllers (16) for the WMRs are taken to be \( \kappa = 1 \) and \( a_i = 1 \), for \( i = 1, \ldots, N \). In all examples, in line with the condition for Theorem 2 to hold, the initial conditions of the WMRs coincide with the initial conditions of the virtual agents, i.e. \( p_i(0) = x_i(0) \), for \( i = 1, \ldots, N \). For the collision avoidance functions (11) the parameter \( \epsilon = 1 \) has been used.

### 5.1. Two robots exchanging positions

In this example we consider the simple scenario in which two WMRs, i.e. \( N = 2 \), on a plane, without any obstacles, are to switch positions without colliding into one another. Consider the

\(^4\) This assumption merely implies that the centres of mass of the robots coincide with the initial positions of the virtual agents. This condition is easily satisfied since the virtual agents can be interpreted as a design tool for the dynamics and initial conditions can be freely assigned.

\(^5\) Videos illustrating the simulations in Section 5.7 can be downloaded from [https://www.dropbox.com/sh/cdah7n5uzg916b/AACpboFZPQhu2_nb4tdP1YHja7?dl=0](https://www.dropbox.com/sh/cdah7n5uzg916b/AACpboFZPQhu2_nb4tdP1YHja7?dl=0)
Fig. 2. Trajectories of the first (black line) and the second (grey line) WMR with $x_1(0) = [8, 8]^T$, $x_2(0) = [-8, -8]^T$, $x_i = [-8, -8]^T$ and $x_2 = [8, 8]^T$. The trajectory for $\xi(0) = [8, -3, -3.8]^T$ (solid lines), $\xi(0) = [20, -10, -10, 20]^T$ (dashed lines) and $\xi(0) = [-2, -2, -50, 1]^T$ (dotted lines). The circular markers indicate the initial positions of the WMRs.

Fig. 3. Trajectories of the first (black line) and the second (grey line) WMR with $x_1(0) = [8, 8]^T$, $x_2(0) = [-8, -8]^T$, $x_i = [-8, -8]^T$ and $x_2 = [8, 8]^T$. The trajectory for $\xi(0) = [-8, 3.3, -8]^T$ (solid lines), $\xi(0) = [-20, 10, 10, -20]^T$ (dashed lines) and $\xi(0) = [-2, 2.1, -50]^T$ (dotted lines). The circular markers indicate the initial positions of the WMRs.

direction of travel. For the case in which $\xi(0) = [8, -3, -3.8]^T$, the top and middle graphs of Fig. 4 show the time history of the state of the dynamic extensions $\xi_1(t)$ (top) and $\xi_2(t)$ (middle), where solid and dashed lines represent the first and second components of $\xi_i$, $i = 1, 2$, respectively, and the bottom graph shows the time history of the distance between the two WMRs, i.e. $d = ||p_1(t) - p_2(t)||$ (solid line) as well as the quantity $r_1 + r_2$ (dashed line). As can be seen both WMRs reach their targets and both agents manoeuvre themselves while maintaining a minimum distance of $r_1 + r_2$ at all times. The time histories of the control efforts $u_1$ (top) and $u_2$ (bottom) for $i = 1$ (left) and $i = 2$ (right) are shown in Fig. 5.

5.2. Avoiding deadlock

Consider now the case in which $N = 3$. $x_1(0) = [-0.2, -0.2]^T$, $x_2(0) = [0.1, 0.1]^T$, $x_3(0) = [0, 0]^T$, $x_i^0 = [0.1, 0.1]^T$, $x_2^0 = [0, 0]^T$, and $x_i^* = [-0.2, -0.2]^T$. This is a setting similar to the one considered in [33]. Similarly to the previous example it is a scenario in which, intuitively, one may expect a deadlock to occur, as is observed in [33]. The remaining parameters are $r_1 = r_2 = 0.05$, $\alpha_1 = \alpha_2 = 1$, $\beta_1 = \beta_2 = 0.1$, $k = 0.2$, $R_1 = R_2 = I$, and $\gamma_1 = \gamma_2 = 1$. $\xi(0) = [-0.2, -0.2, 0.5, -0.5, 0.4, -0.5]^T$. The trajectories of the first (solid line), the second (dashed line) and the third (dotted line) WMRs are shown in Fig. 6 and it is clear that the WMRs avoid reaching a deadlock and instead converge to their desired targets.

5.3. Two robots avoiding static obstacles

Consider now the scenario in which two WMRs (i.e. $N = 2$) have to perform their manoeuvres in the presence of a static obstacle. The static obstacle is taken to occupy a circular region centred at $(0, 0)$ with radius 2. The parameters relating to the WMRs are $r_1 = r_2 = 1$, $\alpha_1 = \alpha_2 = 1$, $\beta_1 = \beta_2 = 1$, $x_1(0) = [-8, -8]^T$, $x_2(0) = [8, -8]^T$, $x_1^0 = [8, 8]^T$ and $x_2^0 = [-8, 8]^T$. The parameters
for the dynamic strategies are $\epsilon = 1$, $R_1 = R_2 = l$, $k = 1$, $\gamma_1 = \gamma_2 = 0.1$, $\xi(0) = [12, -12]^\top$ and $\hat{\xi}(0) = [12, 12]^\top$.

The grey circular region in Fig. 7 indicates the static obstacle, whereas the trajectories of first and second WMR are shown by the black and grey lines, respectively. The dotted circles illustrate the safety radius about the points at which the WMRs are closest to one another and the arrows indicate the direction of travel. As can be seen the WMRs manoeuvre around the obstacle while avoiding collisions with one another.

5.4. Overtaking

Consider again a scenario with $N = 2$. However, consider the case in which $\alpha_1 \neq \alpha_2$, i.e. the penalty on the control efforts of the two WMRs are not identical and consequently one WMR is faster than the other. In particular we consider the case in which $\alpha_1 = 10$ and $\alpha_2 = 0.01$, namely robot 2 is faster than robot 1. Furthermore $\beta_1 = \beta_2 = 1$ and $r_1 = 1.5$ and $r_1 = 0.5$. The initial position of the WMRs are $x_1(0) = [-35, 0]^\top$ and $x_2(0) = [50, 0]^\top$, whereas the target positions are $x_1^* = [25, 0]^\top$ and $x_2^* = [40, 0]^\top$. Note that the selection of initial and target positions are such that the fast robot overtakes the slow robot. The parameters for the dynamic strategies are $\gamma_1 = \gamma_2 = 2.5$, $R_1 = R_2 = l$, $k = 1$ and $\xi(0) = [50, 0, -14.5]^\top$ and $\hat{\xi}(0) = [0, -14.5]^\top$.

The trajectories of the first (black line) and second (grey line) WMR are shown in Fig. 8. The dotted circles indicate the safety radius at the points along the trajectories at which the WMRs are closest to one another, whereas the arrows indicate the direction of travel. Note that robot 2, which is the fast agent overtakes robot 1 as expected.

5.5. Four robots creating speed lanes

Consider now the case in which $N = 4$, $\alpha_1 = 10$, $\alpha_2 = 1$, $\alpha_3 = 0.1$ and $\alpha_4 = 0.01$, i.e. there are four WMRs and the $i$-th robot is slower than the $j$-th WMR for all $i > j$, where $i = 1, \ldots, 4$ and $j = 1, \ldots, 4$. Furthermore, $\beta_i = 0.1$ and $r_i = 0.5$ for $i = 1, \ldots, 4$. The
initial and target positions of all the WMRs are $x_1(0) = [-5, 0]^	op$, $x_2(0) = [-10, 0]^	op$, $x_3(0) = [-15, 0]^	op$, $x_4(0) = [-20, 0]^	op$ and $x_5 = [5, 0]^	op$, $x_6 = [10, 0]^	op$, $x_2 = [15, 0]^	op$, $x_4 = [20, 0]^	op$, respectively. The parameters for the dynamic strategies are $\gamma_i = 1$ and $\delta_i = 1$, for $i = 1, \ldots, 4$, $k = 0.1$, $\xi_1(0) = [0, 0]^	op$, $\xi_2(0) = [0, -5]^	op$, $\xi_3(0) = [0, -10]^	op$, $\xi_4(0) = [0, -15]^	op$.

Fig. 9 shows the trajectories of the four WMRs, i.e. $x_1$ (black, solid line), $x_2$ (grey, dashed line), $x_3$ (grey, dash-dotted line) and $x_4$ (grey, dotted line). The arrows indicate the direction of travel. Note that the faster robots manoeuvre themselves towards their targets by moving around the slower WMRs.

5.6. Robots switching positions on a circle

Consider the case in which $N = 8$ WMRs are positioned, with equal spacing, along the circumference of a circle and their target positions are such that robots which are opposite to one another should switch positions. This example illustrates the efficiency of the proposed method for a scenario in which there are high levels of conflicts of interests between the robots.

The initial and target positions are selected as $x_1(0) = x_2 = [0, 8]$, $x_3(0) = x_4 = [0, -8]$, $x_1(0) = x_2 = [8, 0]$, $x_3(0) = x_4 = [-8, 0]$, $x_5(0) = x_6 = [2\sqrt{3}, 2\sqrt{3}]$, $x_7(0) = x_8 = [2\sqrt{3}, -2\sqrt{3}]$, $x_7(0) = x_8 = [-2\sqrt{3}, -2\sqrt{3}]$ and $x_8(0) = x_7 = [2\sqrt{3}, -2\sqrt{3}]$. Consider the case in which $\alpha_i = 1$ and $\beta_i = 0.1$, for $i = 1, \ldots, 8$ and the following parameters associated with the dynamic controllers have been used: $k = 0.025$, $\delta_i = 1$, for $i = 1, \ldots, 9$, and $\gamma_i = [0, 20]$, $\xi_1(0) = [0, 20]$, $\xi_2(0) = [50, -20]$, $\xi_3(0) = [-50, 20]$, $\xi_4(0) = [-50, -20]$, $\xi_5(0) = [-50, 20]$ and $\xi_6(0) = [50, -20]$. $\xi_7(0) = [50, 20]$ and $\xi_8(0) = [50, -20]$.

The resulting trajectories of the robots are shown in Fig. 10, where the trajectories are shown pairwise for clarity of presentation. Namely, the trajectories of $x_1$ (black line) and $x_2$ (grey line) are shown in the top left plot, the trajectories of $x_3$ (black line) and $x_4$ (grey line) are shown in the top right plot, the trajectories...
of $x_5$ (black line) and $x_6$ (grey line) are shown in the bottom left plot and the trajectories of $x_7$ (black line) and $x_8$ (grey line) are shown in the bottom right plot. In all plots the grey, dotted lines indicate the circle on which the initial and target positions lie, the circular markers indicate the initial and target positions, whereas arrows indicate the directions of travel.

### 5.7. Robots forming a cordon

In this final example consider the case in which $N = 9$. Consider the scenario in which robots 1, ..., 8, initially at their target positions, form a circular cordon and the ninth WMR, initially outside this circle, has its target position at the centre of the cordon. In particular the initial and target positions of WMRs 1, ..., 8 are $x_1(0) = x_1^* = [0, 4]^\top$, $x_2(0) = x_2^* = [2\sqrt{2}, 2\sqrt{2}]^\top$, $x_3(0) = x_3^* = [4, 0]^\top$, $x_4(0) = x_4^* = [2\sqrt{2}, -2\sqrt{2}]^\top$, $x_5(0) = x_5^* = [0, -4]^\top$, $x_6(0) = x_6^* = [-2\sqrt{2}, -2\sqrt{2}]^\top$, $x_7(0) = x_7^* = [-4, 0]^\top$ and $x_8(0) = x_8^* = [2\sqrt{2}, 2\sqrt{2}]^\top$, whereas the initial and target positions of the 9-th WMR are $x_9(0) = [10, 0]^\top$ and $x_9^* = [0, 0]^\top$, respectively. Consider the case in which $\alpha_i = 1$ and $\beta_i = 0.1$, $r_i = 1$, for $i = 1, \ldots, 9$ and the following parameters associated with the dynamic controllers have been used: $k = 0.4$, $R_i = I$, for $i = 1, \ldots, 9$, and $\xi_i(0) = [0, 0]^\top$, for $i = 1, \ldots, 6$, $\xi_7(0) = [-5, -10]^\top$, $\xi_8(0) = [-10, 5]^\top$ and $\xi_9(0) = [-5, 10]^\top$

The trajectories of the robots are shown in Fig. 11 in which the numbering indicates the individual WMRs. Robots 1, ..., 6 remain almost stationary whereas the seventh and eighth robots (dark grey and light grey lines, respectively) move to make room for the ninth robot (black line). The black dashed circles denote the radii of the stationary robots. The dark grey dotted circles denote the radii of the seventh and ninth robots at the points along their trajectories at which the distance between the two robots is at a minimum. The light grey dotted circle denote the same quantity for the eighth and ninth robots. The solid circular markers denote the initial and final positions of robots 1, ..., 8, and the final position of the ninth robot, whereas the solid square marker denotes

![Fig. 12. Time histories of the x-components (top) and y-components (bottom) of the seventh (dark grey lines), eighth (light grey lines) and ninth (black lines) robots.](image1)

![Fig. 13. The positions of each WMR at six successive time instants. Small circular markers denote the positions of robots 1, ..., 6, whereas the larger dark grey, light grey and black circular markers denote the positions of the seventh, eighth and ninth robots, respectively.](image2)
the initial position of the ninth robot. The arrows indicate the direction of travel. The time histories of the x-components (top) and y-components (bottom) of the positions of the seventh (dark grey lines), eighth (light grey lines) and ninth (black lines) are shown in Fig. 12. The positions of the WMRs at six successive time instants are shown in Fig. 13 where the positions of robots 1…6 are denoted by the small black circular markers whereas the positions of robots 7 (dark grey marker), 8 (light grey marker) and 9 (black marker) are denoted by the larger circular markers. The dashed circular lines show the radii of each WMR.

6. Conclusion

The multi-agent collision avoidance problem for a team of WMRs satisfying unicycle dynamics is considered in this paper. The problem of manoeuvring the robots from their initial positions to predefined targets while avoiding collisions is solved in two steps through the introduction of virtual agents and the formulation of a differential game. The approach allows for the systematic construction of feedback strategies for the WMRs which ensure locally that the robots reach their targets while avoiding collisions, provided the initial and target configurations are feasible. The approach is illustrated on a series of simulations including challenging scenarios involving potential deadlocks. A sensitivity analysis with respect to the selection of the initial condition $\xi(0)$ is also provided.

Directions of future research include taking into consideration distributed settings in which the communication between agents is limited, possibly through further developments based on the preliminary results relating to distributed differential games presented in [21] and [16] as mentioned in Remark 2. Experimental validation of the results will also be sought through the implementation of the control laws on teams of small-scale robots. It is of interest to extend the results presented herein to enable operation in uncertain environments in which the agents should create a map of a partially or fully unknown environment while avoiding collisions. It is also of interest to take into consideration dynamic targets for the agents. Robust collision avoidance, where inter-agent collisions should be avoided in the presence of disturbances acting on the robots, may also be considered.

Acknowledgement

The work of M. Mylvaganam is partially supported by the “STABLE-NET” research project funded by the EPSRC’s Energy Programme (Contract no.: EP/L014343/1).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ejcon.2017.11.005.

References

[1] G. Arslan, J.R. Marden, J.S. Shamma, Autonomous vehicle-target assignment: a game theoretical formulation, ASME J. Dyn. Syst. Meas. Control 129 (2007) 584–596.
[2] J. Cortes, S. Martinez, T. Karatas, F. Bullo, Coverage control for mobile sensing networks, IEEE Trans. Robot. Autom. 20 (2) (2004) 243–355.
[3] M. De Gennaro, A. Jadadbeibe, Formation control for a cooperative multi-agent system using decentralized navigation functions, in: Proceedings of the American Control Conference, 2006.
[4] A. De Luca, G. Oriolo, M. Vendittelli, Control of Wheeled Mobile Robots: An Experimental Overview, Ramsete, Springer, 2001, pp. 181–226.
[5] C. De Wit, O. Sardanel, Exponential stabilization of mobile robots with nonholonomic constraints, IEEE Trans. Autom. Control 37 (11) (1992) 1791–1797.
[6] D. Dimarogonas, M. Zavlanos, S. Loizou, K. Kyriakopoulos, Decentralized motion control of multiple holonomic agents under input constraints, in: Proceedings of IEEE Conference on Decision and Control, 2003.
[7] R. Ghosh, C. Tomlin, Maneuver design for multiple aircraft conflict resolution, in: Proceedings of the American Control Conference, Vol. 1, 2000, pp. 672–676.
[8] B. Hugue, A.R. Rahmani, M.B. Egerstedt, Biologically inspired confinement of multi-agent systems, ASME J. Dyn. Syst. Meas. Control 3 (2011) 213–224.
[9] L. Hussein, D.M. Stipanovic, Effective coverage control for mobile sensor networks with guaranteed collision avoidance, IEEE Trans. Control Syst. Technol. 15 (4) (2007) 642–657.
[10] L. Hussein, D.M. Stipanovic, Y. Wang, Reliable coverage control using heterogeneous vehicles, in: Proceedings of the 46th IEEE Conference on Decision and Control, 2007, pp. 6142–6147, doi:10.1109/CDC.2007.4434510.
[11] G. Lettmann, J. Skowronski, Avoidance control, J. Optim. Theory Appl. Vol. 23(4)581–591.
[12] N.E. Leonard, E. Fiorelli, Virtual leaders, artificial potentials and coordinated control of groups, in: Proceedings of the 40th Conference on Decision and Control, 2001, pp. 2968–2973.
[13] W. Lin, Distributed UAV formation control using differential game approach, Aerospace Sci. Technol. 35 (2014) 54–62.
[14] J.R. Marden, J.S. Shamma, Game theory and distributed control, Handbook of Game Theory with Economic Applications, 4, Elsevier, 2015, pp. 891–899.
[15] M. Mesbahi, M. Egerstedt, Graph Theoretic Methods in Multiagent Networks, 1st ed., Princeton University Press, 2010.
[16] T. Mylvaganam, A game theoretic approach to distributed control of homogenous multi-agent systems, in: Proceedings of the 56th IEEE Conference on Decision and Control, 2017, pp. 6217–6222.
[17] T. Mylvaganam, A. Astolfi, Approximate optimal monitoring: preliminary results, in: Proceedings of the American Control Conference, 2012, pp. 4745–4750.
[18] T. Mylvaganam, A. Astolfi, Approximate optimal monitoring, in: Proceedings of the 14th European Control Conference, 2014, pp. 1199–1204.
[19] T. Mylvaganam, A. Astolfi, Control of microgrids using a differential game theoretic framework, in: Proceedings of the 54th IEEE Conference on Decision and Control, 2015, pp. 5839–5844.
[20] T. Mylvaganam, A. Astolfi, A differential game approach to formation control for a team of agents with one leader, in: Proceedings of the American Control Conference, 2015, pp. 1469–1474.
[21] T. Mylvaganam, A. Astolfi, Towards a systematic solution for differential games with limited communication, in: Proceedings of the American Control Conference, 2016.
[22] T. Mylvaganam, M. Sassano, A. Astolfi, A constructive differential game approach to collision avoidance in multi-agent systems, in: Proceedings of the American Control Conference, 2014, pp. 311–316.
[23] T. Mylvaganam, M. Sassano, A. Astolfi, Constructive $\epsilon$-Nash equilibria for nonzero-sum differential games, IEEE Trans. Autom. Control 60 (4) (2015) 950–965.
[24] T. Mylvaganam, M. Sassano, A. Astolfi, A differential game approach to multi-agent collision avoidance, IEEE Trans. Autom. Control 62 (8) (2017) 4228–4235.
[25] B. Nabet, N.E. Leonard, I. Couzin, S. Levin, Dynamics and decision making in animal group motion, J. Nonlinear Sci. 19 (2009) 399–405.
[26] P. Ogren, E. Fiorelli, N.E. Leonard, Formations with a mission: stable coordination of vehicle group maneuvers, Proceedings of Symposium on Mathematical Theory of Networks and Systems (2002) 1–22.
[27] P. Ogren, E. Fiorelli, N.E. Leonard, Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment, IEEE Trans. Autom. Control 49 (8) (2004) 1292–1302.
[28] R. Olafati-Saber, J. Fax, R. Murray, Consensus and cooperation in networked multi-agent systems, Proc. IEEE 95 (1) (2007) 215–233.
[29] D.A. Paley, N.E. Leonard, R. Sepulchre, D. Grünbaum, J.K. Parrish, Oscillator models and collective motion, IEEE Control Syst. Mag. 27 (4) (2007) 89–105.
[30] A. Ranter, Using Game Theory for Distributed Control Engineering, Technical Report, Department of Automatic Control, Lund Institute of Technology, Lund University, 2008.
[31] E. Rimon, D. Koditschek, Exact robot navigation using artificial potential functions, IEEE Trans. Robot. Autom. 8 (5) (1992) 501–518.
[32] H. Su, X. Wang, Z. Lin, Flocking of multi-agents with a virtual leader, IEEE Trans. Autom. Control 54 (2) (2009) 293–307.
[33] H. Tanner, A. Boddu, Multiagent navigation functions revisited, IEEE Trans. Robot. Autom. 28 (6) (2012) 1346–1359.
[34] H. Tanner, A. Kumar, Towards decentralization of multi-robot navigation functions, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2005.
[35] C. Tomlin, G. Pappas, S. Sastry, Conflict resolution for air traffic management: a study in multiagent hybrid systems, IEEE Trans. Autom. Control 43 (4) (1998) 558–571.
[36] F. Zhang, N.E. Leonard, Cooperative filters and control for cooperative exploration, IEEE Trans. Autom. Control 55 (3) (2010) 640–663.