Collision Avoidance of 3D Rectangular Planes by Multiple Cooperating Autonomous Agents

Jai Raj, Krishna Raghuwaiya, and Jito Vanualailai

1School of Computing Information & Mathematical Sciences, The University of the South Pacific, Suva, Fiji
2School of Education, The University of the South Pacific, Suva, Fiji

Correspondence should be addressed to Krishna Raghuwaiya; raghuwaiya_k@usp.ac.fj

Received 14 February 2020; Revised 23 September 2020; Accepted 8 October 2020; Published 28 October 2020

1. Introduction

The motion planning and control (MPC) of mobile robots or agents is a challenging task and an interesting problem attracting considerable attention to the robotic community over the last couple of decades. The design of a particular robotic system and motion planning are usually treated independently [1]. Typically, MPC algorithms are applied to systems with fully fixed geometric and kinematic features, while the system design in robotics takes into account robustness, stiffness, workspace volume, obstacle avoidance schemes, and other performance features. The principle goal for any MPC problem is to find the most optimum design to optimise the motion between given configurations [2–7]. In an MPC problem, multiple robots are favoured as they are able to cooperate for faster and more efficient results [4, 5, 8–10], including other fields where multiagent operations are always preferred [11].

Path planning or MPC algorithms for mobile robots operating in an environment cluttered with obstacles are usually grouped according to the methodologies used to generate the geometric path, namely, the road map techniques, cell decomposition algorithms, and artificial potential field (APF) methods [4, 12]. These path planning algorithms have a common objective, which is to find the shortest and most optimal geometric path taking into account the moving objects and obstacles in the workspace [13–15]. While the calculation of a hindrance-free way may take care of numerous significant issues in industrial settings where the robot may move cautiously, it is inadequate and practically futile when the robot needs to move at sensibly high speeds, for example, multiple mobile robots navigating through dynamic cluttered situations and autonomous vehicles navigating in a highway traffic situation.

In this research article, we use the Lyapunov controllers, constructed via the Lyapunov-based Control Scheme...
that require every pair of adjacent states to be on the same side of an obstacle, as shown in Figure 1(b) [23].

Furthermore, comparing Arantes et al. approach to the MDT, the latter results in a smooth, continuous path for the avoidance of irregular shaped (rectangular plane) obstacles. An illustration of the MDT for the avoidance of a rectangular plane is shown in Figure 1(c).

The main contributions of this paper are summarised as follows:

1. The design of the velocity algorithm for a point-mass mobile robot which is based on a Lyapunov function that acts as an energy function of the system. The velocity algorithm ensures safe, collision-free trajectories that converge to the intended target.

2. The design of the velocity algorithm for the point-mass mobile robot which is based on the development of a Lyapunov function that acts as an energy function of the system. The velocity algorithm applied here is altogether not quite the same as the ones in the literature. Consistently enduring velocities are utilised; nonetheless, the robot needs to stop after it has accomplished its objective. This stop should not be unexpected by a truncation of speed; rather, the robot should slow down its motion and afterward come to rest. The velocity algorithm and the objective target intended for the robot guarantee a protected and safe stop at the goal objective and furthermore guarantee that the robot stays there.

3. A three-dimensional rectangular-plane obstacle avoidance scheme using the MDT. While in motion, the distance between the point-mass robot and the closest point on the surface of the wall is computed and the point-mass robot avoids this point on the surface of the wall, resulting in the avoidance of the entire wall. In addition, we only consider the wall closest to the point-mass robot en route to its target. Subsequently, our obstacle avoidance scheme is more straightforward contrasted with, for instance, the avoidance schemes used in the artificial potential strategies where all of the obstacles are considered in parallel [4, 17].

4. Stability analysis pertaining to the kinodynamic system. We use the Direct Method of Lyapunov to carry out the stability analysis, proving that the equilibrium point of the system, representing the target of a point mass, is stable.

The paper is organised as follows: In Section 2, we define the kinematic model of the point-mass robot; in Section 3, the APF functions are defined; in Section 4, the Lyapunov function is constructed and the robust nonlinear continuous control laws for the mobile robot are extracted; in Section 5, the stability of the system is discussed; in Section 6, the simulation results are presented to show the robustness and effectiveness of the proposed control inputs and followed by conclusion in Section 7.
2. Modelling a Point-Mass Robot or Agent in 3D

The modelling process of a robotic system involves the conceptualisation of the problem, residing on the abstraction level. Simulation, however, mainly focuses on the implementation of the execution of the model to study the behavior and performance of an actual or theoretical system. This section proposes a simple kinematic model for the moving point-mass robot, an abstraction of a simple form of a robotic system. A two-dimensional schematic representation of a point-mass robot with and without rectangular obstacle avoidance is shown in Figure 2. We begin with the following definition.

**Definition 1.** A point mass, \( P_i \), is a sphere of radius \( r_{p0} \) and centred at \((x_i(t), y_i(t), z_i(t)) \in \mathbb{R}^3 \) for \( t \geq 0 \). That is, it is the set

\[
P_i = \{(Z_1, Z_2, Z_3) \in \mathbb{R}^3 : (Z_1 - x_i)^2 + (Z_2 - y_i)^2 + (Z_3 - z_i) \leq r_{p0}^2 \}.
\]

At time \( t \geq 0 \), the instantaneous velocity of the point mass will be given as \((v_i(t), w_i(t), u_i(t)) = (\dot{x}_i(t), \dot{y}_i(t), \dot{z}_i(t))\). Assuming the initial conditions, a system of the first-order ODEs governing \( P_i \) is

\[
\begin{align*}
\dot{x}_i &= v_i(t), \\
\dot{y}_i &= w_i(t), \\
\dot{z}_i &= u_i(t), \\
x_{0i} &= x_i(t_0), \\
y_{0i} &= y_i(t_0), \\
z_{0i} &= z_i(t_0),
\end{align*}
\]

for \( i = 1, \ldots, n \). Let \( x_i = (x_i, y_i, z_i) \in \mathbb{R}^3 \) and \( x = (x_1, \ldots, x_n) \in \mathbb{R}^{3n} \).

Next, we will formulate the components that form the Lyapunov function, essentially the attractive and repulsive potential field functions.

3. Construction of the APF Functions

In this section, we construct the components of the Lyapunov function. We assume that \( P_i \) has a priori knowledge of the entire workspace. The principle objective is to construct the Lyapunov function from which we derive the nonlinear velocity control inputs \( v_i(t), w_i(t), \) and \( u_i(t) \) for \( i = 1, \ldots, n \) such that \( P_i \) navigates and reaches its target configuration, avoiding any obstacle, whether fixed, moving, or artificial, while it is in motion. The design of the nonlinear control inputs is captured in Figure 3, clearly illustrating the roles of the individual components in the design of the control scheme.

3.1. Attractive Potential Field Functions. We introduce basic mathematical notions to design and construct attractive functions for target attraction for \( P_i \).

3.1.1. Attraction to Target Function. To initiate movement and ensure convergence, we propose to have a target \( T_i \) for each of the point-mass mobile robots \( P_i \). The convergence of \( P_i \) to \( T_i \) will be guaranteed by the Lyapunov function.

**Definition 2.** The assigned target for the point-mass mobile robot of \( P_i \) is a sphere with centre \((\tau_{i1}, \tau_{i2}, \tau_{i3})\) and radius \( r_{T_i} \). That is, it is the set

\[
T_i = \{(Z_1, Z_2, Z_3) \in \mathbb{R}^3 : (Z_1 - \tau_{i1})^2 + (Z_2 - \tau_{i2})^2 + (Z_3 - \tau_{i3})^2 \leq r_{T_i}^2 \}.
\]

The next function will measure the Euclidean distance of \( P_i \) from its designated target \( T_i \) at time \( t \geq 0 \). It will be used as an attraction function:

\[
V_i(x) = \frac{1}{2} \left[ (x_i - \tau_{i1})^2 + (y_i - \tau_{i2})^2 + (z_i - \tau_{i3})^2 \right].
\]

An illustration of the total potentials for the target attraction function is shown in Figure 4(a), while Figure 4(b) shows the analogous contour plot generated over a workspace \( 0 < Z_1 < 100 \) and \( 0 < Z_3 < 100 \) for the point-mass mobile robot. For simplicity, we consider the target function in a 2-dimensional environment. The disk-shaped target for the point-mass mobile is fixed at \((\tau_{01, 02}) = (50, 50)\) with a radius of \( r_{p0} = 1 \).

3.1.2. Auxiliary Function. In the MPC problem, \( P_i \) starts from an initial position and navigates towards its target. While navigating, the motion of \( P_i \) is such that it will avoid
all obstacles, whether it is fixed or moving, with respect to the kinodynamic constraints that are tagged with the robotic system including the constraints on velocity and angles before reaching its objective target. Once it has reached the target, it essentially means that it has accomplished the task that was given to the robot, and hence it needs to stop at the target configuration. Intuitively, this means that the energy of the robotic system needs to be zero at the target configuration; that is, the nonlinear controllers need to vanish at the target. Hence, to achieve this and to ensure the convergence of $\mathcal{P}_i$ to its target configuration, we consider the auxiliary function of the form

$$G_i(x) = V_i(x).$$

### 3.2. Repulsive Potential Field Functions

#### 3.2.1. Workspace Boundaries

We shall confine the motion of $\mathcal{P}_i$ in a cuboid constrained by the dimensions $\eta_1 \times \eta_2 \times \eta_3$. Since the motion is confined within these boundary walls, the walls are hence treated as fixed obstacles. Therefore, for the avoidance of these walls, the following functions are proposed:

$$WS_{i1}(x) = x_i - r\eta_i, WS_{i2}(x) = \eta_2 - (y_i + r\eta_i),$$
$$WS_{i3}(x) = \eta_1 - (x_i + r\eta_i), WS_{i4}(x) = y_i - r\eta_i,$$
$$WS_{i5}(x) = z_i - r\eta_i, WS_{i6}(x) = \eta_3 - (z_i + r\eta_i),$$

for $i = 1, \ldots, n$, which are positive within the rectangular cuboid.

#### 3.2.2. Rectangular-Plane Obstacle Avoidance

Disks in 2D and spheres in 3D are the simplest models of obstacles. However, they encompass extraspaces that are not needed for avoidance. For example, enclosing a rod-like structure within a sphere introduces spaces that need not be avoided. As an illustration, Figure 5(a) shows the contour plot of the total potentials and the corresponding collision-free path of a point mass over the workspace $0 < Z_1 < 40$ and $0 < Z_2 < 40$ encompassing a rod-shaped obstacle, while Figure 5(b) showcases the contour plot of the total potentials and the resulting path if the rod is replaced by a disk-shaped obstacle. The initial and final coordinates of the rod are $(10, 20)$ and $(30, 20)$, respectively. The disk portrayed of the rod has a diameter of 10, which matches the length of the rod, and is centred at $(20, 20)$. The path generated in the presence of the rod-shaped obstacle is optimal in terms of the distance traversed since the obstacle space is small in contrast to the disk portrayed of the rod. Therefore, in this article, we introduce rectangular obstacles.

To avoid the rectangular obstacles via the MDT, the surface wall of the rectangular plane is classified as a fixed obstacle. Let us fix $\ell = 1, \ldots, m, m \in \mathbb{N}$ rectangular-plane-shaped obstacles within the workspace. An illustration of a rectangular plane is showcased in Figure 6, which we shall use to derive the new mathematical equations for its avoidance. Three points are sufficient for deriving the saturation functions and hence designing the rectangular-plane avoidance functions. We begin with the following definition.

**Definition 3.** Assume that the three-dimensional $(\ell)$th planar obstacle has the three coordinates $(a_{1\ell}, b_{1\ell}, c_{1\ell})$, $(a_{2\ell}, b_{2\ell}, c_{2\ell})$, and $(a_{3\ell}, b_{3\ell}, c_{3\ell})$, $\ell = 1, \ldots, m, m \in \mathbb{N}$ (see Figure 6). A single point in the plane is defined by

$$SP_{i\ell} = (Z_1, Z_2, Z_3) = (P_{X_{i\ell}}, P_{Y_{i\ell}}, P_{Z_{i\ell}}) \in \mathbb{R}^3.$$  

Then the plane can be precisely described by the set

$$P_{\ell} = \{SP_{1\ell}, SP_{2\ell}, \ldots, SP_{m\ell}\} \in \mathbb{R}^{3m}.$$  

Then the set of $\ell$ planes, $\ell \in \bar{m}$, is

$$CP = \{P_1, P_2, \ldots, P_m\} \in \mathbb{R}^{m_{\text{min}}},$$

where $P_{X_{i\ell}} = a_{1\ell} + \lambda_{i\ell}(a_{2\ell} - a_{1\ell}) + \lambda_{i\ell}(a_{3\ell} - a_{1\ell}), P_{Y_{i\ell}} = b_{1\ell} + \lambda_{i\ell}(b_{2\ell} - b_{1\ell}) + \lambda_{i\ell}(b_{3\ell} - b_{1\ell}),$ and $P_{Z_{i\ell}} = c_{1\ell} + \lambda_{i\ell}(c_{2\ell} - c_{1\ell}) + \lambda_{i\ell}(c_{3\ell} - c_{1\ell})$ are the parametric representation for $0 \leq \lambda_{i\ell} < 1, \ell = 1, \ldots, m$, and $i = 1, \ldots, n$.

The MDT necessitates that we identify the closest point on each of $\ell$, the rectangular plane measured from the centre of $\mathcal{P}_i$. We compute the minimum Euclidian distance from the centre of $\mathcal{P}_i$ to the surface of the $\ell$th rectangular plane.

The avoidance of the closest point of the surface of the rectangular plane at any time $t \geq 0$ results in the avoidance of the entire plane by $\mathcal{P}_i$. Minimising the Euclidian distance between the points $(x_i, y_i, z_i)$, which is the centre of $\mathcal{P}_i$ and the $\ell$th rectangular plane, yields
The saturation functions are $\lambda_{\ell_{1,2}}: \mathbb{R}^3 \rightarrow [0, 1] \subseteq \mathbb{R}$, defined as

$$
\lambda_{\ell_{1,2}}(\bar{c}_{\ell_{1,2}}, \bar{d}_{\ell_{1,2}}, \bar{h}_{\ell_{1,2}}) = \begin{cases} 0, & \text{if } \lambda_{\ell_{1,2}} < 0, \\ \lambda_{\ell_{1,2}}, & \text{if } 0 \leq \lambda_{\ell_{1,2}} \leq 1, \\ 1, & \text{if } \lambda_{\ell_{1,2}} > 1. 
\end{cases}
$$

The new obstacle avoidance functions are therefore of the form

$$
\text{RP}_{i\ell}(x) = \frac{1}{2} \left[ (x_i - P_{x,i})^2 + (y_i - P_{y,i})^2 + (z_i - P_{z,i})^2 - (r_{P_i})^2 \right],
$$

for $i = 1, \ldots, n$ and $\ell = 1, \ldots, \bar{m}$. The function $\text{RP}_{i\ell}(x)$ is the measure of the distance between the closest point on the surface of the $\ell$th rectangular-plane-shaped obstacle and the centre of $\mathcal{P}_i$.  

3.2.3. Moving Obstacles. While in motion, each moving robot itself becomes a moving obstacle to every other mobile robot. For $\mathcal{P}_i$ to avoid $\mathcal{P}_j$, we consider the following function:

$$
\text{MO}_{ij}(x) = \frac{1}{2} \left[ (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 - (r_{P_i} + r_{P_j})^2 \right],
$$

where

$$
\begin{align*}
\lambda_{\ell_{1}} &= (x_i - a_{\ell_{1}})\bar{c}_{\ell_{1}} + (y_i - b_{\ell_{1}})\bar{d}_{\ell_{1}} + (z_i - c_{\ell_{1}})\bar{h}_{\ell_{1}}, \\
\lambda_{\ell_{2}} &= (x_i - a_{\ell_{2}})\bar{c}_{\ell_{2}} + (y_i - b_{\ell_{2}})\bar{d}_{\ell_{2}} + (z_i - c_{\ell_{2}})\bar{h}_{\ell_{2}},
\end{align*}
$$

\begin{align*}
\bar{c}_{\ell_{1}} &= \frac{(a_{\ell_{2}} - a_{\ell_{1}})}{(a_{\ell_{2}} - a_{\ell_{1}})^2 + (b_{\ell_{2}} - b_{\ell_{1}})^2 + (c_{\ell_{2}} - c_{\ell_{1}})^2}, \\
\bar{d}_{\ell_{1}} &= \frac{(b_{\ell_{2}} - b_{\ell_{1}})}{(a_{\ell_{2}} - a_{\ell_{1}})^2 + (b_{\ell_{2}} - b_{\ell_{1}})^2 + (c_{\ell_{2}} - c_{\ell_{1}})^2}, \\
\bar{h}_{\ell_{1}} &= \frac{(c_{\ell_{2}} - c_{\ell_{1}})}{(a_{\ell_{2}} - a_{\ell_{1}})^2 + (b_{\ell_{2}} - b_{\ell_{1}})^2 + (c_{\ell_{2}} - c_{\ell_{1}})^2}, \\
\bar{c}_{\ell_{2}} &= \frac{(a_{\ell_{3}} - a_{\ell_{1}})}{(a_{\ell_{3}} - a_{\ell_{1}})^2 + (b_{\ell_{3}} - b_{\ell_{1}})^2 + (c_{\ell_{3}} - c_{\ell_{1}})^2}, \\
\bar{d}_{\ell_{2}} &= \frac{(b_{\ell_{3}} - b_{\ell_{1}})}{(a_{\ell_{3}} - a_{\ell_{1}})^2 + (b_{\ell_{3}} - b_{\ell_{1}})^2 + (c_{\ell_{3}} - c_{\ell_{1}})^2}, \\
\bar{h}_{\ell_{2}} &= \frac{(c_{\ell_{3}} - c_{\ell_{1}})}{(a_{\ell_{3}} - a_{\ell_{1}})^2 + (b_{\ell_{3}} - b_{\ell_{1}})^2 + (c_{\ell_{3}} - c_{\ell_{1}})^2},
\end{align*}
$$

\begin{align*}
\bar{c}_{\ell_{1,2}} &= \frac{(a_{\ell_{2}} - a_{\ell_{1}})}{(a_{\ell_{2}} - a_{\ell_{1}})^2 + (b_{\ell_{2}} - b_{\ell_{1}})^2 + (c_{\ell_{2}} - c_{\ell_{1}})^2}, \\
\bar{d}_{\ell_{1,2}} &= \frac{(b_{\ell_{2}} - b_{\ell_{1}})}{(a_{\ell_{2}} - a_{\ell_{1}})^2 + (b_{\ell_{2}} - b_{\ell_{1}})^2 + (c_{\ell_{2}} - c_{\ell_{1}})^2}, \\
\bar{h}_{\ell_{1,2}} &= \frac{(c_{\ell_{2}} - c_{\ell_{1}})}{(a_{\ell_{2}} - a_{\ell_{1}})^2 + (b_{\ell_{2}} - b_{\ell_{1}})^2 + (c_{\ell_{2}} - c_{\ell_{1}})^2}.
\end{align*}
$$

The 3D visualization of the attractive potential fields and the analogous contour plot (50 contours) generated using the target attractive function, equation (4). (a) 3D visualization. (b) Contour plot.
In a nutshell, all these components will now be incorporated to form a Lyapunov function, which will eventually lead to the design of the control inputs for the robotic system.

### 4. Design of the Control Inputs

In this section, we will first construct the Lyapunov function, followed by its time derivative from which we will ultimately extract the nonlinear control inputs for system (2).

#### 4.1. Lyapunov Function

The Lyapunov function, the total potentials that guarantee target convergence and obstacle and collision avoidance, is the sum of the attractive and repulsive potential fields. We begin first by introducing the control/tuning parameters:

(i) \( \varrho'_i > 0 \) and \( s = 1, \ldots, 6 \), for the avoidance of the 3th boundary of the workspace (see Section 3.2.1).

(ii) \( \varsigma'_i > 0 \) and \( \ell = 1, \ldots, m \), for the avoidance of the surface wall of the \( \ell \)th rectangular plane (see Section 3.2.2).

(iii) \( \beta_{ij} > 0 \) and \( i, j = 1, \ldots, n \), \( i \neq j \), for the avoidance of the \( j \)th point-mass mobile robot (see Section 3.2.3).

Suitably combining all the attractive and repulsive potential filed functions using these tuning parameters, we define a Lyapunov function for system (2) as

\[
L(x) = \sum_{i=1}^{n} V_i(x) + G_i(x) \left( \sum_{s=1}^{6} \varrho'_{is} WS_{is}(x) + \sum_{\ell=1}^{m} \varsigma'_{i\ell} RP_{i\ell}(x) + \sum_{j=i}^{n} \beta_{ij} MO_{ij}(x) \right),
\]

\( (15) \)
which is positive over the domain 
\( \mathcal{D}(\ell) = \{ x \in \mathbb{R}^m : W_{\ell \ell^t} (x) > 0, s = 1, \ldots, 6, \text{RP}_{\ell^t} (x) > 0, \ell = 1, \ldots, m \text{ and } \text{MO}_{ij} (x) > 0, j = 1, \ldots, n, i \neq j, i = 1, \ldots, n \} \).

\[ f_{i1} = \left( 1 + \sum_{s=1}^{m} \frac{p_{si}}{W_{s \ell}} + \sum_{s=1}^{m} \frac{\gamma_{s \ell}}{\text{MO}_{ij}} + \sum_{s=1}^{n} \frac{\beta_{ij}}{\text{MO}_{ij}} \right) \left( x_i - \tau_{i1} \right) - G_i - \frac{p_{s \ell}}{(W_{s \ell})^2} + G_i - \frac{p_{s \ell}}{(W_{s \ell})^2} \]

\[ -2G_i \sum_{j=1}^{n} \frac{\beta_{ij}}{\text{MO}_{ij}} (x_i - x_j) - G_i \sum_{s=1}^{m} \frac{\gamma_{s \ell}}{\text{RP}_{\ell^t}} \left( x_i - P \right) \left( 1 - (a_{s \ell} - a_{s \ell}) \alpha_{s \ell} - (a_{s \ell} - a_{s \ell}) \alpha_{s \ell} \right) \]

\[ f_{i2} = \left( 1 + \sum_{s=1}^{m} \frac{p_{si}}{W_{s \ell}} + \sum_{s=1}^{m} \frac{\gamma_{s \ell}}{\text{MO}_{ij}} + \sum_{s=1}^{n} \frac{\beta_{ij}}{\text{MO}_{ij}} \right) \left( y_i - \tau_{i2} \right) + G_i - \frac{p_{s \ell}}{(W_{s \ell})^2} - G_i - \frac{p_{s \ell}}{(W_{s \ell})^2} \]

\[ -2G_i \sum_{j=1}^{n} \frac{\beta_{ij}}{\text{MO}_{ij}} (y_i - y_j) - G_i \sum_{s=1}^{m} \frac{\gamma_{s \ell}}{\text{RP}_{\ell^t}} \left( y_i - P \right) \left( 1 - (b_{s \ell} - b_{s \ell}) \alpha_{s \ell} - (b_{s \ell} - b_{s \ell}) \alpha_{s \ell} \right) \]

\[ f_{i3} = \left( 1 + \sum_{s=1}^{m} \frac{p_{si}}{W_{s \ell}} + \sum_{s=1}^{m} \frac{\gamma_{s \ell}}{\text{MO}_{ij}} + \sum_{s=1}^{n} \frac{\beta_{ij}}{\text{MO}_{ij}} \right) \left( z_i - \tau_{i3} \right) - G_i - \frac{p_{s \ell}}{(W_{s \ell})^2} + G_i - \frac{p_{s \ell}}{(W_{s \ell})^2} \]

\[ -2G_i \sum_{j=1}^{n} \frac{\beta_{ij}}{\text{MO}_{ij}} (z_i - z_j) - G_i \sum_{s=1}^{m} \frac{\gamma_{s \ell}}{\text{RP}_{\ell^t}} \left( z_i - P \right) \left( 1 - (c_{s \ell} - c_{s \ell}) \alpha_{s \ell} - (c_{s \ell} - c_{s \ell}) \alpha_{s \ell} \right) \]

\[ (16) \]

for \( i = 1, \ldots, n \). In order to ensure stability in the sense of Lyapunov of system (2), we define the accompanying continuous velocity control laws as follows:

\[ v_i = -\frac{1}{a_{i1}} f_{i1} \]

\[ w_i = -\frac{1}{a_{i2}} f_{i2} \]

\[ u_i = -\frac{1}{a_{i3}} f_{i3} \]

4.2. Control Inputs. Next, we differentiate the various components of \( L(x) \) separately with respect to \( t \) to obtain (on suppressing \( x \)) the control inputs for system (2):

5. Stability Analysis

Using the notations \( x_c := (\tau_{i1}, \tau_{i2}, \tau_{i3}) \in \mathbb{R}^3 \) and \( x_c := (x_i, x_j, \ldots, x_n) \in \mathbb{R}^m \), we state the following theorem.

Theorem 1. A stable equilibrium point of system (2) is \( x_c \in \mathcal{D}(L(x)) \).
Proof. Since the Lyapunov function $L(x)$ of system (2) is defined, continuous and positive over the domain $\mathbb{D}(L) = \{ x \in \mathbb{R}^{3n}; \text{WS}_{ij}(x) > 0, i = 1, . . . , 6, \text{RP}_{ij}(x), \ell = 1, . . . , m \}$ and $\text{MO}_{ij}(x) > 0, j = 1, . . . , n, i \neq j, i = 1, . . . , n \}$, it can easily be verified that $L(x)$ satisfies the following properties:

1. $L(x)$ is continuous in the region $\mathbb{D}$ in the neighborhood of the point $x_c$ of system (2)

2. $L(x_c) = 0$

3. $L(x) > 0, \forall x \in \mathbb{D}(L(x)/x_c)$

Then, along a solution of system (2), we have

$$\dot{L}(x) = \sum_{i=1}^{n} \left[ f_{i1} \dot{x}_i + f_{i2} \dot{y}_i + f_{i3} \dot{z}_i \right] = 0.$$ 

(18)

Using (17), we have the following time derivative of $L(x)$ seminegative definite function for system (2):

$$\dot{L}(x) = -\sum_{i=1}^{n} \left[ \alpha_{i1} v_i^2 + \alpha_{i2} w_i^2 + \alpha_{i3} u_i^2 \right] \leq 0.$$ 

(19)

Therefore, $\dot{L}(x) \leq 0, \forall x \in \mathbb{D}(L(x))$ and $\dot{L}(x) = 0$. Moreover $L(x) \in C^1(\mathbb{D}(L(x)))$; hence, for system (2), $L(x)$ is classified as its Lyapunov function and $x$ is a stable equilibrium point.

Furthermore, with the design of the new controllers and the stability analysed for the robotic system, the effectiveness of the control scheme is verified using computer simulations. \(\square\)

6. Simulation Results

The three situations given in this section capture realistic situations to illustrate the adequacy, effectiveness, and robustness of the velocity-based controllers and the control scheme. In the following scenarios, the data use international units in the sense that parameters are unitless whereas the times can be treated consistently. For instance, the units of time can follow the international units like seconds or minutes and the distance can be in centimeters or meters.

6.1. Scenario 1. In this scenario, we consider a simple setup where $\mathcal{P}_i$ navigates itself from its initial position to its predefined target in the presence of a fixed rectangular-plane obstacle. There are 3 point-mass mobile robots and a rectangular-plane obstacle. Each of the point-mass mobile robots avoids each other as well as the rectangular-plane obstacle while en route to its target. It is very interesting to observe the proximity of the point-mass mobile robots to the wall as it tries to evade it, exerts just enough energy to move above the wall and converge to its targets. The behavior exhibited by $\mathcal{P}_i$ is quite intriguing as it mimics a similar behavior that a swarm of birds exhibits while the swarm approaches a wall. Figure 7(a) shows the default 3D view and Figure 7(b) shows the top 3D view while Figure 7(c) shows the front 3D view of the motion of the point-mass mobile robots. The obstacle has transparency to allow us to view the position and path of $\mathcal{P}_i$. The blue sphere represents the motion of $\mathcal{P}_i$ at $t = 0$ unit of time, red sphere at $t = 700$ units of time, green sphere at $t = 3500$ units of time, and purple sphere at $t = 15000$ units of time. Table 1 provides all the values of the initial conditions, constraints, and different parameters utilised in the simulation.

6.2. Scenario 2. Here we model rectangular towers, which could represent tall buildings in cities. These towers, constructed with 15 planes, block the path of a swarm of 5 point-mass mobile robots. The agents are observed to start from their initial positions and maneuver themselves to their predefined targets, while ensuring avoidance of the towers as well as interindividual collision avoidance. Each $\mathcal{P}_i$ computes the shortest and a collision-free path to its destination. Split maneuvers are observed while the robots are en route along their paths. Such an example with multiple towers can be used to model the obstacle avoidance capability of UAVs. Figure 8(a) shows the default 3D view and Figure 8(b) shows the top 3D view while Figure 8(c) shows the front 3D view of the motion of the point-mass mobile robots. The blue sphere represents the motion of $\mathcal{P}_i$ at $t = 0$ unit of time, red sphere at $t = 700$ units of time, green sphere at $t = 3120$ units of time, and purple sphere at $t = 8000$ units of time. Table 2 provides all the values of the initial conditions, constraints, and different parameters utilised in the simulation, if different from the previous scenario. For the construction of the towers, the reader is referred to the figures for the extraction of the coordinates.

6.3. Scenario 3. An interesting research domain involves tunnel passing maneuvers. In this scenario, there are 3 point-mass mobile robots and we design tunnels using rectangular planes. We use 8 rectangular planes to construct the tunnel. In addition, the top and one of the side views have been strategically made transparent to show the trajectory and the position of $\mathcal{P}_i$ as it maneuvers through the tunnel. The snapshots show how the point-mass robots strategize their motion to allow which robot will pass through the tunnel first and how they will converge to their respective predefined targets. Drones could be deployed in areas that are deemed to be "dull, dirty, and dangerous" as well as "difficult" such as that of collapsed tunnel passages to capture, store, check, and send data for analysis. Figure 9(a) shows the default 3D view and Figure 9(b) shows the top 3D view while Figure 9(c) shows the front 3D view of the motion of the point-mass mobile robots. The blue sphere represents the motion of $\mathcal{P}_i$ at $t = 0$ unit of time, red sphere at $t = 550$
units of time, green sphere at \( t = 2110 \) units of time, and purple sphere at \( t = 8000 \) units of time. Table 3 provides all the values of the initial conditions, constraints, and different parameters utilised in the simulation, if different from the previous scenario. The evolution of the Lyapunov function and its time derivative together with its control inputs along the trajectories are depicted in Figure 9, respectively.
Figure 8: Scenario 2. The different viewpoints of $\mathcal{P}_i$ while it is in motion to its target with towers constructed from the rectangular planes. (a) Default 3D motion view. (b) Top 3D motion view. (c) Front 3D motion view.

Table 2: Scenario 2. The parameters utilised in the numerical simulation, with $n = 3$ and $m = 15$.

| Description                                      | Value                                                                 |
|-------------------------------------------------|----------------------------------------------------------------------|
| Initial state of the point-mass mobile robots   |                                                                      |
| Initial position, radius                        | $(x_1, y_1, z_1) = (10, 50, 40), r_p_1 = 2$                           |
|                                                  | $(x_2, y_2, z_2) = (10, 70, 40), r_p_2 = 2$                           |
| Constraints                                     |                                                                      |
| Target centre, radius                           | $(\tau_{11}, \tau_{12}, \tau_{13}) = (190, 70, 40), r_r_1 = 1$      |
|                                                  | $(\tau_{21}, \tau_{22}, \tau_{23}) = (190, 40, 40), r_r_2 = 1$      |
|                                                  | $(\tau_{31}, \tau_{32}, \tau_{33}) = (190, 50, 40), r_r_3 = 1$      |
| Avoidance of workspace                          | $\varphi_i = 20$ for $i = 1, \ldots, 5$ and $s_i = 1, \ldots, 6$   |
| Avoidance of rectangular plane                  | $s_{it} = 0.5$, for $i = 1, \ldots, 5$ and $t = 1, \ldots, 15$      |
| Interindividual collision avoidance             | $\beta_{ij} = 5$ for $i = j = 1, \ldots, 5$ and $j \neq i$          |

Table 3: Scenario 3. The parameters utilised in the numerical simulation, with $n = 3$ and $m = 8$.

| Description                                      | Value                                                                 |
|-------------------------------------------------|----------------------------------------------------------------------|
| Initial state of the point-mass mobile robots   |                                                                      |
| Initial position, radius                        | $(x_1, y_1, z_1) = (10, 70, 65), r_p_1 = 2$                           |
|                                                  | $(x_2, y_2, z_2) = (10, 10, 10), r_p_2 = 2$                           |
|                                                  | $(x_3, y_3, z_3) = (10, 50, 40), r_p_3 = 2$                           |
| Constraints                                     |                                                                      |
| Target centre, radius                           | $(\tau_{11}, \tau_{12}, \tau_{13}) = (150, 30, 20), r_r_1 = 1$      |
|                                                  | $(\tau_{21}, \tau_{22}, \tau_{23}) = (160, 80, 60), r_r_2 = 1$      |
|                                                  | $(\tau_{31}, \tau_{32}, \tau_{33}) = (180, 50, 40), r_r_3 = 1$      |
| Avoidance of workspace                          | $\varphi_i = 10$ for $i = 1, 2, 3$ and $s_i = 1, \ldots, 6$         |
| Avoidance of rectangular plane                  | $s_{it} = 5$ for $i = 1, 2, 3$ and $t = 1, \ldots, 8$               |
Figure 9: Scenario 3. The evolutionary behavior of the Lyapunov function, its time derivative, and its control inputs along the trajectories of Figure 10. (a) Lyapunov function. (b) Lyapunov function time derivative. (c) Controllers.

Figure 10: Scenario 3. The different viewpoints of $P$, while it is in motion to its target with tunnel passing maneuvers. (a) Default 3D motion view. (b) Top 3D motion view. (c) Front 3D motion view.
Interestingly, the behavior exhibited in this scenario can be seen in nature, namely, the leader-follower strategy, where the leader guides the followers to food sources, safety, and so on. We note that the leader-follower strategy, cooperative hunting, and avoidance in the military are drone-based applications considered common nowadays.

7. Conclusions and Future Work

Mathematical modelling and the design of motion planners for robotic systems are a complex, computationally expensive yet a fascinating research area. In this paper, the LbCS was applied to derive a set of robust, unique continuous time-invariant velocity-based control inputs that effectively handle the problem of MPC of point-mass mobile robots in a dynamic environment that, for the absolute first time, incorporates rectangular-plane obstacles. The convergence of the mobile robots to a neighborhood of a predefined target is ensured by the Lyapunov direct method. The effectiveness and robustness of the control scheme were illustrated via computer simulation of virtual scenarios that depicts real-life situations.

To the authors’ knowledge, this is the first time in literature whereby the MDT was used to derive the mathematical functions for the successful avoidance of rectangular-plane-shaped obstacles. The introduction of the rectangular-plane obstacle into the MPC problem has created new dimensions and potentials for research. The advantages of the MDT are numerous such as making it possible for plane-shaped (and other irregular) obstacles to be treated within the motion planners, help in simplifying collision-avoidance algorithms, and permit maximum free-space for the robots traversing the workspace.

This work paves the way for numerous future directions. Our principal objective is to extend the rectangular-plane obstacles in a workspace for the MPC of a flock of quadrotors performing hovering maneuvers and undergoing split-and-rejoin maneuvers when encountered with towers and tunnels.

Data Availability

The research data for this article are purely simulation-based. These would be available upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] K. Molloy, D. Laurent, M. Vaisset, T. Siméon, and Juan Cortés, “Simultaneous system design and path planning: a sampling-based algorithm,” The International Journal of Robotics Research, vol. 38, no. 2-3, pp. 375–387, 2019.

[2] A. Gasparetto, P. Boscariol, A. Lanzutti, and R. Vidoni, Path Planning and Trajectory Planning Algorithms: A General Overview, Springer International Publishing, Berlin, Germany, 2015.

[3] N. Lashkari, M. Biglarbegian, S. X. Yang, and Yang, “Development of novel motion planning and controls for a series of physically connected robots in dynamic environments,” Journal of Intelligent & Robotic Systems, vol. 95, no. 2, pp. 291–310, 2019.

[4] B. N. Sharma, R. Raj, and J. Vanualailai, “Navigation of carlike robots in an extended dynamic environment with swarm avoidance,” International Journal of Robust and Nonlinear Control, vol. 28, no. 2, pp. 678–698, 2018.

[5] M. Liu, H. Ma, J. Li, and S. Koenig, “Task and path planning for multi-agent pickup and delivery,” in Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS ’19, Richland, SC, USA, May 2019.

[6] D. Ali and V. Sezer, “Motion planning and control with randomized payloads using deep reinforcement learning,” in Proceedings of the 2019 Third IEEE International Conference on Robotic Computing (IRC), Naples, Italy, February 2019.

[7] J. Raj, K. Raghukwaiya, and J. Vanualailai, “Novel liapunov-based autonomous controllers for quadrotors,” IEEE Access, vol. 8, no. 1, pp. 47393–47406, 2020.

[8] F. Ho, A. Salta, R. Geraldes, A. Goncalves, M. Cavazza, and H. Prendinger, “Multi-agent path finding for uav traffic management,” in Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS ’19, Richland, SC, USA, May 2019.

[9] J. T. Ebert, M. Gauci, and R. Nagpal, “Multi-feature collective decision making in robot swarms,” in Proceedings Of The 17th International Conference On Autonomous Agents and MultiAgent Systems, AAMAS ’18, Richland, SC, USA, May 2018.

[10] B. Sharma, J. Vanualailai, and S. Singh, “Motion planning and posture control of multiple n-link doubly nonholonomic manipulators,” Robotics, vol. 35, no. 1, pp. 1–25, 2017.

[11] D. A. L. Burbano, P. Delellis, and M. dibernardo, “Self-tuning proportional integral control for consensus in heterogeneous multi-agent systems,” European Journal of Applied Mathematics, vol. 27, no. 6, pp. 923–940, 2016.

[12] J. Raj, R. Krishna, S. Singh, B. Sharma, and J. Vanualailai, N. Che, “Swarming intelligence of 1-trailer systems,” Hamzah Asyrrani Sulaiman, Mohd Azlishah Othman, Mohd Fairuz Iskandar Othman, Yahaya Abdul Rahim, and, pp. 251–264, Springer International Publishing, Berlin, Germany, 2016.

[13] R. Krishna, B. Sharma, and J. Vanualailai, “Leader-follower based locally rigid formation control,” Journal of Advanced Transportation, vol. 2018, Article ID 5278565, 14 pages, 2018.

[14] J. Raj, R. Krishna, J. Vanualailai, and B. Sharma, “Navigation of car-like robots in three-dimensional space,” in Proceedings of the 2018 5th Asia-Pacific World Congress on Computer Science and Engineering (APWc on CSE), Nadi, Fiji, December 2018.

[15] T. S. A. Ebitohal, “A complete multi-robot path-planning algorithm: jaamas track,” in Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS ’19, Richland, SC, USA, May 2019.

[16] B. Sharma, J. Vanualailai, and A. Prasad, “A dello-strategy: facilitating dual-formation control of a virtually connected team dello-strategy: facilitating dual-formation control of a virtually connected team,” Journal of Advanced Transportation, vol. 2017, Article ID 9213805, 17 pages, 2017.

[17] R. Krishna and S. Singh, “Formation types of multiple steerable 1-trailer mobile robots via split/rejoin maneuvers,” New Zealand Journal of Mathematics, vol. 43, no. 7–21, 2013.

[18] J. Vanualailai, “Stable emergent formations for a swarm of autonomous car-like vehicles,” International Journal of Advanced Robotic Systems, vol. 16, no. 5, pp. 1–17, 2019.
[19] D. Albani, T. Manoni, D. Nardi, and V. Trianni, “Dynamic uav swarm deployment for non-uniform coverage,” in Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS ’18, Richland, SC, USA, May 2018.

[20] E. Hartuv, N. Agmon, and S. Kraus, “Scheduling spare drones for persistent task performance under energy constraints,” in Proceedings Of the 17th International Conference On Autonomous Agents and MultiAgent Systems, AAMAS ’18, Richland, SC, USA, May 2018.

[21] B. Sharma, J. Vanualailai, and S. Singh, “Tunnel passing maneuvers of prescribed formations,” International Journal of Robust and Nonlinear Control, vol. 24, no. 5, pp. 876–901, 2014.

[22] M. d. S. Arantes, C. F. M. Toledo, B. C. Williams, and M. Ono, “Collision-free encoding for chance-constrained nonconvex path planning,” IEEE Transactions on Robotics, vol. 35, no. 2, pp. 433–448, 2019.

[23] K. Ariu, F. Cheng, M. da Silva Arantes, C. Toledo, and B. C. Williams, “Chance-constrained path planning with continuous time safety guarantees,” in Proceedings of the Workshops at the Thirty-First AAAI Conference on Artificial Intelligence, Palo Alto, CA, USA, February 2017.