Robust Formation Control and Trajectory Tracking of Multiple Quadrotors Using a Discrete-Time Sliding Mode Control Technique

Aydin Can * **Imil Hamda Imran** * **Joshua Price** **
Allahyar Montazeri *

* *Engineering Department, Lancaster University, Lancaster, LA1 4YW, UK*
** *National Nuclear Laboratory, 5th Floor, Chadwick House, Warrington Road, Birchwood Park, Warrington, WA3 6AE, UK*

Abstract: In this paper, a centralised robust discrete-time sliding mode controller is proposed for the formation control of a multi-quadrotor system in the presence of disturbances. The control system, based on consensus control, is designed around the full, nonlinear, under actuated dynamics of the quadrotor. Graph theory is used to define the communication topology of the multi-agent system. Results obtained through simulation in MATLAB and Simulink are used to verify the control system.

Keywords: Sliding mode control, Quadrotor control, Nonlinear discrete-time control, Hector SLAM, Trajectory tracking.

1. INTRODUCTION

In recent decades, rapid technological improvements in several areas have forced industries to adapt to new challenges, and created a several new opportunities. One such area adapting to these new changes is the Nuclear Industry, which aims to provide a clean energy solution to keep up with increasing global energy demands (Lu et al., 2020). Industry 4.0, or the fourth industrial revolution, aims to combine emerging technologies, such as the Internet of Things (IoT), with Artificial Intelligence and Cyber-Physical systems to combat these new challenges (Blanchet and Confais, 2016).

More recently, Unmanned Aerial Vehicles (UAVs), and more specifically quadrotors, have been proposed as a platform for these Cyber-Physical systems. An autonomous network of UAVs, equipped with various sensors and actuators, alongside software installed on on-board computational devices, provides a capable platform for the gathering and analysis of data (Wang et al., 2020).

The control of quadrotor platforms is a widely researched area, with early control systems being developed using linear methods, such as PID (Pratama et al., 2018; Khan and Kadri, 2015) and LQR (Martins et al., 2019). More recently, robust techniques such as sliding mode control (Pan et al., 2018; Can et al., 2020; Sarkar et al., 2017), adaptive control (Imran et al., 2021), and backstepping control (He et al., 2016) have also been developed to control the quadrotors with introduced parametric uncertainties, or in harsh environments where there may be added noise and external disturbances.

As these quadrotor systems rely on sensors, actuators, and computation for their control, they are inherently affected by the discrete sampling times of these devices. Sensors used in the implementation of Simultaneous Localisation and Mapping (SLAM), such as RGBD cameras and LiDAR scanners, usually operate at low sampling rates which can vary greatly depending on the device used (Liu et al., 2018; Slamtec, n.d.; Intel, 2020). For this reason, discrete-time control methods are becoming increasingly popular. By implementing these control systems in discrete time, it is possible to design controllers that are capable of robust control at these lower sampling rates. One paper develops a discrete-time sliding mode controller to control the position and attitude of a quadrotor to a desired reference point (Xiong and Zhang, 2016). By combining discrete-time sliding mode control with a disturbance observer, another author was able to develop a robust controller capable of controlling a quadrotor in the presence of parametric uncertainties, as well as external disturbances (Han et al., 2019). Another method of adjusting controller for sampling time is through the introduction of event-triggering, in which a control system is implemented with a variable sampling time. These event-triggered controllers often display good controller performance, while reducing the overall control effort of the system (Sarkar et al., 2017; Tian et al., 2021; Nokhodberiz et al., 2019).

⋆ This work has been supported by the Centre for Innovative Nuclear Decommissioning (CINDe), which is led by the National Nuclear Laboratory, in partnership with Sellafield Ltd. and a network of Universities that includes the University of Manchester, Lancaster University, the University of Liverpool and the University of Cumbria. The authors would also like to acknowledge the Engineering and Physical Sciences Research Council (EPSRC), grant number EP/R02572X/1, and National Centre for Nuclear Robotics (NCNR).

A Cyber-Physical System based on a multi-quadrotor platform must be capable of autonomous movement around unknown areas. Modern research has provides an insight into various methods to control these multi-quadrotor systems. Multi-agent control problems include the synchronization problem, where a number of agents meet at a common location (Wang et al., 2013), the formation problem, where a multi-agent network must maintain a desired formation (Liu and Jiang, 2013), and the flocking problem, where multiple agents must mimic the flocking behaviour displayed in nature in birds (Wen et al., 2012).

In this paper, we are addressing the formation problem for a multi-agent system of networked quadrotors. A number of methods have been proposed to solve the formation problem in quadrotors. In one paper, the authors propose a linear PID controller, alongside a leader-follower network structure, to solve the formation problem (Wu et al., 2017). While PID can be robust, quadrotors are complex systems comprised of nonlinear and under-actuated dynamics that can be exposed to parametric uncertainties and disturbances, in which case a more robust control algorithm may be necessary. Another paper proposes a robust control approach to solve the formation problem based on sliding-mode control (Abdoli et al., 2018). In this paper, a group of quadrotors were capable of maintaining formation while a load is suspended from the quadrotors in simulation. This was compared to linear methods such a linear control system using LQR-PID control and was shown to be superior. In another paper, adaptive control is proposed as a solution to the formation problem in the presence of parametric uncertainties (Wang and Yu, 2017). The paper is successful in developing an algorithm that estimates the parameters of each quadrotor while controlling the formation of quadrotors.

In this paper we aim to extend current research by developing a discrete-time sliding mode control system for the formation control of a number of quadrotor agents in a network. By designing the controller in the discrete domain we aim to reduce the affects introduced through sampling at lower sampling rates.

The structure of the paper is as follows: In section 2, we formulate the problem; section 3 discusses the controller design; section 4 displays results obtained from simulation; finally, section 5 concludes the paper and suggests avenues for futures research.

2. PROBLEM FORMULATION

2.1 Discrete-time Quadrotor Model

A discrete time quadrotor model can be formulated through the discretisation of a continuous time model using the forward Euler method:

$$\dot{x}_k = \frac{x_{k+1} - x_k}{T}. \tag{1}$$

This gives the following sets of equations representing the quadrotor dynamics:

$$\begin{cases} x_{k+1} = x_k + T\dot{x}_k, \\ \dot{x}_{k+1} = \dot{x}_k + T(c\varphi_k c\psi_k s\theta_k + s\varphi_k s\psi_k)\dfrac{u_{1,k}}{m}, \\ y_{k+1} = y_k + T\dot{y}_k \\ \dot{y}_{k+1} = \dot{y}_k + T(c\varphi_k s\theta_k s\psi_k - c\psi_k s\varphi_k)\dfrac{u_{1,k}}{m}, \end{cases} \tag{2}$$

$$\begin{cases} z_{k+1} = z_k + T\dot{z}_k, \\ \dot{z}_{k+1} = \dot{z}_k + T\big(g - (c\varphi_k c\theta_k \dfrac{u_{1,k}}{m})\big), \\ \varphi_{k+1} = \varphi_k + T\dot{\varphi}_k, \\ \dot{\varphi}_{k+1} = \dot{\varphi}_k + T(\dfrac{(I_{yy} - I_{zz})\dot{\theta}_k\dot{\psi}_k + u_{2,k}}{I_{xx}}), \\ \varphi_{k+1} = \varphi_k + T\dot{\varphi}_k, \\ \dot{\theta}_{k+1} = \dot{\theta}_k + T(\dfrac{(I_{yy} - I_{zz})\dot{\varphi}_k\dot{\psi}_k + u_{3,k}}{I_{xx}}), \\ \psi_{k+1} = \psi_k + T\dot{\psi}_k, \\ \dot{\psi}_{k+1} = \dot{\psi}_k + T(\dfrac{(I_{yy} - I_{zz})\dot{\varphi}_k\dot{\theta}_k + u_{4,k}}{I_{xx}}). \end{cases} \tag{3}$$

Equation (2) shows the outer-loop position subsystem of the quadrotor, while equation (3) is the inner-loop attitude and altitude subsystem. Here, x_k, y_k, and z_k represent the positions of the quadrotor at time step k. Terms φ_k, θ_k, and ψ_k denote the Euler angles roll, pitch, and yaw of the quadrotor at time step k. The subscript $k+1$ denotes one time step in the future and the velocities of each value are represented using an overdot, e.g. $\dot{x}_k$ is the velocity of x at time step k. Term T is the time step of the discrete system. Term g is acceleration acting on the quad due to gravity. The mass of the quadrotor is represented by term m, while I_{xx}, I_{yy}, and I_{zz} represent the quadrotor moments of inertia. Finally, $u_{1,k}$ is the total thrust acting on the system, and $u_{2,k}$, $u_{3,k}$, and $u_{4,k}$ are the torques acting on roll, pitch and yaw axis respectively. It should be noted that these four terms denote the control inputs for the quadrotor system.

Remark 1. Here, cos and sin are replaced with c and s in each case that they are used. For example, $c\varphi_k$ represents $cos(\varphi_k)$ and $s\varphi_k$ represents $sin(\varphi_k)$.

2.2 Multi-Agent Problem

To describe the communication topology between each agent, graph theory is used.

Definition 1. A directed graph, $G = \{\mathbf{V}, \mathbf{E}, \mathbf{A}\}$, contains a set of nodes $\mathbf{V} \in \mathbb{R}^n$, where each node represents an agent in the network where n is the total number of agents, and a set of directed edges $\mathbf{E} \in \mathbf{V} \times \mathbf{V}$, representing a communication link between two agents in the network. The matrix $\mathbf{A} = [a^{ij}] \in \mathbb{R}^{n \times n}$ is the adjacency matrix of graph G such that

$$\begin{cases} a^{ij} = 0 & if \ \ i = j \\ a^{ij} = 1 & V^i \ has\ a\ directed\ connection\ to\ V^j \\ a^{ij} = 0 & otherwise \end{cases}$$

Furthermore, as the communication topology is time-varying, we assume graph $G \in \hat{\boldsymbol{G}}$ where $\hat{\boldsymbol{G}} = [G^1, G^2, ...]$ is a set of all possible graphs containing a spanning tree. At any time-step k, graph G can be on of any of the possible graphs in the set $\hat{\boldsymbol{G}}$. The Laplacian matrix of graph G is given by $L^G \in \mathbb{R}^{n\times n}$.

For the purpose of formation control, we take the dynamics of the outer-loop position subsystem only. The set of equations in (2) can be converted into state space to describe the position dynamics of each agent i in the network.

$$\begin{cases}\boldsymbol{\eta}^i_{1,k+1} = \boldsymbol{\eta}^i_{1,k} + T\boldsymbol{\eta}^i_{2,k},\\ \boldsymbol{\eta}^i_{2,k+1} = \boldsymbol{\eta}^i_{2,k} + T(\mathbf{f}^i(\boldsymbol{\eta}^i_k) + \mathbf{g}^i(\boldsymbol{\eta}^i_k)\mathbf{u}^i_k + \mathbf{d}^i_k).\end{cases} \tag{4}$$

$$\boldsymbol{\eta}^i_k = \begin{bmatrix}\boldsymbol{\eta}^i_{1,k}\\ \boldsymbol{\eta}^i_{2,k}\end{bmatrix}, \tag{5}$$

where $\boldsymbol{\eta}^i_{1,k} = [x^i_k \;\; y^i_k]$, $\boldsymbol{\eta}^i_{1,k} \in \mathbb{R}^2$ represents the state vector of the quadrotor position states for each agent i. The vector $\boldsymbol{\eta}^i_{2,k} = [\dot{x}^i_k \;\; \dot{y}^i_k]^\top$, $\boldsymbol{\eta}^i_{2,k} \in \mathbb{R}^2$ is a vector of velocities of the position states for agent i. The states, and their derivatives are concatenated into vector $\boldsymbol{\eta}^i_k \in \mathbb{R}^4$. The vector $\mathbf{u}^i_k = [u^i_{1,k} \;\; u^i_{2,k}, \;\; u^i_{3,k}, u^i_{4,k}]^\top, \mathbf{u}^i_k \in \mathbb{R}^4$. Terms $\mathbf{f}^i : \mathbb{R}^4 \rightarrow \mathbb{R}^2$ and $\mathbf{g}^i : \mathbb{R}^4 \rightarrow \mathbb{R}^{2\times 4}$ are nonlinear functions that describe the system dynamics and $\mathbf{d}^i_k \in \mathbb{R}^2$ represents an external matched disturbance acting on the positional subsystem of agent i. It is assumed that $\mathbf{d}^i_k$ is bounded such that $\|\mathbf{d}^i_k\|_\infty \leq d^i_{max}, \forall k \in \mathbb{N}$.

The compact form for the dynamics of the graph are therefore:

$$\begin{cases}\boldsymbol{\eta}_{1,k+1} = \boldsymbol{\eta}_{1,k} + T\boldsymbol{\eta}_{2,k},\\ \boldsymbol{\eta}_{2,k+1} = \boldsymbol{\eta}_{2,k} + T(\mathbf{f}(\boldsymbol{\eta}_k) + \mathbf{g}(\boldsymbol{\eta}_k)\mathbf{u}_k + \mathbf{d}_k),\end{cases} \tag{6}$$

$$\boldsymbol{\eta}_k = \begin{bmatrix}\boldsymbol{\eta}_{1,k}\\ \boldsymbol{\eta}_{2,k}\end{bmatrix}, \tag{7}$$

where the vector $\boldsymbol{\eta}_k \in \mathbb{R}^{4n}$ contains the position and velocity components. The vector $\boldsymbol{\eta}_{1,k} = [\boldsymbol{\eta}^{1^\top}_{1,k}, \boldsymbol{\eta}^{2^\top}_{1,k}, ..., \boldsymbol{\eta}^{n^\top}_{1,k}]^\top$, $\boldsymbol{\eta}_{1,k} \in \mathbb{R}^{2n}$. Vector $\boldsymbol{\eta}_{2,k} = [\boldsymbol{\eta}^{1^\top}_{2,k}, \boldsymbol{\eta}^{2^\top}_{2,k}, ..., \boldsymbol{\eta}^{n^\top}_{2,k}]^\top, \boldsymbol{\eta}_{2,k} \in \mathbb{R}^{2n}$. Input vector $\mathbf{u}_k = [\mathbf{u}^{1^\top}_k, \mathbf{u}^{2^\top}_k, ..., \mathbf{u}^{n^\top}_k]^\top, \mathbf{u}_k \in \mathbb{R}^{4n}$. The vector $\mathbf{f}(\boldsymbol{\eta}_k) = [\mathbf{f}^1(\boldsymbol{\eta}^1_k)^\top, \mathbf{f}^2(\boldsymbol{\eta}^2_k)^\top, ..., \mathbf{f}^n(\boldsymbol{\eta}^n_k)^\top]^\top$, $\mathbf{f}(\boldsymbol{\eta}_k) \in \mathbb{R}^{2n}$ and the matrix $\mathbf{g}(\boldsymbol{\eta}_k) = diag[\mathbf{g}^1(\boldsymbol{\eta}^1_k)^\top, \mathbf{g}^2(\boldsymbol{\eta}^2_k)^\top, ..., \mathbf{g}^n(\boldsymbol{\eta}^n_k)^\top]^\top, \mathbf{g}(\boldsymbol{\eta}_k) \in \mathbb{R}^{4n\times 2n}$. Finally the vector $\mathbf{d}_k = [\mathbf{d}^{1^\top}_k, \mathbf{d}^{2^\top}_k, ..., \mathbf{d}^{n^\top}_k]^\top, \mathbf{d}_k \in \mathbb{R}^{2n}$.

The dynamics of the virtual leader can be taken as the desired trajectory of center of the formation of quadrotors, and thus can be represented as

$$\begin{cases}\boldsymbol{\eta}^0_{1,k+1} = \boldsymbol{\eta}^0_{1,k} + T\boldsymbol{\eta}^0_{2,k},\\ \boldsymbol{\eta}^0_{2,k+1} = -c_1\boldsymbol{\eta}^0_{1,k} - c_2T\boldsymbol{\eta}^0_{2,k} + c_1\boldsymbol{\eta}^{d0}_{1,k} + c_2T\boldsymbol{\eta}^{d0}_{2,k},\end{cases} \tag{8}$$

where agent 0 is considered the virtual leader of the system, with the same states as the other agents in the system. Scalar values $c_1 \in \mathbb{R}$ and $c_2 \in \mathbb{R}$ are gains to be designed. $\boldsymbol{\eta}^{d0}_{1,k}$ is the desired values of positions $\boldsymbol{\eta}^0_{1,k}$, and $\boldsymbol{\eta}^{d0}_{2,k}$ is the desired value of velocities $\boldsymbol{\eta}^0_{2,k}$.

Furthermore, for the purpose of formation control, the term $\Delta\boldsymbol{\eta}^i_{1,k} = [\Delta x^i_k \;\; \Delta y^i_k]^\top$, $\Delta\boldsymbol{\eta}^i_{1,k} \in \mathbb{R}^2$ is introduced as the desired distance of agent i from the leader x^0. Term $\Delta\boldsymbol{\eta}^i_{2,k} = [\Delta\dot{x}^i_k \;\; \Delta\dot{y}^i_k]^\top$, $\Delta\boldsymbol{\eta}^i_{2,k} \in \mathbb{R}^2$ is the rate of change of the positions in the formation. The formation is considered time-varying if any value of $\Delta\boldsymbol{\eta}^i_{2,k}$ is non-zero.

Definition 2. Assuming that the virtual leader 0 is the center of the formation the quadrotor swarm has achieved formation if the following limit is satisfied:

$$\begin{cases}\lim\limits_{k\to\infty}(\mathbf{e}^i_{1,k}) = 0, \;\; \forall i,\\ \lim\limits_{k\to\infty}(\mathbf{e}^i_{2,k}) = 0, \;\; \forall i,\end{cases} \tag{9}$$

where

$$\mathbf{e}^i_{1,k} = \sum_{j\in n_j} a^{ij}(\boldsymbol{\eta}^j_{1,k} - \Delta\boldsymbol{\eta}^j_{1,k} - \boldsymbol{\eta}^i_{1,k} + \Delta\boldsymbol{\eta}^i_{1,k}) + b^i(\boldsymbol{\eta}^0_{1,k} - \boldsymbol{\eta}^i_{1,k} + \Delta\boldsymbol{\eta}^i_{1,k}), \tag{10}$$

$$\mathbf{e}^i_{2,k} = \sum_{j\in n_j} a^{ij}(\boldsymbol{\eta}^j_{2,k} - \Delta\boldsymbol{\eta}^j_{2,k} - \boldsymbol{\eta}^i_{2,k} + \Delta\boldsymbol{\eta}^i_{2,k}) + b^i(\boldsymbol{\eta}^0_{2,k} - \boldsymbol{\eta}^i_{2,k} + \Delta\boldsymbol{\eta}^i_{2,k}). \tag{11}$$

Here, $\mathbf{e}^i_{1,k} \in \mathbb{R}^2$ contains the x and y components of the position error dynamics, and $\mathbf{e}^i_{2,k} \in \mathbb{R}^2$ contains the x and y components of the velocity error dynamics. The term b^i is the scalar components of the diagonal matrix $B \in \mathbb{R}^{n\times n}$ for each agent, where $b^i = 1$ if agent i is directly connected to the virtual leader 0, otherwise $b^i = 0$.

From equations (10) and (11), the compact form of the error dynamics can be derived for each position channels of the quadrotor agents.

$$\mathbf{e}_{1,k} = -\Big((L+B)\otimes\mathbf{I}_n\Big)(\boldsymbol{\eta}_{1,k} - \Delta\boldsymbol{\eta}_{1,k} - \underline{1}\otimes\boldsymbol{\eta}^0_{1,k}), \tag{12}$$

$$\mathbf{e}_{2,k} = -\Big((L+B)\otimes\mathbf{I}_n\Big)(\boldsymbol{\eta}_{2,k} - \Delta\boldsymbol{\eta}_{2,k} - \underline{1}\otimes\boldsymbol{\eta}^0_{2,k}), \tag{13}$$

where $\mathbf{e}_{1,k} \in \mathbb{R}^{2n}$ contains the error dynamics for each of the x and y positions for each quadrotor agent. $\mathbf{e}_{2,k} \in \mathbb{R}^{2n}$ contains the error for each velocity in the x and y directions for each agent. The Kronecker product is denoted by $\otimes$ and $\mathbf{I}_N \in \mathbb{R}^{2\times 2}$ is the identity matrix.

Remark 2. For clarity, the position and velocity error dynamics in equations (12) and (13) can be separated into their individual components by removing the Kronecker product. For example, the compact form of the error dynamics for the x subsystem given by $\mathbf{e}_{1,k,x} \in \mathbb{R}^n$ and $\mathbf{e}_{2,k,x} \in \mathbb{R}^n$ can be represented by

$$\mathbf{e}_{1,k,x} = -(L+B)(\mathbf{x}_{1,k} - \Delta\mathbf{x}_{1,k} - \underline{1}x^0_k), \tag{14}$$

$$\mathbf{e}_{2,k,x} = -(L+B)(\mathbf{x}_{2,k} - \Delta\mathbf{x}_{2,k} - \underline{1}\dot{x}^0_k), \tag{15}$$

where $\mathbf{x}_{1,k} \in \mathbb{R}^n$ contains the x^i_k position of each agent i in the network. $\mathbf{x}_{2,k} \in \mathbb{R}^n$ contains the velocity $\dot{x}^i_k$ of each agent i in the network. $\Delta\mathbf{x}_{1,k} \in \mathbb{R}^n$ and $\Delta\mathbf{x}_{2,k} \in \mathbb{R}^n$ contain the desired separation Δx^i_k for each agent and rate of change of that separation $\Delta\dot{x}^i_k$ respectively.

3. CONTROLLER DESIGN

3.1 Discrete-Time Sliding Mode Formation Control

First, we take the nominal sliding surface of the networked system as

$$\boldsymbol{\sigma}_k = \boldsymbol{\alpha}(\mathbf{e}_{1,k}) + \mathbf{e}_{2,k}, \tag{16}$$

where $\boldsymbol{\sigma}_k = [\sigma_k^1, \sigma_k^2, ..., \sigma_k^n]^\top, \boldsymbol{\sigma}_k \in \mathbb{R}^n$ is the vector of sliding surfaces for each agent i. $\boldsymbol{\alpha} \in \mathbb{R}^{n \times n}$ is a diagonal matrix of gain parameters. The nominal sliding surface of the system at time step $k+1$ is given by

$$\boldsymbol{\sigma}_{k+1} = \boldsymbol{\alpha}(\mathbf{e}_{1,k+1}) + \mathbf{e}_{2,k+1}, \tag{17}$$

where $\mathbf{e}_{1,k+1}$ and $\mathbf{e}_{2,k+1}$ are error dynamics at times step $k+1$. A reaching law first proposed by Gao et al. (1995) is then implemented to force the system to slide along the sliding surface (16) and is given by

$$\boldsymbol{\sigma}_{k+1} - \boldsymbol{\sigma}_k = -\boldsymbol{\mu}\boldsymbol{\sigma}_k T - \boldsymbol{\epsilon} T \boldsymbol{sgn}(\boldsymbol{\sigma}_k) \tag{18}$$

Here, $\boldsymbol{\mu} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{\epsilon} \in \mathbb{R}^{n \times n}$ are diagonal matrices of tunable parameters. $\boldsymbol{sgn}$ is the signum vector function $\mathbb{R}^n \mapsto \mathbb{R}^n$ where
$(\sigma_k^1, \sigma_k^2, ..., \sigma_k^n) \mapsto (sgn(\sigma_k^1), sgn(\sigma_k^2), ..., sgn(\sigma_k^n))$.

3.2 *Outer-loop formation controller*

As the dynamic system of each quadrotor is underactuated, in order to design the controller for the x and y subsystems, virtual control inputs must be used. These are given for each agent i in the network by

$$u_{x,k}^i = (c\varphi_k^i c\psi_k^i s\theta_k^i + s\varphi_k^i s\psi_k^i), \tag{19}$$

$$u_{y,k}^i = (c\varphi_k^i s\theta_k^i s\psi_k^i - c\psi_k^i s\varphi_k^i). \tag{20}$$

Each agents control inputs can be concatenated into vectors, such that $\mathbf{u}_{x,k} = [{u_{x,k}^1}^\top, {u_{x,k}^2}^\top, ..., {u_{x,k}^n}^\top]^\top \in \mathbb{R}^n$ and $\mathbf{u}_{y,k} = [{u_{y,k}^1}^\top, {u_{y,k}^2}^\top, ..., {u_{y,k}^n}^\top]^\top \in \mathbb{R}^n$. By substituting equations (2), (14) and (15) into the reaching law in equation (18) and rearranging for the virtual control inputs $\mathbf{u}_{x,k}$ and $\mathbf{u}_{y,k}$ we obtain the two following control equations:

$$\begin{aligned}\mathbf{u}_{x,k} = (T\mathbf{u}_{1,k}\mathbf{m})^{-1}\Bigg(&(\mathbf{x}_{2,k} - \Delta\mathbf{x}_{2,k+1} - \underline{\mathbf{1}}\dot{x}_{k+1}^0)\\ &-(L+B)^{-1}\Big(\boldsymbol{\mu}_x\boldsymbol{\sigma}_{x,k}T + \boldsymbol{\epsilon}_x\boldsymbol{sgn}(\boldsymbol{\sigma}_{x,k})T\\ &+\boldsymbol{\alpha}(e_{1,k+1,x}) - \boldsymbol{\sigma}_{k,x}\Big)\Bigg),\end{aligned} \tag{21}$$

$$\begin{aligned}\mathbf{u}_{y,k} = (T\mathbf{u}_{1,k}\mathbf{m})^{-1}\Bigg(&(\mathbf{y}_{2,k} - \Delta\mathbf{y}_{2,k+1} - \underline{\mathbf{1}}\dot{y}_{k+1}^0)\\ &-(L+B)^{-1}\Big(\boldsymbol{\mu}_y\boldsymbol{\sigma}_{y,k}T + \boldsymbol{\epsilon}_y\boldsymbol{sgn}(\boldsymbol{\sigma}_{y,k})T\\ &+\boldsymbol{\alpha}(e_{1,k+1,y}) - \boldsymbol{\sigma}_{k,y}\Big)\Bigg),\end{aligned} \tag{22}$$

$$\begin{aligned}\mathbf{e}_{1,k+1,x} = -(L+B)\Big(&(\mathbf{x}_{1,k} + T\mathbf{x}_{2,k})\\ &-(\Delta\mathbf{x}_{1,k} + T\Delta\mathbf{x}_{2,k}) - (\underline{\mathbf{1}}x_k^0 + T\dot{x}_k^0)\Big),\end{aligned} \tag{23}$$

$$\begin{aligned}\mathbf{e}_{1,k+1,y} = -(L+B)\Big(&(\mathbf{y}_{1,k} + T\mathbf{y}_{2,k})\\ &-(\Delta\mathbf{y}_{1,k} + T\Delta\mathbf{y}_{2,k}) - (\underline{\mathbf{1}}y_k^0 + T\dot{y}_k^0)\Big).\end{aligned} \tag{24}$$

where $\mathbf{u}_{1,k} \in \mathbb{R}^{n \times n}$ is a diagonal matrix containing each component u_1^i for each agent. $\mathbf{m} \in \mathbb{R}^{n \times n}$ is a diagonal matrix containing the masses of each agent i. The control inputs for each quadrotor can be converted into desired angles $\varphi_{k,0}^i \in \mathbb{R}$ and $\theta_{k,0}^i \in \mathbb{R}$ using the following conversion:

$$\varphi_{k,0}^i = \arcsin\left(u_{x,k}^i \sin(\psi_k^i) - u_{y,k}^i \cos(\psi_k^i)\right), \tag{25}$$

$$\theta_{k,0}^i = \arcsin\left(\frac{u_{x,k}^i \cos(\psi_k^i) + u_{y,k}^i \sin(\psi_k^i)}{\cos(\varphi_k^i)}\right). \tag{26}$$

Using this, an internal controller can be designed for each quadrotor to allow the angles φ_k^i and θ_k^i to converge to their desired values.

4. SIMULATION

4.1 *Simulation design*

To assess and validate the performance of the designed control system, a numerical simulation was developed. MATLAB and Simulink were used to simulate a scenario in which 3 quadrotors were tasked with tracking a virtual leader. As the discrete-time sliding mode controller was designed for the 2D xy plane, it is assumed that each quadrotor tracks a desired altitude of $z_{k,d}^i = 1$. Furthermore, the desired yaw angle for each quadrotor is set to $\psi_{k,d}^i = 0$.

The virtual leader is designed to follow a sinusoidal trajectory for the desired x and y positions of the leader, such that the virtual leader travelled in a circular path. Finally, the formation is designed as an equilateral triangle, where the virtual leader is considered the centroid of the triangle at a height of 1m.

4.2 *Results*

The results were recorded in Simulink which were then plotted using MATLAB. Figure 1 demonstrates the x, y and z response of each quadrotor, as well as the trajectory of the virtual leader. The 3D positions of each drone are plotted in figure 2. To show the transient response of a single UAV in this system, figure 3 shows the roll, pitch and yaw angles of agent 1 over time.

To assess the stability of the system, the error values in equations (12) and (13) were plotted against time.

According to Definition 2, the system is stable if the error dynamics approach 0 as t approaches infinity. From figures 4 and 5, it can be seen that this is condition is met for both the position and velocity error dynamics, $\mathbf{e}_{1,k}^i$ and $\mathbf{e}_{2,k}^i$ for each agent.

5. CONCLUSIONS AND FUTURE WORK

In this paper, a new control system was proposed for the formation control of a group of quadrotor agents. The proposed control system, based on discrete-time sliding mode control, allows a group of agents to obtain a desired formation, and track a virtual leader following a trajectory. The performance of the proposed control system was validated using a numerical simulation developed in MATLAB and Simulink.

In future works, the proposed control system will be adapted for use in the nuclear industry to allow the group of agents to climb a gradient and find a hotspot in an

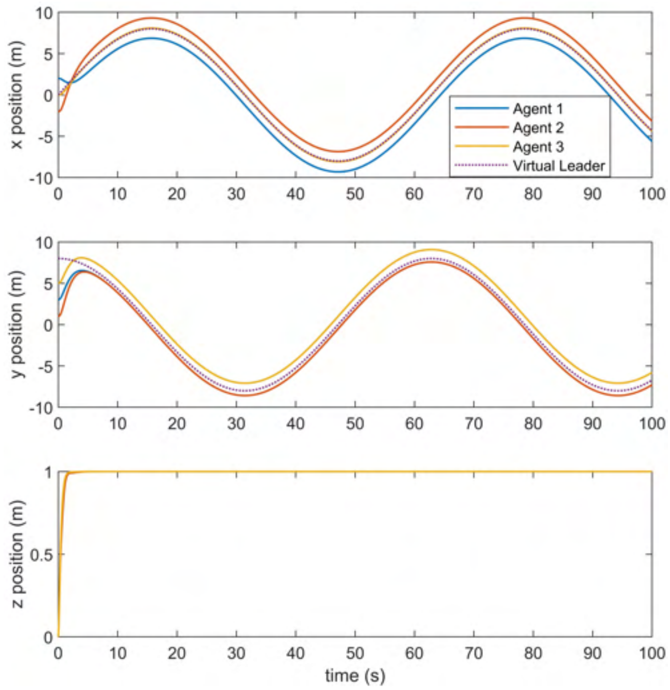

Fig. 1. The x, y and z position response of each agent, with the trajectory of the virtual leader

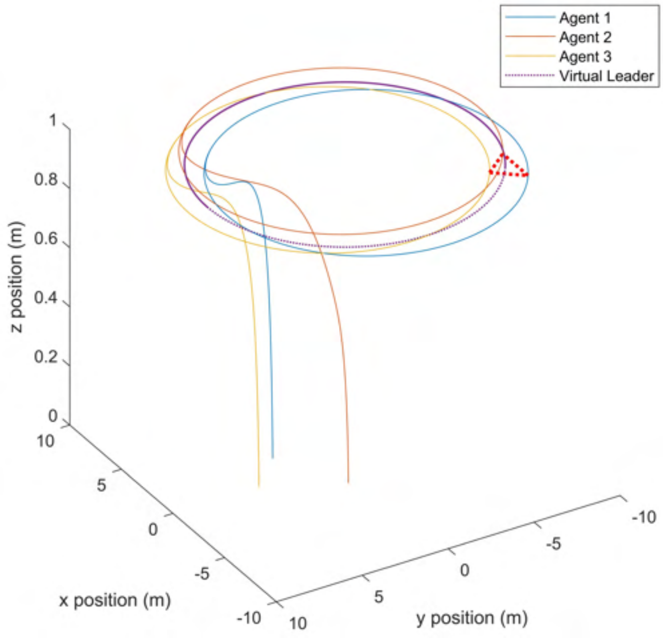

Fig. 2. 3D plot of all agent positions, with the position of the virtual leader

environmental field such as temperature or radiation. The algorithm will also be extended to also control for the altitude of the quadrotors, providing full position control over each of the quadrotor agents. Finally, this work can be extended by introducing collision avoidance, and evaluating the work through both a robustness analysis, as well as validating the work experimentally.

REFERENCES

Abdoli, H.M., Najafi, M., Izadi, I., and Sheikholeslam, F. (2018). Sliding mode approach for formation control of multi-agent systems with unknown nonlinear interactions. *ISA Transactions*, 80, 65–72.

Blanchet, M. and Confais, E. (2016). 4.0 industry: a new industrial challenge and a new economic model. *Revue Generale Nucleaire*, 12–16.

Can, A., Efstathiades, H., and Montazeri, A. (2020). Design of a chattering-free sliding mode control system for robust position control of a quadrotor. In *2020 International Conference Nonlinearity, Information and Robotics, NIR 2020*. Institute of Electrical and Electronics Engineers Inc.

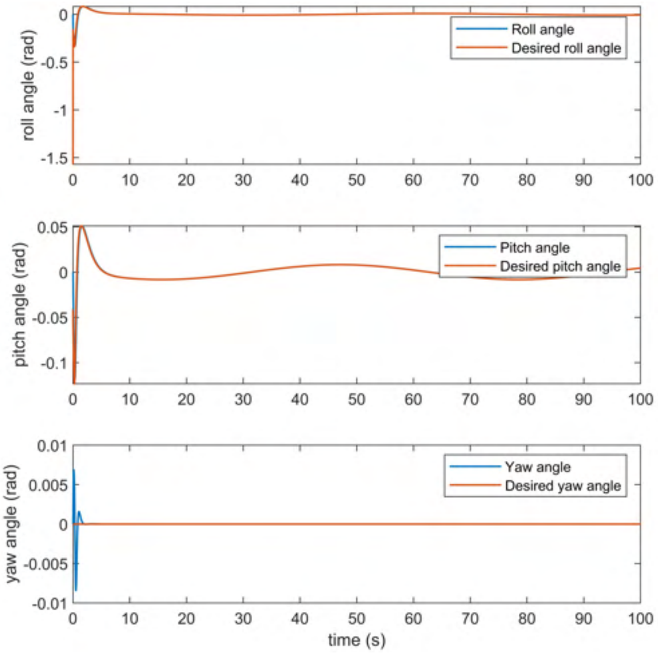

Fig. 3. Plot of the actual and desired attitude of agent 1

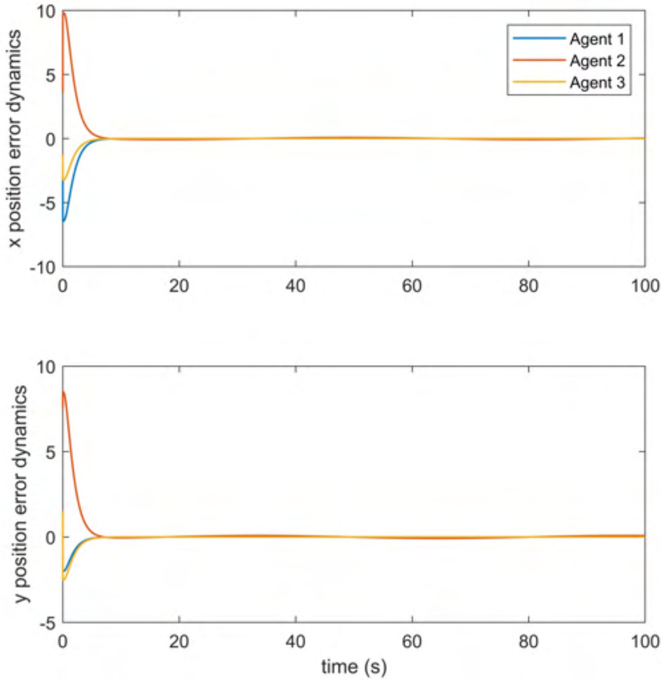

Fig. 4. Error plots for each agent for the x and y position channels

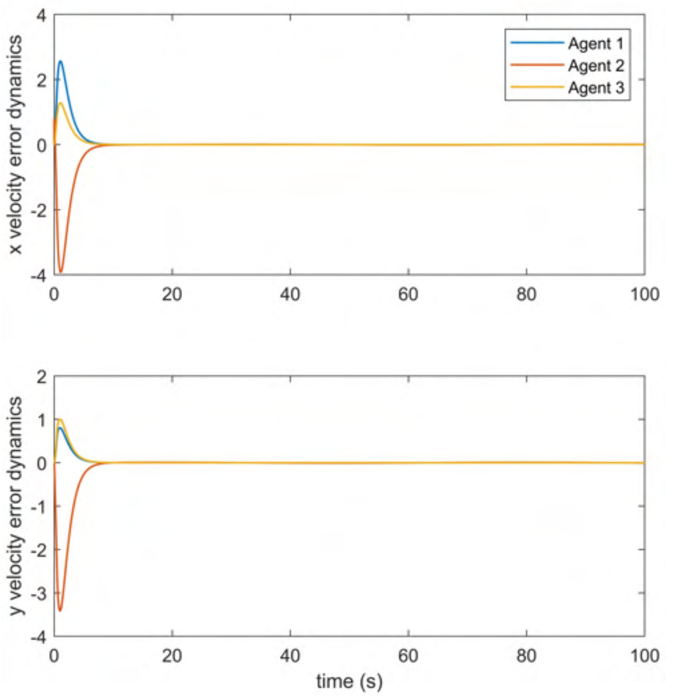

Fig. 5. Error plots for each agent for the x and y position channels

Gao, W., Wang, Y., and Homaifa, A. (1995). Discrete-time variable structure control systems. *IEEE Transactions on Industrial Electronics*, 42(2), 117–122. doi:10.1109/41.370376.

Han, J.S., Kim, T.I., Oh, T.H., Kim, Y.S., Lee, J.H., Kim, S.O., Lee, S.S., Lee, S.H., and Cho, D.I.D. (2019). Error-dynamics-based performance shaping methodology for discrete-time sliding mode control with disturbance observer. volume 52, 460–464. Elsevier B.V.

He, Z., Zhao, L., and Zhao, L. (2016). Robust chattering free backstepping/backstepping sliding mode control for quadrotor hovering. In *2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, 616–620.

Imran, I., Stolkin, R., and Montazeri, A. (2021). Adaptive Closed-Loop Identification and Tracking Control of an Aerial Vehicle with Unknown Inertia Parameters. In *19th IFAC SYMPOSIUM on SYSTEM IDENTIFICATION: learning models for decision and control, SYSID 2021*.

Intel (2020). External synchronization of Intel RealSense™ depth cameras. URL `https://dev.intelrealsense.com/docs/external-synchronization-of-intel-realsense-depth-cameras`.

Khan, H.S. and Kadri, M.B. (2015). Position control of quadrotor by embedded PID control with hardware in loop simulation. In *17th IEEE International Multi Topic Conference: Collaborative and Sustainable Development of Technologies, IEEE INMIC 2014 - Proceedings*, 395–400. Institute of Electrical and Electronics Engineers Inc.

Liu, H., Gao, B., Shen, Y., Hussain, F., Addis, D., and Kai Pan, C. (2018). Comparison of Sick and Hokuyo UTM-30LX laser sensors in canopy detection for variable-rate sprayer. *Information Processing in Agriculture*, 5(4), 504–515.

Liu, T. and Jiang, Z.P. (2013). Distributed formation control of nonholonomic mobile robots without global position measurements. *Automatica*, 49(2), 592–600.

Lu, C., Lyu, J., Zhang, L., Gong, A., Fan, Y., Yan, J., and Li, X. (2020). Nuclear power plants with artificial intelligence in industry 4.0 era: Top-level design and current applications—a systemic review. *IEEE Access*, 8, 194315–194332.

Martins, L., Cardeira, C., and Oliveira, P. (2019). Linear quadratic regulator for trajectory tracking of a quadrotor. *IFAC-PapersOnLine*, 52(12), 176–181. 21st IFAC Symposium on Automatic Control in Aerospace ACA 2019.

Nokhodberiz, N.S., Nemati, H., and Montazeri, A. (2019). Event-triggered based state estimation for autonomous operation of an aerial robotic vehicle. *IFAC-PapersOnLine*, 52(13), 2348–2353.

Pan, Y., Yang, C., Pan, L., and Yu, H. (2018). Integral Sliding Mode Control: Performance, Modification, and Improvement. *IEEE Transactions on Industrial Informatics*, 14(7), 3087–3096.

Pratama, B., Muis, A., Subiantoro, A., Djemai, M., and Ben Atitallah, R. (2018). Quadcopter trajectory tracking and attitude control based on euler angle limitation. In *2018 6th International Conference on Control Engineering and Information Technology, CEIT 2018*. Institute of Electrical and Electronics Engineers Inc.

Sarkar, M.K., Arkdev, and Singh, S.S.K. (2017). Sliding mode control: A higher order and event triggered based approach for nonlinear uncertain systems. In *2017 8th Industrial Automation and Electromechanical Engineering Conference, IEMECON 2017*, 208–211. Institute of Electrical and Electronics Engineers Inc.

Slamtec (n.d.). RPLIDAR-A1 360°Laser Range Scanner. URL `https://www.slamtec.com/en/Lidar/A1`.

Tian, B., Cui, J., Lu, H., Liu, L., and Zong, Q. (2021). Attitude control of UAVs based on event-triggered supertwisting algorithm. *IEEE Transactions on Industrial Informatics*, 17(2), 1029–1038.

Wang, H., Zhao, H., Zhang, J., Ma, D., Li, J., and Wei, J. (2020). Survey on unmanned aerial vehicle networks: A cyber physical system perspective. *IEEE Communications Surveys Tutorials*, 22(2), 1027–1070.

Wang, W. and Yu, X. (2017). Chattering free and nonsingular terminal sliding mode control for attitude tracking of a quadrotor. In *Proceedings of the 29th Chinese Control and Decision Conference, CCDC 2017*, 719–723. Institute of Electrical and Electronics Engineers Inc.

Wang, Y., Wu, Q., Wang, Y., and Yu, D. (2013). Consensus algorithm for multiple quadrotor systems under fixed and switching topologies. *Journal of Systems Engineering and Electronics*, 24(5), 818–827.

Wen, G., Duan, Z., Li, Z., and Chen, G. (2012). Flocking of multi-agent dynamical systems with intermittent nonlinear velocity measurements. *International Journal of Robust and Nonlinear Control*, 22(16), 1790–1805.

Wu, F., Chen, J., and Liang, Y. (2017). Leader-Follower Formation Control for Quadrotors. *IOP Conference Series: Materials Science and Engineering*, 187(1), 012016.

Xiong, J.J. and Zhang, G. (2016). Discrete-time sliding mode control for a quadrotor UAV. *Optik*, 127(8), 3718–3722.