Teknomekanik, Vol. 8, No. 2, pp. 223-242, December 2025
e-ISSN: 2621-8720 p-ISSN: 2621-9980

Monitoring mission for multi-drones using decentralized chaos-bidding consensus with backstepping control via lyapunov barrier functions

Muhammad Zakiyullah Romdlony[1], Rashad Abul Khayr[2*], Yul Yunazwin Nazaruddin[3], Tua Agustinus Tamba[4], and Md. Abdus Samad Kamal[2]

[1] Department of Electrical Engineering, School of Electrical Engineering, Telkom University, **Indonesia**
[2] Domain of Mechanical Science and Technology, School of Science and Technology, Gunma University, **Japan**
[3] Department of Engineering Physics, Faculty of Industrial Technology, Institut Teknologi Bandung, **Indonesia**
[4] Department of Electrical Engineering, Faculty of Engineering Technology, Parahyangan Catholic University, **Indonesia**

* Corresponding Author: t252b603@gunma-u.ac.jp
Received: 25 Apr 2025; *1st Revised:* 17 Jul 2025; *2nd Revised*: 08 Nov 2025; *Accepted:* 13 Nov 2025

Cite this https://doi.org/10.24036/teknomekanik.v8i2.36072

Abstract: The mobility and flexibility of a quadrotor make it a popular choice for monitoring missions in remote areas. However, remote environments introduce constraints due to limited charging and communication stations that must be considered, alongside the possibility of collision with the environment. To ensure the quadrotor task was completed, a decentralized chaos-bidding consensus for decentralized task allocation was proposed, accompanied by control, Lyapunov, and barrier functions. These functions were simplified using the backstepping method to ensure the quadrotor's safety during task execution. The proposed method was evaluated through numerical simulation in multiple situations. The results indicate a minimum of 3% reduction in task completion time compared to other methods. When the battery constraint was applied, the proposed method successfully directed the drone to return to base before battery depletion and reassigned the task to other available quadrotors, thereby reducing the overall completion time for the entire system. Furthermore, this framework demonstrates the potential to support long-duration missions where continuous operation is required without relying heavily on ground control. The decentralized nature of the system also increases scalability, allowing multiple quadrotors to cooperate efficiently under dynamic environmental conditions. These advantages highlight the relevance of the proposed control strategy for practical field deployment, particularly in inaccessible locations.

Keywords: UAV; safety control; backstepping method; Lyapunov barrier function; decentralized scheduling

1. Introduction

The rising popularity of unmanned aerial vehicles (UAVs), particularly quadrotors, is driven by declining production costs and their high mobility and flexibility [1], [2], [3]. One of the applications the drone can perform is a monitoring task. Quadrotors are often deployed in remote areas where charging infrastructure and communication support are limited [4], [5], [6][7], [8]. Despite the limitation, the quadrotor is still invaluable for this task, like a search and rescue in a recently disaster-affected area [9] or to monitor any anomaly for a remote power station [10], [11], [12]. The communication problem can be mitigated by decentralizing communication between drones and limiting ground control reliance during operations [7], [8], [13]. Meanwhile, battery constraints may be addressed through scheduling strategies to maximize energy efficiency [14], [15]. Numerous

studies have focused on optimizing quadrotor power usage, either directly [16], [17] or indirectly [18], [19].

Infrastructure limitations are not the only challenges encountered by quadrotors in remote environments. The environment that a quadrotor operates in can greatly impact the quadrotor's performance [20], [21], [22]. Therefore, safety must be considered along with stability and energy efficiency. However, the stability and safety problems are usually solved separately. Stability may be achieved through simple PID control or more advanced nonlinear methods [23] for nonlinear systems by utilizing the properties of the Lyapunov function [24], [25] or even with the help of artificial intelligence [26]. Safety can be addressed using AI-based approaches or Lyapunov-derived barrier functions [27]. Each of these methods has its own strengths, such as the simplicity of PID, the hand-off approach of artificial intelligence, or the skipping of finding a derivative solution for the solution utilizing the Lyapunov function. Of course, each method also has its own weaknesses, such as the need to find a derivative solution for PID, the need for training data for artificial intelligence, or the Lyapunov function itself for a solution utilizing the Lyapunov function. As for the drone application, the reliability of the Lyapunov function for both safety and stability problems to combat the complexity of the drone dynamics. As for the weakness of the Lyapunov-based method, a backstepping method can simplify the search for the Lyapunov function [27], [28], [29].

However, the method tackles the stability and safety problem separately, which can result in a self-eliminating force between the stability solution and the safety solution. A method that combines both stability and safety can elevate this problem, such as Prioritized Communication Learning (PICO) [26], PID-Barrier Function (PID-BF) [30], or the Control Lyapunov-Barrier Function (CLBF) [31]. The PICO method utilized a learning-based search algorithm with a grid-based map to help navigation; however, this method has the problem of artificial intelligence requiring previous data. The PID-BF used both PID and Control Barrier Function (CBF) to ensure stability and safety; however, the use of PID can increase computational load from the need to solve the system derivative to gain a solution. The CLBF sidesteps the problem from PID by implementing the Control Lyapunov Function (CLF); however, with the implementation of CLF and CBF comes the complexity in finding both functions that can work for the specified system. Therefore, backstepping approach to the control of Lyapunov barrier function is proposed to simplify its process for complex systems. In addition, with the help of chaos-bidding consensus, the suggested method can be used to let drones safely fly to their destination without the need for a central control system.

2. Material and methods

The Control Lyapunov-Barrier Function (CLBF) is a powerful control method for stability and safety for a nonlinear system. Due to its complexity, this method has rarely been applied. In this study, the backstepping method, which can be used to construct either a Lyapunov or a barrier function, was employed to simplify the formulation of CLBF for quadrotor dynamics. This control method can be integrated with the chaos-bidding consensus to eliminate quadrotor reliance on a centralized control system in a multiple-agent environment. The overview of the proposed method is presented in Figure 1.

2.1 Trajectory tracking

The quadrotor is a drone that uses four motors to move in a 3-dimensional environment with 6-DoF (Degrees of Freedom), which consists of rotational movement (ϕ, θ, ψ) and translational movement (x, y, z) [32]. For this research, the X configuration for the quadrotor was employed, which utilized all four motors for pitch and roll control to enhance stability. The schematic for the X configuration can be seen in Figure 2.

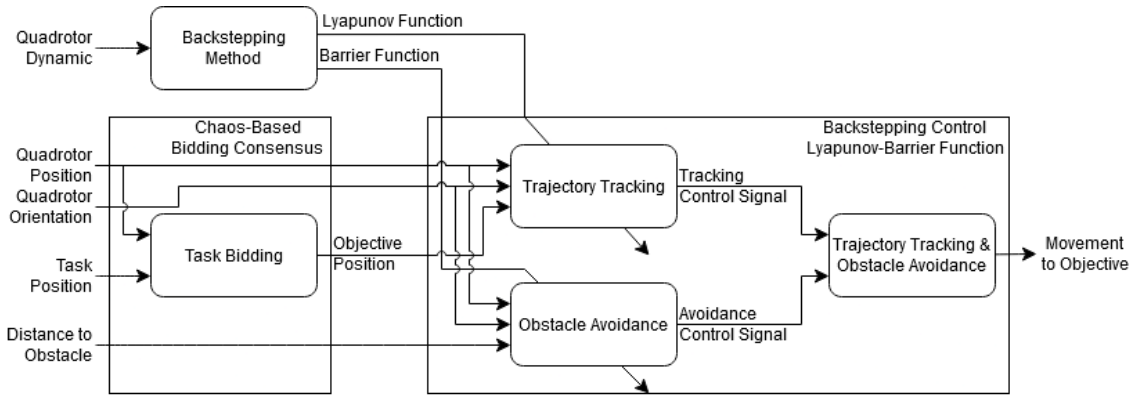


Figure 1. Data processing overview for the proposed method

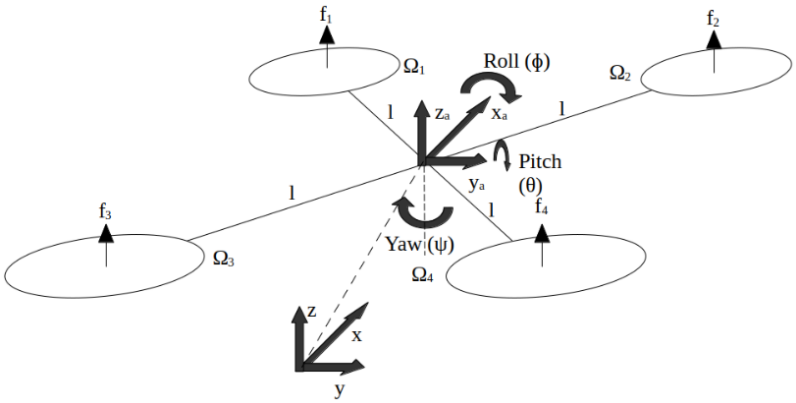


Figure 2. Quadrotor schematics for X configuration

Quadrotors, as mentioned above, have 6-DoF despite only having 4 control inputs in the form of 4 motors' rotational speed (Ω), making it an underactuated system. The 4 motor speeds can be converted into force applied to the quadrotor with the help of the thrust factor (b), distance between the motor and the center of the quadrotor (l), and drag factor (d), which can be seen in (1).

$$U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ bl(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \\ bl(-\Omega_1^2 - \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ d(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \end{bmatrix} \tag{1}$$

Three of the forces applied, including U_2, U_3, and U_4, are directly responsible for the roll, pitch, and yaw, respectively [33]. The acceleration of each variable can be calculated by using (2) with I_x, I_y, and I_z as the quadrotor's inertia at each axis. The last force, U_1, were shared between the calculation of quadrotor translational acceleration a dependent on quadrotor roll, pitch, yaw, gravitational acceleration (g), and quadrotor mass (m) [33]. The equation to calculate the translational acceleration can be seen in (3).

$$\ddot{\phi} = \dot{\theta}\dot{\psi}\left(\frac{I_y - I_z}{I_x}\right) + \frac{l}{I_x}U_2 \tag{2a}$$

$$\ddot{\theta} = \dot{\phi}\dot{\psi}\left(\frac{I_z - I_y}{I_y}\right) + \frac{l}{I_y}U_3 \tag{2b}$$

$$\ddot{\psi} = \dot{\theta}\dot{\phi}\left(\frac{I_x - I_y}{I_z}\right) + \frac{l}{I_z}U_4 \tag{2c}$$

$$\ddot{x} = \big(cos(\phi)sin(\theta)cos(\psi) + sin(\phi)sin(\psi)\big)\frac{U_1}{m} \tag{3a}$$

$$\ddot{y} = cos(\phi)sin(\theta)sin(\psi) - sin(\phi)cos(\psi)\frac{U_1}{m} \tag{3b}$$

$$\ddot{z} = cos(\phi)cos(\psi)\frac{U_1}{m} - g \tag{3c}$$

In order to simplify the controller for this system, a virtual input for x and y movement can be made. These virtual inputs are defined in (4) with the new acceleration for x and y in (5).

$$U_x = cos(\phi)sin(\theta)cos(\psi) + sin(\phi)sin(\psi) \tag{4a}$$

$$U_y = cos(\phi)sin(\theta)sin(\psi) - sin(\phi)cos(\psi) \tag{4b}$$

$$\ddot{x} = \frac{U_1}{m}U_x \tag{5a}$$

$$\ddot{y} = \frac{U_1}{m}U_y \tag{5b}$$

The acceleration defined above can be controlled using a control Lyapunov function (CLF) to reach a desired position. CLF managed it by utilizing the Lyapunov function (V) properties to check system stability [34]. CLF can achieve stability if the Lyapunov function (V) satisfied (6) where $L_f V(x)$ and $L_g V(x)$ is the lie derivative of V to $f(x)$ and $g(x)$ respectively.

$$L_f V(x) < 0 \quad \forall x \in \{w \in \mathbb{R}^n \setminus \{0\} \vee L_g V(w) = 0\} \tag{6}$$

The CLF, while powerful, can be rather tedious, especially in finding the appropriate Lyapunov function. That is why the backstepping method was used, which is a recursive method of developing nonlinear feedback controllers by focusing on a known stable subsystem and progressively stabilizing each outer system. Consider a generalized acceleration where k is the position or rotation, k_1 is the acceleration, and k_2 is the speed.

$$\dot{k}_1 = k_2 \tag{7}$$

Stabilizing the k_1 was done by defining an appropriate Lyapunov function, as can be seen in (8) and (9) with a as a definite positive constant.

$$V_1 = a\frac{1}{2}k_1^2 \tag{8}$$

$$\dot{V}_1 = ak_1\dot{k}_1 \tag{9}$$

The above function can satisfy (6) if a virtual input (w) is defined as shown in (10). This virtual input will be matched by the k_2, as such, the error between the two is defined in (11).

$$\dot{k}_1 = w = -k_1 \tag{10}$$

$$h = k_2 - w \tag{11}$$

From the virtual input and error, the acceleration function in (7) is as follows.

$$\begin{aligned} \dot{k}_1 &= h - k_1 \\ \dot{h} &= f(k) + g(k)u + k_1 \end{aligned} \tag{12}$$

Regarding the formula, the system's Lyapunov function can be stated by combining the Lyapunov function of k_1 and h so acceleration stability can be reached while ensuring that both k_2 and w convergence. The Lyapunov function can be seen in (13) and (14).

$$V_{(k,h)} = V_1 + V_h = \frac{1}{2}k_1^2 + \frac{1}{2}h^2 \tag{13}$$

$$\dot{V}_{(k,h)} = -k_1^2 + hf(k) + hg(k)u + hk_1 \tag{14}$$

The functions in (13) and (14) meet (6) if the input (u) is defined in (15).

$$u = -\frac{1}{g(x)}(h + f(k) + k_1) \tag{15}$$

From the new input, it can be specified to each input and virtual input for the quadrotor dynamics, which can be seen in (16) and (17).

$$U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} -\dfrac{m}{cos(\phi)cos(\theta)}(\dot{z} + z - g) \\ -\dfrac{I_x}{l}\left((\dot{\phi} - \phi) + \dot{\theta}\dot{\psi}\left(\dfrac{I_y - I_z}{I_x}\right)\right) \\ -\dfrac{I_y}{l}\left((\dot{\theta} - \theta) + \dot{\phi}\dot{\psi}\left(\dfrac{I_z - I_x}{I_y}\right)\right) \\ -\dfrac{I_z}{l}\left((\dot{\psi} - \psi) + \dot{\theta}\dot{\phi}\left(\dfrac{I_x - I_y}{I_z}\right)\right) \end{bmatrix} \tag{16}$$

$$U_x = -\frac{m}{U_1}(\dot{x} - x) \tag{17a}$$

$$U_y = -\frac{m}{U_1}(\dot{y} - y) \tag{17b}$$

2.2 Trajectory and obstacle avoidance

A new input for the quadrotor was made to ensure stability. However, if the quadrotor meets an obstacle, a collision between the quadrotor and the obstacle might happen. That is where an obstacle avoidance system is needed. In this research, a control barrier function (CBF) was used, which is a modified control Lyapunov function to solve the safety problem, where there exists no system trajectory that makes the state enter the unsafe state, given the initial state not inside the unsafe

state in a finite time [35]. Given a set of unsafe states X, if the function B satisfies (18), then it is called a control barrier function.

$$k \in X \Rightarrow B(k) > 0 \tag{18a}$$

$$L_g B(k) = 0 \Rightarrow L_f B(k) < 0 \tag{18b}$$

$$,\{k \in X \vee B(k) \leq 0\} \neq \emptyset \tag{18c}$$

Since the CBF is a modified CLF, the backstepping method can be used to make the barrier function as well. Using the generalized acceleration (7) as our foundation, rather than defining a Lyapunov function, a barrier function as seen in (19) and (20) was defined with the virtual input in (21).

$$B = \frac{1}{2} e^{-k_1^2} \tag{19}$$

$$\dot{B} = -e^{-k_1^2}\left(k_1 \dot{k}_1\right) \tag{20}$$

$$\dot{k}_1 = w = k_1 \tag{21}$$

A new acceleration is found as seen in (22)

$$\begin{aligned} \dot{k}_1 &= h + \mathrm{k} \\ \mathrm{l}\dot{h} &= f(k) + g(k)u - k_1 \end{aligned} \tag{22}$$

While ensuring the safety in terms of position by limiting the acceleration, the error between speed and virtual input needs to converge to zero, rather than avoiding a certain area. In order to achieve convergence, Lyapunov function was used for the error, resulting in system control as follows.

$$W_{(k.h)} = B + V_h = \frac{1}{2} e^{-k_1^2} + \frac{1}{2} h^2 \tag{23}$$

$$\dot{W}_{(k,h)} = -k_1^2 e^{-k_1^2} - h e^{-k_1^2} + h f(k) + h g(k) u - k_1 h \tag{24}$$

$$u = \frac{1}{g(k)}\left(e^{-k_1^2} - f(k) - h + \mathrm{k}\right) \tag{25}$$

The system control, as described in (23) and (24), uses a similar principle to that in [31] where both CBF and CLF were combined to ensure stability while maintaining safety as long as the function satisfied (26).

$$W(k) > 0 \quad \forall k \in X \tag{26a}$$

$$L_f W(k) < 0 \quad \forall k \in \left\{w \in \mathbb{R}^n \setminus \left(X \cup \{0\} \vee L_g W(w) = 0\right)\right\} \tag{26b}$$

As the obstacle only appeared in Cartesian coordinates, this input was only for translational movement, resulting in the following new input.

$$U_1 = \frac{m}{cos(\phi)cos(\theta)}\left(e^{-z^2} + g - \dot{z} + 2z\right) \tag{27a}$$

$$U_x = -\frac{m}{U_1}\left(e^{-x^2} - \dot{x} + 2x\right) \tag{27b}$$

$$U_y = -\frac{m}{U_1}\left(e^{-y^2} - \dot{y} + 2y\right) \tag{27c}$$

Using a similar concept to [31], the trajectory tracking and the obstacle avoidance were combined by using a sufficiently smooth function, such as the hyperbolic tangent, as a switching function to satisfy (26), making a new input as follows.

$$u = \begin{cases} -\dfrac{1}{g(x)}\left(h + f(x)\right) & \forall x \notin X \\ \dfrac{1}{g(x)}\left(e^{-x_1^2} - f(x) - h + x_1\right) - \dfrac{1}{g(x)}\left(h + f(x)\right) & \forall x \in X \end{cases} \tag{28}$$

2.3 Multi-agent consensus

The control system that was defined above was used to ensure the success of a single quadrotor in performing the assigned task. When it came to multi-agent systems, each quadrotor had to establish a common understanding of its intended destination. To combat this problem, a chaos-based bidding consensus was used, a decentralized task assignment where the agent reached a consensus by bidding on the tasks they wished to fulfill [36], which provided an efficient way for a many-to-many interaction [37]. Utilizing the A* algorithm, the quadrotor will find the minimum distance of travel for the n targets to bid using a cost function (J) that can be seen in (29) with D being the distance between a previously chosen node and the next [38]. Since only the quadrotor's distance and target were considered, the best bidder for this algorithm is the one who can bid with the lowest value possible.

$$J = \sum_{i=1}^{n} D_{i-1,i} \tag{29}$$

This system promotes the bidding of trajectories up to n targets. The bid value was kept under a singular shared queue, which is similar to the method proposed by Mahato *et al.* [39]. Unlike [39], as mentioned above, the agent was allowed to bid a trajectory instead of a single target to minimize time when the agent was forced to take a target that was far away. A graphic representation of this method can be seen in Figure 3. Algorithm 1 shows the pseudocode of this method, where D_{Qi}, D_{iminj}, and T_i are the distance between the quadrotor i and waypoint, the smallest distance for the quadrotor i to waypoint j, and the bid value on the waypoint j, respectively.

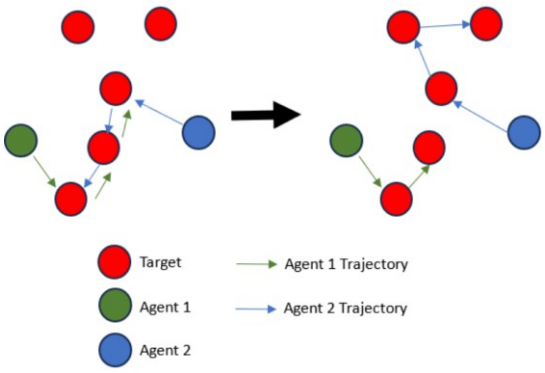


Figure 3. Representation of bidding consensus with disputed bidding and new consensus

Algorithm 1 Chaos-based bidding consensus

DEFINE WAYPOINT
DEFINE ORIGIN
While there are unassigned waypoints
 For remaining waypoints
  If $D_{Q1} < D_{1min1}$ and $D_{Q1} < T_i$
   $D_{1min1} = D_{Q1}$
  Else if $D_{Q1} < D_{1min2}$ and $D_{Q1} < T_i$
   $D_{1min2} = D_{Q1}$
  Else if $D_{Q1} < D_{1min3}$ and $D_{Q1} < T_i$
   $D_{1min3} = D_{Q1}$
  End if
 End for
 PLACE D_{1min1} WAYPOINT BID
 PLACE D_{1min2} WAYPOINT BID
 PLACE D_{1min3} WAYPOINT BID
 For remaining waypoints
  If $D_{Q2} < D_{2min1}$ and $D_{Q2} < T_i$
   $D_{2min1} = D_{Q2}$
  Else if $D_{Q2} < D_{2min2}$ and $D_{Q2} < T_i$
   $D_{2min2} = D_{Q2}$
  Else if $D_{Q2} < D_{2min3}$ and $D_{Q2} < T_i$
   $D_{2min3} = D_{Q2}$
  End if
 End for
 PLACE D_{2min1} WAYPOINT BID
 PLACE D_{2min2} WAYPOINT BID
 PLACE D_{2min3} WAYPOINT BID
 For remaining waypoint
  If bid quadrotor1 < bid quadrotor 2
   GIVE WAYPOINT TO QUADROTOR 1
  Else
   GIVE WAYPOINT TO QUADROTOR 2
  End if
 End for
End While

2.4 Simulation parameter

The quadrotor controller and bidding consensus were simulated under three conditions with the help of Octave, for the simulation software. In the first scenario, the quadrotor was directed to circle around the origin point at an altitude of 10 m above, using a radius of 5 m. The second scenario is similar to the first, but it included an obstacle in the form of a sphere, with a radius of 0.5 m, and a core located at coordinates. Lastly, two quadrotors were instructed to visit a waypoint in the shortest amount of time, with each waypoint requiring only one quadrotor visit, all while avoiding collisions with a static obstacle at coordinates (20,10,10), which is a sphere with a radius of 0.5 m. The simulation was done using parameters specified in Table 1, similar to [40]. This condition was then used to compare the proposed method with 2 other methods.

One more simulation was conducted based on the third condition with an added constraint in the form the management of quadrotor power consumption. Each quadrotor carried a battery with a total amount of charge of 2 Ah. Two quadrotors were given the same task as in the simulation of the multi-

agent consensus. They had a total weight of 1 kg, and each motor followed the parameters in Table 2. By adding an overriding flag to return the quadrotor, the task could be bid again as the quadrotor returned to the base. The flag was checking on the quadrotor's distance with the base and comparing it with the battery remaining charge. The result of this experiment was evaluated with numerical simulation in the form of tracking error, position error, obstacle distance, and remaining charge.

Table 1. Simulation parameters

Variable	Value
l	0.23 m
$4I_x, I_y, I_z$	7.5 x 10^{-3} kgm^2
m	0.65 kg
g	9.81 m/s^2

Table 2. Motor thrust table

Throttle	Current (A)	Thrust (g)	RPM
50%	3.4	370	7000
65%	4.4	410	7800
75%	5.3	470	8500
85%	7.4	610	9300
100%	8.5	700	10000

3. Results and discussion

In this section, the simulation results were discussed using the method and parameters that were defined in the previous section. A comparison to show the efficacy of the proposed method was also provided.

3.1 Trajectory tracking and obstacle avoidance

Figure 4 shows the movement of the quadrotor for trajectory tracking using the control system and simulation parameters that were defined above, without the battery constraint. Figure 5 shows the tracking error for the trajectory tracking under low-fidelity simulation.

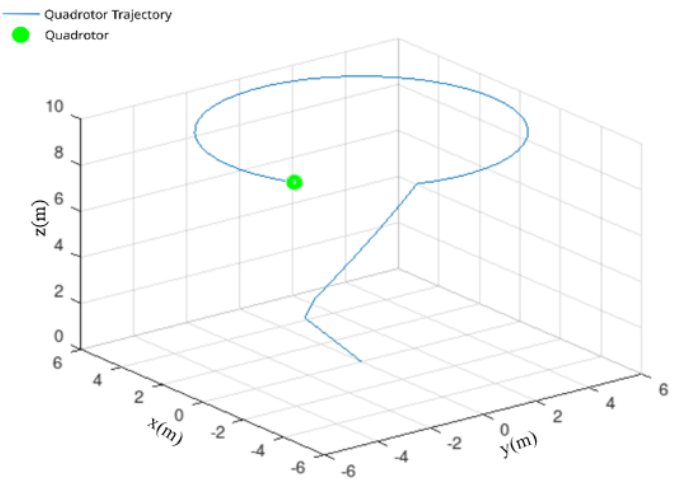


Figure 4. Quadrotor simulation result for circular movement trajectory tracking with center point at (0, 0, 10) and radius of 5 m

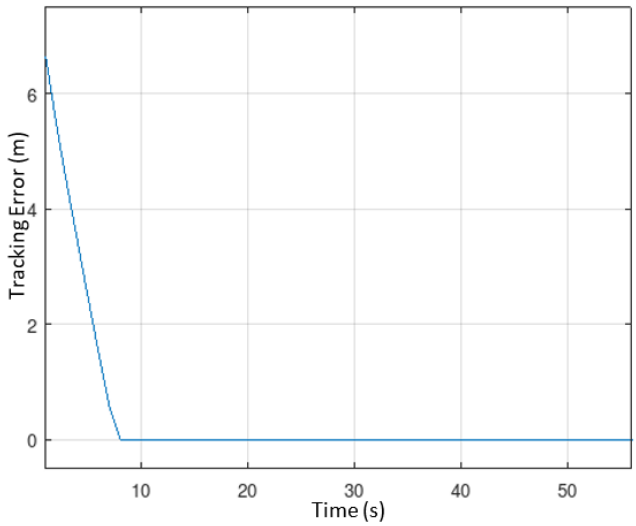


Figure 5. Quadrotor tracking error for circular movement trajectory tracking with center point at (0, 0, 10) and radius of 5 m

The simple trajectory tracking simulation shows that this method reached and followed the circular trajectory in 8 seconds. As for the reactivity of the method, a static obstacle was added in the trajectory with results that can be seen in Figure 6 for the trajectory and Figure 7 for the tracking error and the distance to obstacle.

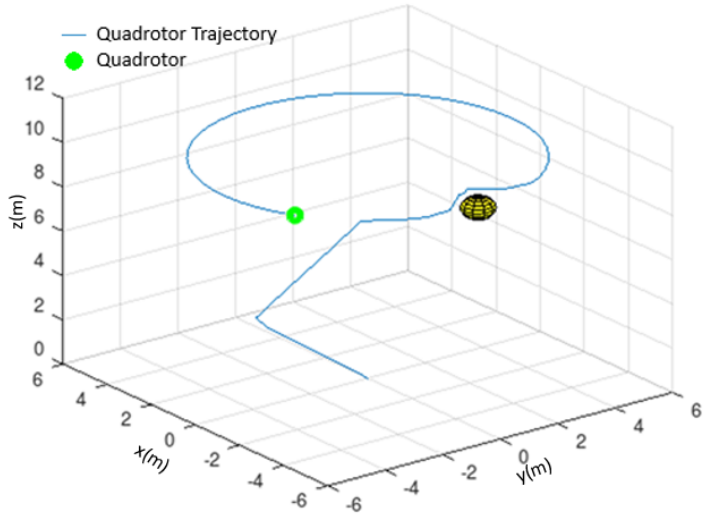
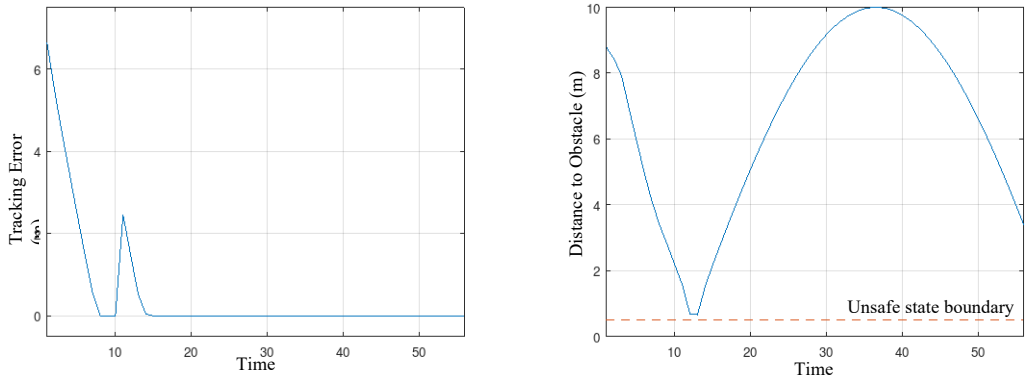


Figure 6. Quadrotor simulation result for circular movement trajectory tracking with center point at (0, 0, 10) and radius of 5 m while avoiding collision with a spherical static obstacle at (-5, 0, 10) and radius of 0.5 m

As shown in this simple, low-fidelity simulation, the proposed method managed to ensure the quadrotor tracked the circular trajectory by converging the error to 0 in 8 seconds from a 7 m error. With the added obstacle, the quadrotor managed to avoid it with a 0.2 m distance to the surface of the obstacle as the minimum value. The avoidance maneuver required 3 seconds from the start of the maneuver until the error reached 0 m again. Comparing it with PID-BF[30], Figure 8 and 9 tell that the trajectory tracking at the start are similar. Once the obstacle was encountered, the time needed for PID-BF to return was 12 seconds slower than the proposed method, resulting in a 2-second lag when both methods returned to a stable tracking.



(a) (b)

Figure 7. Quadrotor (a) tracking error and (b) distance to obstacle for circular movement trajectory tracking with center point at (0, 0, 10) and radius of 5 m while avoiding collision with a spherical static obstacle at (-5, 0, 10) and radius of 0.5 m

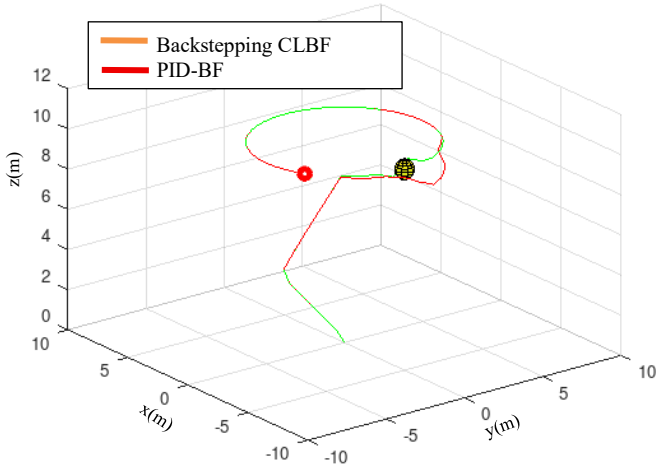
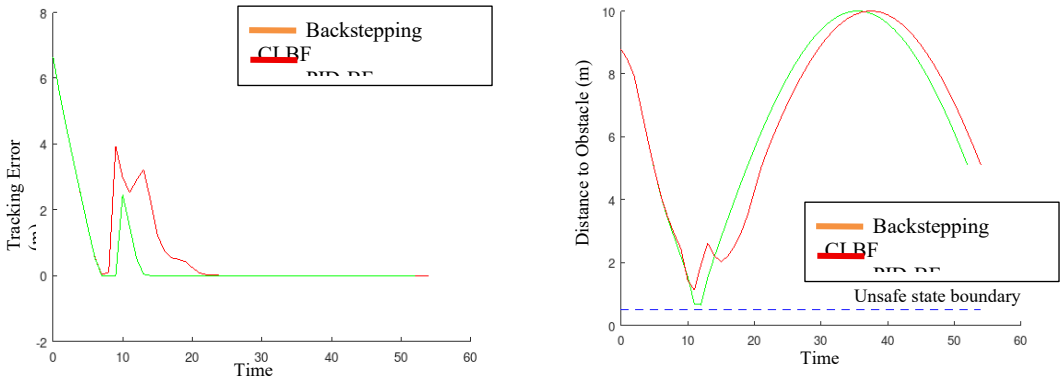


Figure 8. Quadrotor simulation result comparison for circular movement trajectory tracking with center point at (0, 0, 10) and radius of 5 m while avoiding collision with a spherical static obstacle at (-5, 0, 10) and radius of 0.5 m



(a) (b)

Figure 9. Quadrotor (a) tracking error and (b) distance to obstacle comparison for circular movement trajectory tracking with center point at (0, 0, 10) and radius of 5 m while avoiding collision with a spherical static obstacle at (-5, 0, 10) and radius of 0.5 m

3.2 Multi-agent tracking and obstacle avoidance

Figure 10 shows the consensus result of the chaos-based bidding consensus when 8 random points needed to be visited once, with at least a single quadrotor for two quadrotors. The task was evenly distributed between the quadrotors with respect to the distance the quadrotor must traverse using the bid each quadrotor put forward.

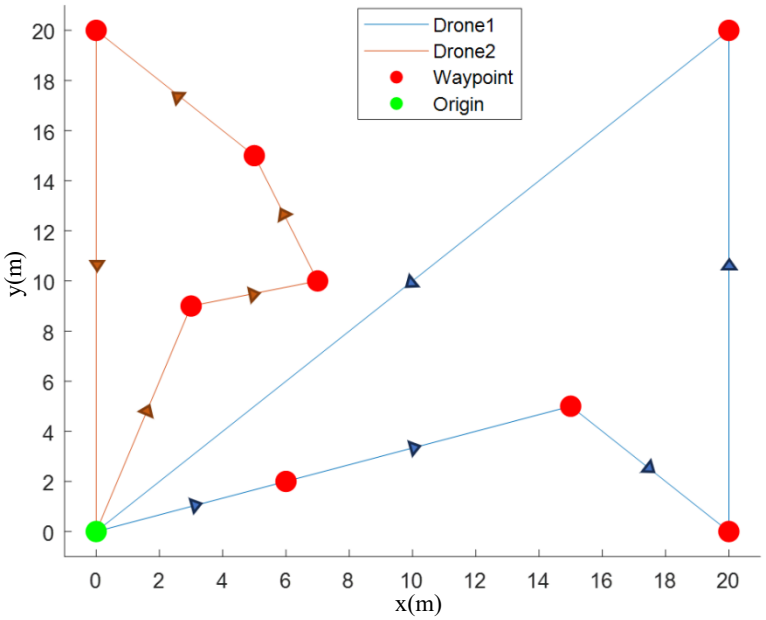


Figure 10. Task assignment using chaos-based bidding consensus for two agents and 8 targets with the expected trajectory

The two quadrotors then go to the previously assigned task using the proposed method. With an added obstacle in the way of quadrotor 1, the movement for both quadrotors can be seen in Figure 11.

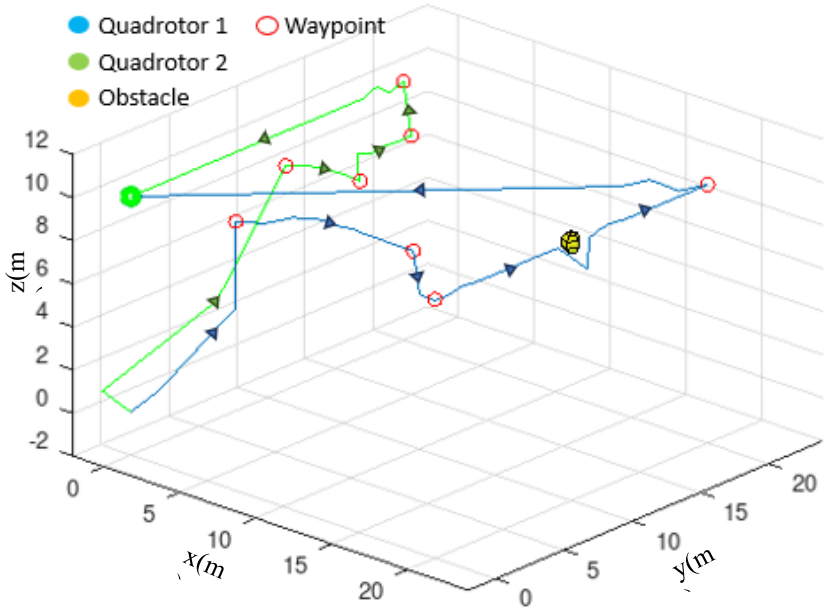
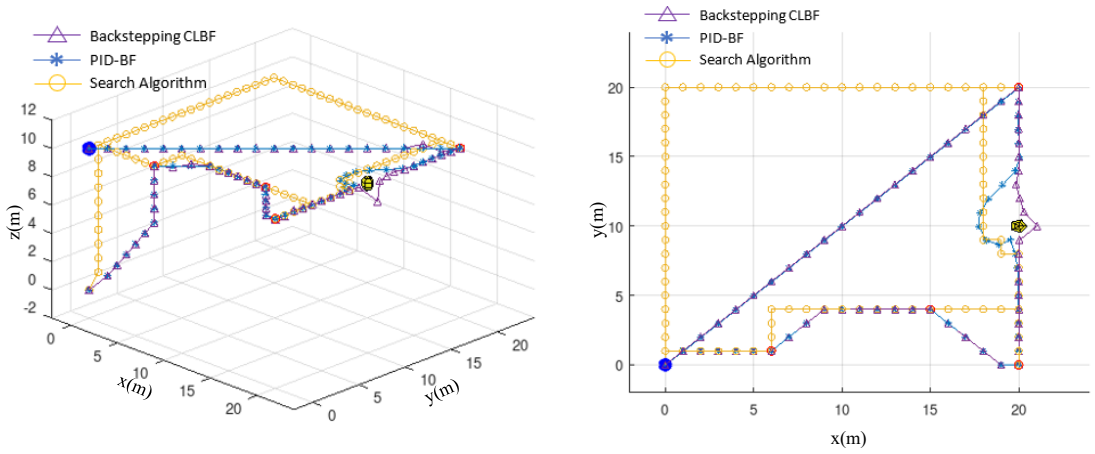


Figure 11. Quadrotor simulation result for waypoint following assignment with chaos-based consensus and collision avoidance for another agent and a static spherical obstacle at (20, 10, 10)

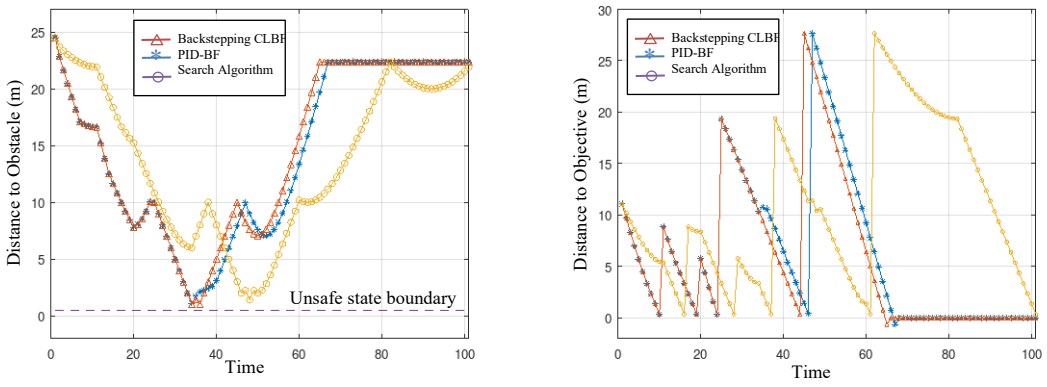
The simulation shows that the proposed method can be used to ensure each agent arrives at each task while avoiding midair collision in a multi-agent scenario, although only shown at the start of the simulation for agent-to-agent avoidance. In order to see the effectiveness of the proposed method, the two other methods from [26] and [30] were simulated by using the trajectory of quadrotor 1 since it would encounter an obstacle. The result of all 3 methods can be seen in Figure 12(a-b).



(a) (b)

Figure 12. Quadrotor (a) isometric and (b) top-down view for the simulation result with 3 different control systems

The search algorithm, with the help of Prioritized Communication Learning (PICO) [26] learning from, was shown to take the longest route since its reliance on a grid map. The PID-BF proposed in [30] and the proposed method has a similar trajectory, with a difference shown in how they avoid the obstacle similar to the trajectory tracking simulation. Figure 12(a-b) shows that the proposed method used a shorter route utilizing its 3-dimensional movement to avoid the obstacle, something that is not properly utilized by the PID-BF method. This problem does not come from the control system itself, but rather the disjointed force of the quadrotor dynamics. However, that is not the only problem introduced by the PID-BF. The route taken by the quadrotor shown that at the start the CBF portion of the control is only enough to slightly alter the trajectory and when the quadrotor come closer to the obstacle, the CBF push the quadrotor far from the obstacle, overpowering the PID control, making the quadrotor required to take a longer route to compensate. The effect of each method can be easily identified from the distance to the objective in Figure 13(a) and to the obstacle in Figure 13(b).



(a) (b)

Figure 13. Quadrotor distance to (a) objectives and (b) obstacles for 3 different control systems

Every method managed to maintain quadrotor safety while performing each assigned task, with the time needed for each method can be seen in Table 3. It presents the proposed method managed to be the fastest, with the PID-BF having a 3.03% increase in time, while the search algorithm required 56.06% more time when compared to the proposed method. As shown before in the trajectory tracking comparison, the proposed method and PID-BF method produce a similar outcome. When an obstacle was introduced, however, the PID-BF method lagged by 2 seconds compared to the proposed method. While negligible, in a more complex environment, this delay can increase proportionally to the number of obstacle they may encounter. As for the PICO method, due to its reliance on the grid-based map, the path produced was longer as the method cannot differentiate the length between a straight and diagonal path.

Table 3. Time comparison between backstepping CLBF, PID-BF, and search algorithm

Method	Time (s)
Backstepping CLBF	66
PID-Barrier Function	68
Search Algorithm	103

3.3 Multi-agent tracking and obstacle avoidance with power constraint

The method with consideration of quadrotor power consumption was tested. Using the multi-agent simulation and the battery and motor parameters as defined in the Simulation Parameter subsection. The new trajectory can be seen in Figure 14.

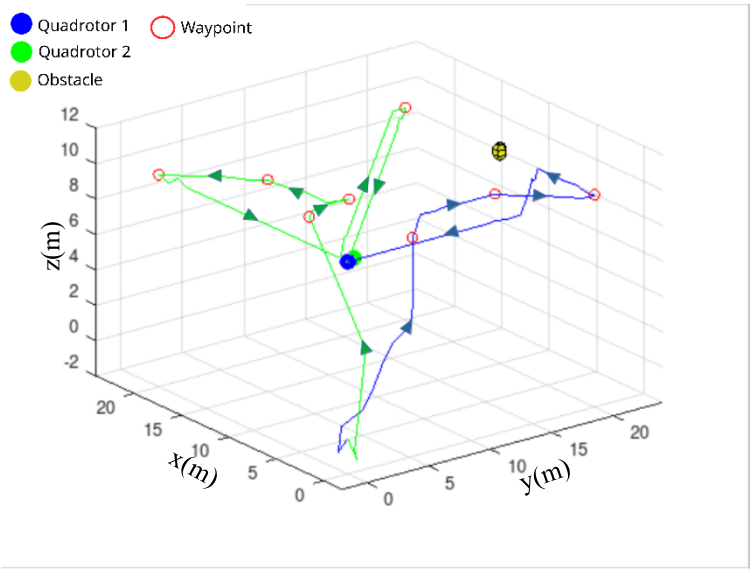


Figure 14. Quadrotor simulation result for waypoint following assignment with chaos-based consensus and collision avoidance for other agents and a static spherical obstacle at (20, 10, 10) under battery power constraint

The task assignment is still similar to the one in the multi-agent, yet the last waypoint for quadrotor 1 was taken by quadrotor 2 after it returned to the base. The overtaking of the task was overtaken because of quadrotor 1 remaining charge was not sufficient to go to the last waypoint and return to the base, and had to relinquish the last task for another quadrotor to take. This task was taken by the quadrotor 2, as this quadrotor managed to return to base and recharge its battery earlier than quadrotor 1, despite running out of charge at approximately the same time. Figure 15 shows the remaining charge for each quadrotor, including the time when each quadrotor had to return to the base due to insufficient charge without completely depleting the power storage. The quadrotor 1

managed to return to the base with 0.354 Ah or 17.7% remaining State of Charge (SoC) and the quadrotor 2 with 0.208 Ah or 10.4% remaining SoC. While in the simulation, the performance of the quadrotor won't be affected by the amount of charge, but in a real-life application a stricter condition must be applied to ensure that both quadrotors can safely return to the base. Without rebidding, the overall time to completion take 91 seconds an increase by 5% compared to 86 seconds needed for the method with rebidding.

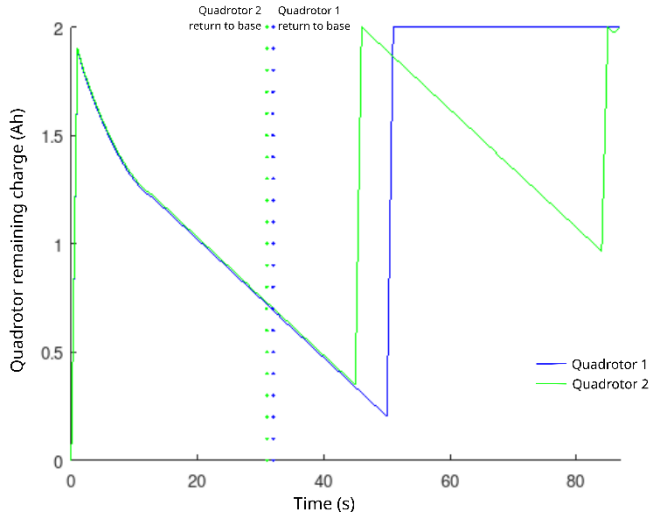


Figure 15. Quadrotor accumulated current draw during simulation with overridden return to base command

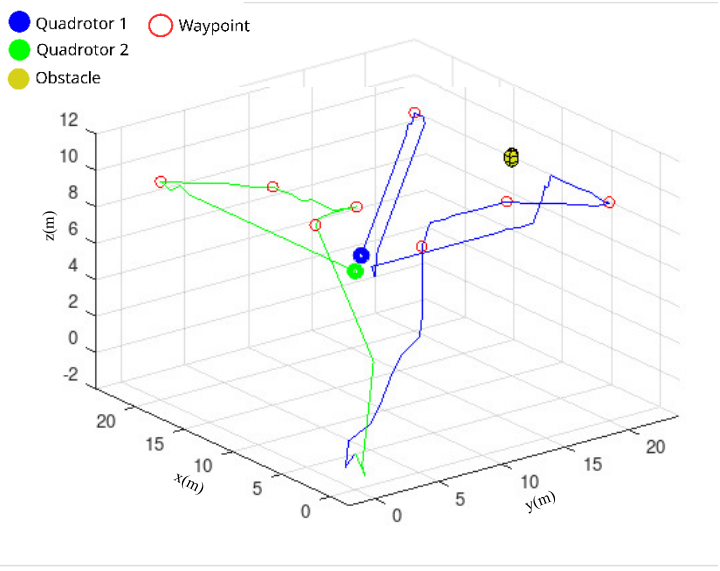


Figure 16. Quadrotor simulation result for waypoint following assignment with chaos-based consensus and collision avoidance for other agents and a static spherical obstacle at (20, 10, 10) under battery power constraint without rebidding

4. Conclusion

Quadrotors, typically perform monitoring or surveillance tasks in remote areas, face multiple constraints. In this paper, the constraint was considered due to limited charging and communication stations, alongside the possible collision between the drone and the environment. The low-fidelity simulation shows that the drone can follow its objective using backstepping control Lyapunov barrier function without going inside the unsafe state. Compared to other methods, the proposed

method managed to finish its task 3% faster compared to the PID-BF and 56% faster compared to the PICO method. As for the chaos-bidding consensus, all drones manage to reach a consensus as to where they go without the help of a central control system. When the battery constraint was introduced, the proposed method managed to return the drone before the battery charge was depleted and rebid the task to other quadrotors that were available to decrease overall time to completion for the entire system. This paper considered the communication constraint only in terms of drone-to-central-control communication, while the drone-to-drone communication limitation was not considered. Alongside the movement from obstacle avoidance, future work is advised to consider the limitations of drone-to-drone communication while smoothing out the movement for the avoidance.

Author's declaration

Author contribution

Muhammad Zakiyullah Romdlony: Conceptualization, Methodology, Validation. **Rashad Abul Khayr**: Software, Investigation, Writing - Original Draft. **Yul Yunazwin Nazaruddin**: Supervision, Funding acquisition. **Tua Agustinus Tamba**: Data Curation, Formal analysis. **Md. Abdus Samad Kamal**: Writing - Review & Editing.

Funding statement

This research is funded by the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia under the KATALIS project with contract number 059/E5/PG.02.00/PL.BATCH.2/2024, 016/SP2H/PL.BATCH.2/LL4/2024, and 152/LIT07/PPM-LIT/2024.

Data Availability

Data generated from the simulation can be requested from the corresponding author.

Acknowledgements

The authors thank Telkom University for providing facilities during this research.

Competing interest

The authors declare that they are NOT affiliated with or involved in any organization or entity that has a financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consulting, stock ownership, or other equity interests; and expert testimony or patent licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge, or beliefs) in the subject matter or materials discussed in this manuscript.

Ethical clearance

This research does not involve humans as subjects.

AI statement

The language use in this article has been validated and verified by an English language expert, and none of the AI-generated sentences is included in this article.

Publisher's and Journal's note

Universitas Negeri Padang as the publisher, and Editor of Teknomekanik state that there is no conflict of interest towards this article publication.

References

[1] J. Pasha *et al.*, "The Drone Scheduling Problem: A Systematic State-of-the-Art Review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14224–14247, Sep. 2022, https://doi.org/10.1109/TITS.2022.3155072

[2] K. Telli *et al.*, "A Comprehensive Review of Recent Research Trends on Unmanned Aerial Vehicles (UAVs)," *Systems*, vol. 11, no. 8, p. 400, Aug. 2023, https://doi.org/10.3390/systems11080400

[3] S. A. H. Mohsan, N. Q. H. Othman, Y. Li, M. H. Alsharif, and M. A. Khan, "Unmanned aerial vehicles (UAVs): practical aspects, applications, open challenges, security issues, and future trends," *Intell Serv Robot*, vol. 16, no. 1, pp. 109–137, Mar. 2023, https://doi.org/10.1007/s11370-022-00452-4

[4] N. Elmeseiry, N. Alshaer, and T. Ismail, "A Detailed Survey and Future Directions of Unmanned Aerial Vehicles (UAVs) with Potential Applications," *Aerospace*, vol. 8, no. 12, p. 363, Nov. 2021, https://doi.org/10.3390/aerospace8120363

[5] P. K. Chittoor, B. Chokkalingam, and L. Mihet-Popa, "A Review on UAV Wireless Charging: Fundamentals, Applications, Charging Techniques and Standards," *IEEE Access*, vol. 9, pp. 69235–69266, 2021, https://doi.org/10.1109/ACCESS.2021.3077041

[6] Y. Qin, M. A. Kishk, and M.-S. Alouini, "Drone Charging Stations Deployment in Rural Areas for Better Wireless Coverage: Challenges and Solutions," *IEEE Internet of Things Magazine*, vol. 5, no. 1, pp. 148–153, Mar. 2022, https://doi.org/10.1109/IOTM.001.2100083

[7] Y. Xiao *et al.*, "Fully Decentralized Federated Learning-Based On-Board Mission for UAV Swarm System," *IEEE Communications Letters*, vol. 25, no. 10, pp. 3296–3300, Oct. 2021, https://doi.org/10.1109/LCOMM.2021.3095362

[8] Z. Cheng, L. Zhao, and Z. Shi, "Decentralized Multi-UAV Path Planning Based on Two-Layer Coordinative Framework for Formation Rendezvous," *IEEE Access*, vol. 10, pp. 45695–45708, 2022, https://doi.org/10.1109/ACCESS.2022.3170583

[9] V. Viswanathan *et al.*, "The challenges and opportunities of battery-powered flight," *Nature*, vol. 601, no. 7894, pp. 519–525, Jan. 2022, https://doi.org/10.1038/s41586-021-04139-1

[10] S. Sasmono *et al.*, "The Resilience of the Grid from the Risk of Failure Due to Kite Thread Disturbance: Case Study Khatulistiwa Grid," in *2022 IEEE PES Innovative Smart Grid Technologies - Asia (ISGT Asia)*, IEEE, Nov. 2022, pp. 120–124. https://doi.org/10.1109/ISGTAsia54193.2022.10003494

[11] I. Høiaas, K. Grujic, A. G. Imenes, I. Burud, E. Olsen, and N. Belbachir, "Inspection and condition monitoring of large-scale photovoltaic power plants: A review of imaging technologies," *Renewable and Sustainable Energy Reviews*, vol. 161, p. 112353, Jun. 2022, https://doi.org/10.1016/j.rser.2022.112353

[12] T. A. Tamba, B. C. G. Cinun, Y. Y. Nazaruddin, M. Z. Romdlony, and B. Hu, "Event-triggered robust formation control of multi quadrotors for transmission line inspection," *Journal of Mechatronics, Electrical Power, and Vehicular Technology*, vol. 15, no. 2, pp. 197–207, Dec. 2024, https://doi.org/10.55981/j.mev.2024.1106

[13] C. Zhao, J. Liu, M. Sheng, W. Teng, Y. Zheng, and J. Li, "Multi-UAV Trajectory Planning for Energy-Efficient Content Coverage: A Decentralized Learning-Based Approach," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 10, pp. 3193–3207, Oct. 2021, https://doi.org/10.1109/JSAC.2021.3088669

[14] B. Dai, J. Niu, T. Ren, Z. Hu, and M. Atiquzzaman, "Towards Energy-Efficient Scheduling of UAV and Base Station Hybrid Enabled Mobile Edge Computing," *IEEE Trans Veh Technol*, vol. 71, no. 1, pp. 915–930, Jan. 2022, https://doi.org/10.1109/TVT.2021.3129214

[15] R. Lai *et al.*, "Energy-Efficient Scheduling in UAV-Assisted Hierarchical Wireless Sensor Networks," *IEEE Internet Things J*, vol. 11, no. 11, pp. 20194–20206, Jun. 2024, https://doi.org/10.1109/JIOT.2024.3369722

[16] R. Ma, J. Song, Y. Zhang, H. Zhang, and M. Yuan, "Lifetime-Optimized Energy Management Strategy for Fuel Cell Unmanned Aircraft Vehicle Hybrid Power System," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 9, pp. 9046–9056, Sep. 2023, https://doi.org/10.1109/TIE.2022.3206687

[17] W. Wang, N. Qi, L. Jia, C. Li, T. A. Tsiftsis, and M. Wang, "Energy-Efficient UAV-Relaying 5G/6G Spectrum Sharing Networks: Interference Coordination with Power Management and Trajectory Design," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1672–1687, 2022, https://doi.org/10.1109/OJCOMS.2022.3209368

[18] X. Yuan, Y. Hu, J. Zhang, and A. Schmeink, "Joint User Scheduling and UAV Trajectory Design on Completion Time Minimization for UAV-Aided Data Collection," *IEEE Trans Wirel Commun*, vol. 22, no. 6, pp. 3884–3898, Jun. 2023, https://doi.org/10.1109/TWC.2022.3222067

[19] A. Banerjee, S. K. Gupta, P. Gupta, A. Sufian, A. Srivastava, and M. Kumar, "UAV-IoT collaboration: Energy and time-saving task scheduling scheme," *International Journal of Communication Systems*, vol. 36, no. 14, Sep. 2023, https://doi.org/10.1002/dac.5555

[20] D. Liu, X. Zhu, W. Bao, B. Fei, and J. Wu, "SMART: Vision-Based Method of Cooperative Surveillance and Tracking by Multiple UAVs in the Urban Environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24941–24956, Dec. 2022, https://doi.org/10.1109/TITS.2022.3203411

[21] A. Khan, S. Gupta, and S. K. Gupta, "Emerging UAV technology for disaster detection, mitigation, response, and preparedness," *J Field Robot*, vol. 39, no. 6, pp. 905–955, Sep. 2022, https://doi.org/10.1002/rob.22075

[22] Z. Yang *et al.*, "UAV remote sensing applications in marine monitoring: Knowledge visualization and review," *Science of The Total Environment*, vol. 838, p. 155939, Sep. 2022, https://doi.org/10.1016/j.scitotenv.2022.155939

[23] G. Li, H. Huang, S. Song, and B. Liu, "A nonlinear control scheme based on input–output linearized method achieving PFC and robust constant voltage output for boost converters," *Energy Reports*, vol. 7, pp. 5386–5393, Nov. 2021, https://doi.org/10.1016/j.egyr.2021.08.169

[24] U. Masood, M. K. Azeem, I. Ahmad, and A. ul Jabbar, "Robust adaptive nonlinear control of plugin hybrid electric vehicles for vehicle to grid and grid to vehicle power flow with hybrid energy storage system," *ISA Trans*, vol. 139, pp. 406–424, Aug. 2023, https://doi.org/10.1016/j.isatra.2023.03.035

[25] C. Jammazi, M. Boutayeb, and K. Saidi, "On the fixed-time extinction based nonlinear control and systems decomposition: applications to bilinear systems," *Chaos Solitons Fractals*, vol. 174, p. 113893, Sep. 2023, https://doi.org/10.1016/j.chaos.2023.113893

[26] W. Li, H. Chen, B. Jin, W. Tan, H. Zha, and X. Wang, "Multi-Agent Path Finding with Prioritized Communication Learning," in *2022 International Conference on Robotics and Automation (ICRA)*, IEEE, May 2022, pp. 10695–10701. https://doi.org/10.1109/ICRA46639.2022.9811643

[27] M. Kosaka and H. Nakamura, "Backstepping CBF Design for Collision Avoidance of Electric Wheelchair," *IFAC-PapersOnLine*, vol. 56, no. 1, pp. 331–336, 2023, https://doi.org/10.1016/j.ifacol.2023.02.056

[28] C. Ha, Z. Zuo, F. B. Choi, and D. Lee, "Passivity-based adaptive backstepping control of quadrotor-type UAVs," *Rob Auton Syst*, vol. 62, no. 9, pp. 1305–1315, Sep. 2014, https://doi.org/10.1016/j.robot.2014.03.019

[29] X. Zheng, X. Yang, H. Zhao, and Y. Chen, "Saturated Adaptive-Law-Based Backstepping and Its Applications to a Quadrotor Hover," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 12, pp. 13473–13482, Dec. 2022, https://doi.org/10.1109/TIE.2021.3139235

[30] M. Li, Z. Sun, and S. Weiland, "Quadrotor Stabilization with Safety Guarantees: A Universal Formula Approach," *ArXiv*, 2024.

[31] M. Z. Romdlony and B. Jayawardhana, "Stabilization with guaranteed safety using Control Lyapunov–Barrier Function," *Automatica*, vol. 66, pp. 39–47, Apr. 2016, https://doi.org/10.1016/j.automatica.2015.12.011

[32] A. M.A. and A. Saleem, "Quadrotor Modeling Approaches and Trajectory Tracking Control Algorithms: A Review," *International Journal of Robotics and Control Systems*, vol. 4, no. 1, pp. 401–426, Apr. 2024, https://doi.org/10.31763/ijrcs.v4i1.1324

[33] Aleksander Nawrat and Karol Jędrasiak, *Innovative Simulation Systems*. Springer, 2016.

[34] E. D. Sontag, "A 'universal' construction of Artstein's theorem on nonlinear stabilization," *Syst Control Lett*, vol. 13, no. 2, pp. 117–123, Aug. 1989, https://doi.org/10.1016/0167-6911(89)90028-5

[35] P. Wieland and F. Allgöwer, "Constructive Safety Using Control Barrier Functions," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 462–467, 2007, https://doi.org/10.3182/20070822-3-ZA-2920.00076

[36] C. Sarkar, S. Dey, and M. Agarwal, "Semantic knowledge driven utility calculation towards efficient multi-robot task allocation," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, IEEE, Aug. 2018, pp. 144–147. https://doi.org/10.1109/COASE.2018.8560555

[37] O. Landsiedel, F. Ferrari, and M. Zimmerling, "Chaos," in *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, New York, NY, USA: ACM, Nov. 2013, pp. 1–14. https://doi.org/10.1145/2517351.2517358

[38] M. Z. Romdlony, R. A. Khayr, P. Pangaribuan, Y. Y. Nazaruddin, and T. A. Tamba, "Drone Decentralized Scheduling for Rural Area Monitoring ," in *5 th International Conference on Electronics, Biomedical Engineering, and Health Informatics*, 2024.

[39] P. Mahato, S. Saha, C. Sarkar, and Md. Shaghil, "Consensus-based fast and energy-efficient multi-robot task allocation," *Rob Auton Syst*, vol. 159, p. 104270, Jan. 2023, https://doi.org/10.1016/j.robot.2022.104270

[40] L. Zhou, J. Zhang, H. She, and H. Jin, "Quadrotor UAV flight control via a novel saturation integral backstepping controller," *Automatika*, vol. 60, no. 2, pp. 193–206, Apr. 2019, https://doi.org/10.1080/00051144.2019.1610838

Nomenclature

ϕ	Quadrotor roll
θ	Quadrotor pitch
ψ	Quadrotor yaw
$\dot{\phi}$	Quadrotor roll speed
$\dot{\theta}$	Quadrotor pitch speed
$\dot{\psi}$	Quadrotor yaw speed
$\ddot{\phi}$	Quadrotor roll acceleration
$\ddot{\theta}$	Quadrotor pitch acceleration
$\ddot{\psi}$	Quadrotor yaw acceleration
x_a	Quadrotor local position in x axis
y_a	Quadrotor local position in y axis
z_a	Quadrotor local position in z axis
x	Quadrotor position in x axis relatives to the earth

y	Quadrotor position in y axis relatives to the earth
z	Quadrotor position in z axis relatives to the earth
$\ddot{x}$	Quadrotor acceleration in x axis relatives to the earth
$\ddot{y}$	Quadrotor acceleration in y axis relatives to the earth
$\ddot{z}$	Quadrotor acceleration in z axis relatives to the earth
Ω_i	Quadrotor motor i rotational speed
b	Thrust factor
l	Distance from the center of the quadrotor to the center of the motor
d	Drag factor
U_1	Total force and is responsible for all translational movement of the quadrotor
U_2	Total force and is responsible for roll movement of the quadrotor
U_3	Total force and is responsible for pitch movement of the quadrotor
U_4	Total force and is responsible for yaw movement of the quadrotor
k	Quadrotor generalized position/rotation
k_1	Quadrotor generalized acceleration
k_2	Quadrotor generalized speed
I_i	Quadrotor inertia in i axis
g	Gravitational acceleration
m	Quadrotor mass
$D_{i-1,i}$	Distance between a previously chosen node to the next
D_{Qi}	Distance between quadrotor i and waypoint
D_{iminj}	The smallest distance for quadrotor i to waypoint j
T_i	bid value on waypoint j