AUT Journal of Electrical Engineering
10.22060/EEJ.2026.25522.5949

Integrating Real Secret Sharing with Distributed MPC for Confidential and Collision-Resilient Multi Quadrotors Formations

Abolfazl Mokhtari [1]

[1] Associate Professor Faculty of Engineering and Flight, Imam Ali University, Tehran

Abstract:

This paper presents a novel framework that integrates secure, distributed computation with advanced control to enable confidential and collision-resilient formation flight for multiple quadrotors. Conventional multi-agent systems are vulnerable to cyber threats such as eavesdropping and data manipulation, which can destabilize formations and compromise missions. In this study, to address this issue, Model Predictive Control is combined with a multi-party computation protocol based on secret sharing. In this approach, each quadrotor converts its state data into secret shares and distributes them among the agents, and the required control computations are performed collaboratively on the encrypted data without revealing any sensitive information. This decentralized approach eliminates the central controller as a single point of failure and attack. Simulation results for six quadrotors flying in a circular formation among obstacles show that the system maintains formation accuracy and collision avoidance while remaining robust against eavesdropping and data manipulation. Quantitative evaluation shows that the formation tracking error remains below 0.08 m, all inter-agent and obstacle collisions are successfully avoided (100% success rate), and the secret-sharing protocol reconstructs the optimal control signals with a normalized mean squared error of less than 10^{-6}, ensuring cryptographic integrity without compromising formation performance. The integration fundamentally enhances operational security, preserves agent privacy, and improves system fault tolerance and scalability for real-world adversarial environments.

Keywords:

Unmanned Aerial Vehicles (UAVs), Formation Control, Model Predictive Control (MPC), Multi-Party Computation, Real Secret Sharing, Cybersecurity, Decentralized Control, Obstacle Avoidance.

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

AUT Journal of Electrical Engineering
10.22060/EEJ.2026.25522.5949

1. Introduction

Unmanned aerial vehicles (UAVs), particularly quadrotors, have rapidly evolved from mere experimental platforms into indispensable tools in a variety of fields, including surveillance, disaster management, environmental monitoring, and parcel delivery [1-3]. Their ability to hover, move with agility in constrained spaces, and capture high-resolution data has catalyzed a surge of both academic interest and commercial investment. In many scenarios, real-time coordination and cooperation among multiple UAVs become central to achieving mission objectives efficiently. For instance, a fleet of quadrotors can collaboratively scan large areas for signs of forest fires, or coordinate supply drops in remote terrains when a single UAV would be insufficient or prone to failure [4]. One critical aspect of UAV deployment in such collaborative contexts is the need for precise and reliable coordination among multiple vehicles, often in potentially unpredictable environments.

Formation control, which enables a group of UAVs to maintain a prescribed geometric pattern such as a line, square, or circle while route to a common destination, has been a focal point of research over the past decade [5, 6]. Classical control strategies, including PID controllers and Linear Quadratic Regulators (LQR), initially demonstrated proof-of-concept implementations for formation flight but also exposed inherent limitations in handling nonlinear flight dynamics, delays in inter-vehicle communications, and unpredictable disturbances like wind gusts [7, 8]. Moreover, as the scale and complexity of UAV missions increased, these linear methods struggled to handle the high-dimensionality of multi-UAV systems and their intricate coupling constraints. Hence, researchers have turned to more advanced control techniques such as adaptive control, sliding mode control, and, most notably, Model Predictive Control (MPC) [9-11].

While sliding mode control (SMC) provides excellent robustness against model uncertainties and external disturbances, and H∞ control offers strong disturbance attenuation in the presence of worst-case perturbations, these methods typically handle constraints (e.g., actuator limits, minimum separation

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

distances) in a less direct manner, often requiring additional schemes such as reference governors or barrier functions. In contrast, Model Predictive Control (MPC) inherently incorporates constraints as part of the optimization problem, making it uniquely suitable for multi-UAV formation flight where collision avoidance and actuator saturation are critical. Moreover, MPC's receding-horizon nature enables predictive collision-free path planning in cluttered environments, a capability that is not naturally provided by SMC or H∞ control. Several recent studies have compared these paradigms for nonlinear systems [12]. For quadrotor applications, SMC has been successfully applied to attitude control [13] and H_∞ control to disturbance rejection [14], but formation coordination with obstacle avoidance remains largely dominated by MPC due to its constraint-handling and predictive features. This paper therefore adopts MPC as the core control layer, while using cryptographic techniques to secure the distributed computations.

Model Predictive Control (MPC) is highly effective for multi-UAV formation, enabling constrained optimization and predictive, collision-free path planning in complex environments [15-19]. However, the reliance on wireless communication exposes these systems to security threats such as eavesdropping and data manipulation, which can destabilize the formation or compromise mission data [20-23]. This underscores the critical need for integrating robust cryptographic protection within decentralized UAV control architectures. As highlighted by Hassan et al. [24], the rising economic and operational costs of cyberattacks across industries further motivate the integration of such cryptographic safeguards into multi-UAV systems. cryptographic safeguards into multi-UAV systems. In parallel, generative adversarial networks (GANs) have recently shown promise for dynamic threat detection and mitigation in cyber-physical environments [25].

In parallel, the concept of Multi-Party Computation (MPC) has gained traction in distributed systems, enabling multiple parties to compute a function over their inputs while keeping those inputs private from one another [26]. The synergy between multi-UAV networks and Multi-Party Computation is compelling, especially when considering formation control and obstacle avoidance, which require frequent data

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

sharing—such as position updates, velocity information, or sensor readings. In this context, optimization of the underlying communication links is equally critical; for instance, Harinitha et al. [27] proposed an enhanced optimization strategy to maximize the achievable rate of millimeter-wave full-duplex UAV communications for multiple users, which complements our secure computation layer by ensuring high-rate, low-latency data exchange among quadrotors. Traditional centralized approaches to UAV coordination not only create a single point of failure but also raise privacy and confidentiality concerns, as sensitive data must be aggregated in one location. By contrast, a distributed approach leveraging MPC ensures that essential computations occur collaboratively across nodes without exposing raw data to any single entity [28].

However, most conventional Multi-Party Computation frameworks and Secret Sharing schemes (e.g., Shamir's Secret Sharing) were designed for discrete data, limiting their direct applicability to dynamic systems like quadrotor control [29-31]. Real Secret Sharing protocols bridge this gap by enabling operations on real-valued numbers [32]. In the proposed scheme, each quadrotor shares its local data (such as position and velocity) not in raw form, but as encrypted shares that are individually meaningless [33]. This approach eliminates reliance on a centralized controller which could be a prime target for cyber-attacks [34, 35] and by distributing the computations, it simultaneously enhances security and improves fault tolerance through inherent redundancy [36].

Within this context, the present work aims to bridge the gap between advanced formation control techniques and state-of-the-art cryptographic protocols. This paper proposes a comprehensive framework that integrates Real Secret Sharing, Multi-Party Computation, and Model Predictive Control to facilitate secure and efficient formation maneuvers for multiple quadrotors. Specifically, this paper considers six quadrotors arranged in a circular formation, traveling from a starting point (location A) to a designated endpoint (location B). Along the flight path, three stationary obstacles demand adaptive collision-avoidance maneuvers, which This paper implement using MPC’s predictive capabilities. At the same time, the Real

Corresponding author’s email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

Secret Sharing and Multi-Party Computation layers ensure that control calculations such as the optimization of control inputs are distributed among the quadrotors themselves, thereby mitigating the risks associated with a single point of failure or a centralized aggregator [37].

the main novel contributions of this paper are as follows: first, the first integration of Real Secret Sharing with distributed MPC for quadrotor formations, where unlike prior works that treat control and cryptography separately, our framework enables collaborative control computation directly on encrypted secret shares without ever reconstructing raw sensitive data (e.g., position, velocity); second, elimination of the central controller as a single point of failure and attack, as by distributing both data and computation across the fleet, the proposed architecture removes reliance on any central aggregator, significantly enhancing resilience against cyber-attacks and system faults; third, preservation of individual quadrotor privacy, meaning each quadrotor's local state remains hidden from other agents and external eavesdroppers throughout the entire formation control and obstacle avoidance process, preserving operational privacy and autonomy; fourth, robust defense against both passive and active attacks, where the Real Secret Sharing mechanism ensures that even with up to t compromised shares, no information leaks and the system maintains correct operation under tampering attempts; and fifth, demonstration of collision-resilient secure formation flight, as simulation results for a six-quadrotor circular formation navigating through an obstacle field validate the dual efficacy of precise formation control, effective collision avoidance, and cryptographic security in a realistic adversarial scenario [38].

Beyond these security-centric advantages, our proposed framework offers practical benefits in terms of scalability and fault tolerance. Distributing computations across multiple quadrotors helps avoid computational bottlenecks and renders the system more resilient to single-point failures [39]. For instance, if one quadrotor temporarily loses communication or experiences a hardware malfunction, the secret-sharing mechanism and distributed MPC can dynamically adjust, ensuring the formation can continue its mission with minimal performance degradation. Furthermore, secret-sharing methods can adapt to nodes

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

that join or leave the network, which proves beneficial in missions where UAVs may be introduced at different stages or must exit once their tasks are complete. This adaptability paves the way for highly flexible UAV swarms capable of autonomously adjusting to changing mission parameters or unanticipated disruptions [40].

To validate our integrated approach, I conduct extensive simulation studies, focusing on the system's ability to maintain a stable circular formation, execute effective obstacle avoidance, and exhibit robust performance under simulated adversarial conditions. his paper evaluates various threat models, including eavesdropping attacks that attempt to intercept secret shares, as well as active tampering attacks that seek to corrupt the underlying computations. Our results indicate that the combination of Real Secret Sharing, Multi-Party Computation, and MPC not only upholds mission objectives but also provides a robust defense against common security threats.

This article is structured as follows. Section II reviews the theoretical foundations of formation control, and Section III covers the fundamentals of the cryptographic techniques used in the proposed framework. Section IV presents the system architecture, which integrates a distributed MPC scheme with Real Secret Sharing. Section V is devoted to the experimental evaluation of the simulation results, and Section VI concludes by summarizing the findings, stating the limitations, and suggesting potential avenues for future research.

2. Theoretical or experimental modeling

A quadrotor is a rotorcraft with four propellers, arranged in a cross configuration, providing thrust for vertical takeoff, landing, and maneuvers. Let us denote by $\boldsymbol{P(t) = \left[x(t), y(t), z(t)\right]^T}$ the position of the quadrotor's center of mass in the inertial (world) frame. This paper uses a Z-X-Y Euler angle convention (or any other standard convention) to define the orientation of the quadrotor. Let

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

$$\eta(t) = \left[\varphi(t), \theta(t), \psi(t)\right]^T \tag{1}$$

represent the roll (φ), pitch (θ), and yaw (ψ) angles, respectively. Under the simplifying assumption that the airframe is rigid and aerodynamic drag is either negligible or modeled as a simple damping term, the translational dynamics of the quadrotor can be written as:

$$m\ddot{P}(t) = R(\varphi(t), \theta(t), \psi(t))T(t) - mge_Z \tag{2}$$

Where m is the mass of the quadrotor, $R(\varphi(t),\theta(t),\psi(t))$ is the rotation matrix transforming forces from the body frame to the inertial frame. $T(t)$ is the total thrust vector generated by the four rotors in the body frame (typically along the body z-axis), g is the gravitational acceleration constant, and $e_Z = [0,0,1]^T$ is the unit vector in the vertical direction of the inertial frame. Often, if the quadrotor is assumed to be symmetrical and the thrust is aligned with the body's z-axis, then $e_Z = [0,0,T]^T$, where T is the magnitude of the thrust. Let $\omega = [p,r,q]^T$ be the angular velocity of the quadrotor in its body frame. The rotational dynamics can be expressed (in vector form) as:

$$I\dot{\omega}(t) + \omega(t) \times (I\omega(t)) = \tau(t) \tag{3}$$

Where I is the moment of inertia matrix (diagonal if the body is assumed symmetric: $I = \text{diag}(I_X, I_y, I_Z)$ and $\tau(t)$ is the net torque vector due to the thrust differences of the rotors (roll, pitch, and yaw torques).

The Euler angles ϕ, θ, ψ relate to $\omega = [p,r,q]^T$ through a kinematic relationship (dependent on the chosen Euler angle convention), for example:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)\sec(\theta) & \cos(\phi)\sec(\theta) \end{bmatrix} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{4}$$

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

While the exact matrix might differ slightly based on convention, the above captures the idea that (p, r, q) combine to produce changes in (ϕ, θ, ψ). Each rotor's angular velocity, ω_i produces both thrust and reaction torques. A common approach is to define four aggregate control inputs, $u_1 = T = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)$, $u_2 = \tau_\phi$, $u_3 = \tau_\theta$, and $u_4 = \tau_\psi$. where b is the thrust coefficient, and τ_ϕ, τ_θ, and τ_ψ represent net roll, pitch, and yaw torques respectively. Hence, the control problem typically becomes finding u_1, u_2, u_3, and u_4 to make the quadrotor follows a desired trajectory or maintain a desired formation.

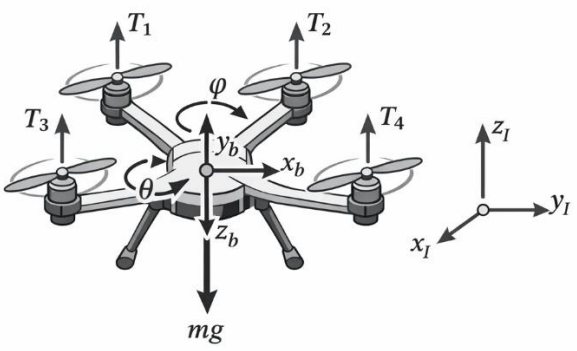

Fig. 1. Schematic of the quadrotor UAV showing the inertial frame [14], body frame [14], rotor thrusts $T_1 - T_4$, and roll (ϕ), pitch (θ), and yaw (ψ) axes.

2.1 Circular Formation Setup

The purpose of this subsection is to define the desired geometric formation a horizontal circle of fixed radius that the team of six quadrotors must maintain while moving from a start point A to an end point B. This formation geometry directly provides the reference trajectories and inter-agent constraints used in the MPC cost function (Section 3).

Uncorrected Proof

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

Consider a team of $N = 6$ quadrotors labeled $i = 1,2,...,6$. Let $c(t) \in R^2$ be the time-varying center of the circle in the horizontal plane. All quadrotors fly at a fixed altitude $z = z_0$ (extensions to 3D are straightforward). The desired position of the i^{th} quadrotor at time t is given by:

$$P_i^{dec}(t) = \left[x_C(t), y_C(t), z_O \right]^T + R\left[\cos(\theta_i(t)), \sin(\theta_i(t)), z_O \right]^T \quad (5)$$

where θ_i is the angular offset for the i^{th} quadrotor around the circle. Often, one can fix an initial phase offset so that $\theta_i(0) = \theta_0 + \frac{2\pi}{n}(i-1)$ where $i = 1,2,...,N$. As $c(t)$ changes over time, the entire circle shifts its position, while the individual offsets $\theta_i(t)$ determine each quadrotor's relative location around the circle. The fundamental requirement is to keep the distance between any quadrotor i and the center $c(t)$ close to R, within a small tolerance ε . Mathematically:

$$\left\| P_i(t) - c(t) \right\| \approx R, \forall i = 1,2,...,N \quad (6)$$

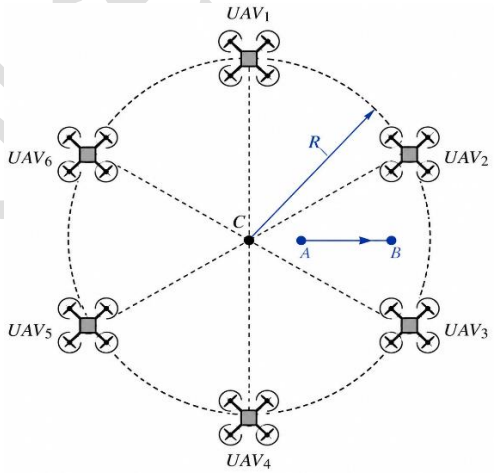

Fig 2. Topological diagram of a circular formation with six quadrotor UAVs equally spaced on a circle of radius R around the formation center C , moving from point *A* to point *B* while maintaining a safe inter-vehicle distance.

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Unc rrected Proof

10.22060/EEJ.2026.25522.5949

Additionally, collision avoidance within the formation is typically enforced via constraints such that each pair of quadrotors (i, j) maintains a minimum separation distance $d_{\min}$ That is $\left\| P_i(t) - P_j(t) \right\| \approx d_{\min}, \forall i \neq j$. Since the system is dynamic, these spatial constraints become part of the predictive control and trajectory design problem. To move from A to B, I can define a desired trajectory $c^{\text{dec}}(t)$ for the center of the circle. Simultaneously, one can hold each $\theta_i(t)$ constant maintaining a static spacing around the circle or allow small adjustments if the group must reconfigure to avoid obstacles. If I assume the radius R remains fixed, then I only need to plan $c^{\text{dec}}(t)$ and manage $\theta_i(t)$ in real time to ensure safe, collision-free transitions.

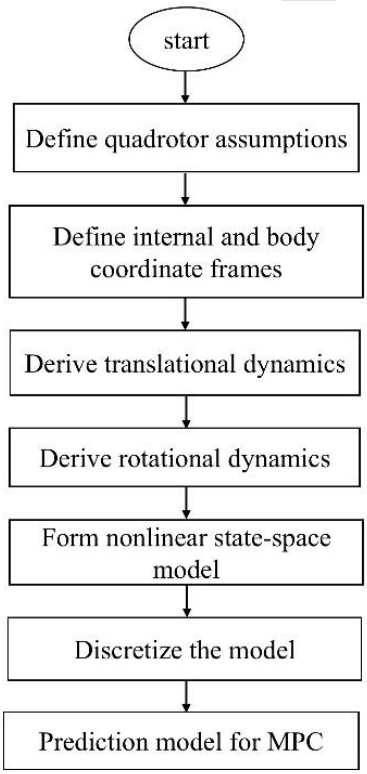

Fig. 3. Flowchart of the quadrotor modeling and circular formation setup procedure.

3. Control Scheme Based on Model Predictive Control

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

This section details the Model Predictive Control (MPC) framework for each quadrotor in a formation. MPC is favored for its ability to handle complex dynamics, incorporate constraints, and optimize control in real-time. The approach is formalized with the state-space model, objective function, and constraints for safe, coordinated flight.

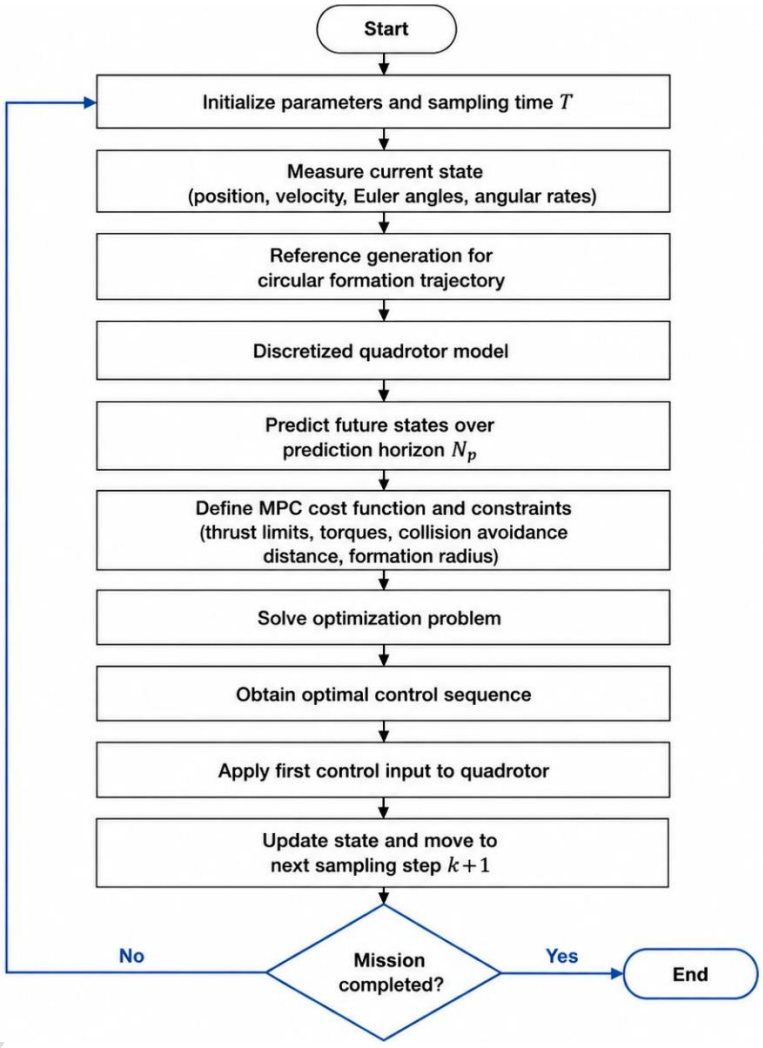

Fig 4. Flowchart of the proposed MPC-based control algorithm for the quadrotor UAV.

For Model Predictive Control (MPC), a quadrotor's continuous-time nonlinear dynamics are often discretized into a discrete-time model. The state vector at step k is $x_k = [P_k, \dot{P}_k, \eta_k, \dot{\eta}_k]^T$, representing

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

position, velocity, Euler angles, and angular rates. The control input $u_k = [u_{1,k}, u_{2,k}, u_{3,k}, u_{4,k}]^T$ corresponds to total thrust, roll torque, pitch torque, and yaw torque. Starting from the continuous model $\dot{x}(t) = f(x(t), u(t))$, a discrete-time approximation is derived over a sampling period Δt using methods like Zero-Order Hold (ZOH) or Runge-Kutta. This yields the nonlinear discrete model $x_{k+1} = F(x_k, u_k)$. which captures the system's behavior for MPC implementation.

Model Predictive Control (MPC) computes an optimal sequence of future control inputs over a finite prediction horizon (N_p) by solving an online optimization problem, subject to system constraints. Only the first control action (u_k) from this sequence is applied to the system. At the next time step, the horizon recedes, and the entire process is repeated with new state measurements, utilizing a control horizon $N_c \leq N_p$ where inputs beyond Nc are typically held constant.

The MPC cost function typically includes terms that penalize deviation from desired trajectories or formation references, as well as terms penalizing large control inputs or control rate changes. For each quadrotor $i \in \{1,2,...,N\}$, define, x_k^i the state of the i^{th} quadrotor at time k, u_k^i the control input for the i^{th} quadrotor at time k, $x_k^{i,ref}$ the reference trajectory for the i^{th} quadrotor's state (or some subset thereof, e.g., only position),and $u_k^{i,ref}$ the nominal or reference input (often set to hover thrust and zero torque). A general form of the stage cost at time step $k+j$ can be written as:

$$l(x_{k+j}^i, u_{k+j}^i) = \left\| W_x (x_{k+j}^i - x_{k+j}^{i,ref}) \right\| + \left\| W_u (u_{k+j}^i - u_{k+j}^{i,ref}) \right\| \quad (7)$$

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

where W_x and W_u are weighting matrices that adjust the importance of state tracking vs. control effort. Over the prediction horizon N_p, the total cost for quadrotor i is typically a sum of stage costs plus a terminal cost term $l_f(x^i_{k+N_p})$ to reflect end-of-horizon objective:

$$J^i_k = \sum_{j=1}^{N_p-1} l(x^i_{k+j}, u^i_{k+j}) + l_f(x^i_{k+N_p}) \quad (8)$$

If each quadrotor is optimized individually (potentially under distributed or decentralized schemes), I sum or combine the costs for all UAVs. In a fully centralized scheme, the cost might look like, $J_k = \sum_{i=1}^{N} J^i_k$.

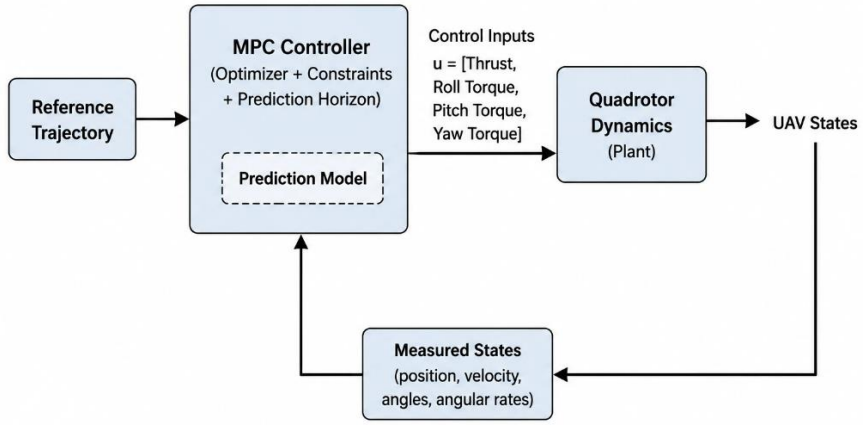

Fig 5. Block diagram of the proposed MPC-based control scheme for the quadrotor UAV.

3.1 Incorporating Formation Error in the Cost

For circle formation, each quadrotor i has a desired position $P^{i,des}_k$. A portion of the state tracking error specifically measures the deviation from this position:

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

$$l_{\text{formation}}(P_k^i) = \left\| W_P (P_k^i - P_k^{\text{i,ref}}) \right\|^2 \quad (9)$$

Where W_P is a diagonal matrix of positional weights. This term can be merged into the overall cost function to reinforce adherence to the formation geometry. MPC integrates constraints at each time step to ensure the solutions remain feasible and safe. The evolution of x_k^i is constrained by the discrete-time dynamics that we can show it as $x_{k+1}^i = F(x_k^i, u_k^i),\ \ k = 0,1,2,...,N_p - 1$.

If there are M known obstacles, each represented by a region Θ_m, I impose $P_k^i \notin \Theta_m,\ m = 0,1,2,...,M$. In practice, these constraints can be encoded as minimum distance constraints, $\left\| P_k^i - \mathrm{O}_m \right\| \geq r_m$, where O_m and r_m denote the center and radius of obstacle m, respectively. For any pair (i, j) with $i \neq j$, ensure $\left\| P_k^i - P_k^j \right\| \geq d_{\min}$ This constraint prevents collisions within the formation. A terminal constraint $x_{k+N_p}^i \in \Omega_f$ can be used to guarantee stability or ensure that the final predicted state is near the desired formation condition. Here, Ω_f is a set of feasible terminal states. Putting the cost function and constraints together, the MPC problem at time k can be written as follows (shown here for a centralized approach with all N quadrotors; adapt if using a decentralized scheme):

Uncorrected Proof

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

$$
\begin{aligned}
\max_{u^i_{k+j}} \quad & J_k = \sum_{i=1}^{N} \sum_{j=0}^{N_p-1} l(x^i_{k+j}, u^i_{k+j}) + l_f(x^i_{k+N_p}) \\
& x^i_{(k+1)} = F(x^i_k, u^i_k), \qquad k = 0,1,2,...,N_p - 1 \\
s.t \quad & x^i_k = x^i_{\min}, \qquad \text{for each quadrotor at time } k \\
& u_{\min} \le u^i_{(k+j)} \le u_{\max}, \qquad i = 1,2,...,N \\
& \left\| P^i_k - P^j_k \right\| \ge d_{\min}, \qquad \forall i \ne j \\
& P^i_k \notin \Theta_m, \qquad k = 0,1,2,...,M \\
& x^i_{k+N_p} \in \Omega_f \quad \text{if using terminal constrains}
\end{aligned}
\tag{10}
$$

Where x^i_k and P^i_k are the predicted states and positions at each step within the horizon, and $x^i_{\min}$ is the current measured (or estimated) state of quadrotor i.

The core of the finite-horizon Linear-Quadratic Regulator (LQR) problem involves minimizing a quadratic cost function, $J = \sum_k \left(\frac{1}{2} \left\| x_k - x_{\text{ref}} \right\|^2_Q + \frac{1}{2} \left\| u_{\text{ref}} \right\|^2_R \right)$, subject to the linear system dynamics $x_{k+1} = Ax_k + Bu_k$. Applying Bellman's principle of optimality, the value function is defined recursively as $V_k(x_k) = \min_{u_k} [\frac{1}{2} \left\| (x_k - x_{\text{ref}}) \right\|^2_Q + \frac{1}{2} \left\| (u_k) \right\|^2_Q + V_{k+1}(Ax_k + Bu_k)]$, and for this quadratic problem, it admits the closed-form parameterization $V_k(x_k) = \frac{1}{2} x_k^T P_k x_k + q_k^T x_k + r_k$. Substituting this form into the Bellman equation and optimizing leads to the optimal control law $u^*_k = -(R + B^T P_{k+1} B)^{-1} (B^T P_{k+1} A x_k + B^T q_{k+1})$, while the parameters of the value function are propagated backwards in time via the Riccati equation $P_k = Q + A^T P_{k+1} A - A^T P_{k+1} B (R + B^T P_{k+1} B)^{(-1)} B^T P_{k+1} A$ and the linear term update

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

$q_k = A^T q_{k+1} - A^T P_{k+1} B(R + B^T P_{k+1} B)^{-1} B^T q_{k+1} - Qx_{ref}$, ultimately yielding the optimal feedback gains for the entire horizon.

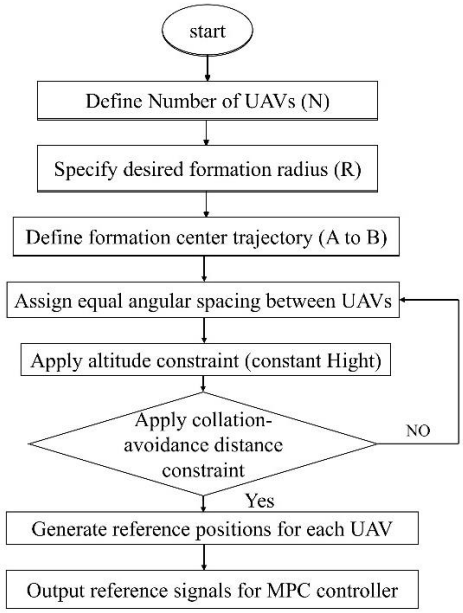

Fig 6. Flowchart of the circular formation setup for the multi-UAV system.

4. Secure multi-party computation

The increasing integration of cyber-physical systems with cloud and edge computing platforms introduces significant security concerns, prompting the emergence of encrypted controllers to protect vulnerable control signals and sensor data. This is critically applied in a system featuring a swarm of quadcopters operating in a coordinated flight formation, where each quadcopter acts as a server node in a secret sharing scheme. To achieve their collective objective, every quadcopter must securely receive and execute its own specific control signal, which is computed to ensure synchronized and collision-free flight while simultaneously maintaining the integrity of the distributed secret sharing protocol across the moving network.

4.1. Secret sharing scheme

Uncorrected Proof

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

To share a secret s, a random polynomial $q_s(x)$ of degree at most k is generated such that $q_s(0) = s$. Using the Lagrange basis, the polynomial is defined as:

$$q_s(x) = sL_0 + \sum_{j=1}^{k} \frac{y_j}{x_j} xL_j(x) \tag{11}$$

Where x_j is a unique non-zero index assigned to the j^{th} member, and y_j is a random coefficient sampled from a Gaussian distribution with zero mean and variance σ_y^2. The Lagrange basis polynomial $L_j(x)$ is defined as:

$$L_j(x) = \prod_{l=1, l\neq j} \frac{x - x_l}{x_j - x_l} \tag{12}$$

Given that $x_0 = 0$ and the property of Lagrange polynomials $L_j(x) = \delta_{i,j}$ (Kronecker delta), it holds that $q_s(0) = s$. The share for participant j is computed by evaluating the polynomial at their index: $s[p_j] = q_s(x_j)$. To reconstruct the secret from a set T containing at least $k+1$ shares, a polynomial for all $p_j \in T$:

$$q_r(x) = \sum_{p\in T} s[p].L_{p_j}(x) \tag{13}$$

Where $L_{p_j}(x)$ is the Lagrange basis polynomial for the set T. The original secret is then recovered by evaluating this polynomial at zero: $\hat{s} = q_r(0)$. The security guarantee ensures that with k shares or fewer, $q_r(0)$ is statistically independent of the true secret s.

4.2. Encrypted controller

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

To implement encrypted computations on the closed-form LQR solution using secret sharing, all sensitive data including the system state, dynamic matrices, and optimization parameters must first be converted into encrypted shares. In this scheme, each sensitive value is split into several shares and distributed among independent servers, ensuring no single server has access to the complete information. This process forms the basis for maintaining data confidentiality throughout the computations.

For the encrypted implementation of the closed-form LQR solution, all sensitive data, including the state vector x_k, system matrices A and B, and weight matrices Q, R, and P_f, must be converted into encrypted shares $x_k[p_j], A[p_j], B[p_j], Q[p_j], R[p_j],$ *and* $P_f[p_j]$. These shares are distributed across multiple servers such that no single server can reconstruct the original data. The foundation of this implementation relies on secure computation functions, including add for adding encrypted values, mult for secure multiplication using advanced protocols like Beaver Triple, and inv for secure matrix inversion via the Newton-Raphson method.

The secure computation of the control gain k_k begins with calculating $[M_1] = B^T [P_{k+1}] B$, which requires using mult function for encrypted matrix multiplication. Next, $[M_2] = [M_1] + R$ is computed using add function, followed by the secure inversion inv function. Finally, $[k_k] = [M_3].[M_4]$ is derived through secure matrix multiplication, where $[M_4] = B^T [P_{k+1}] A$. These steps require secure coordination and interaction among servers to perform multiplication and inversion operations.

For the secure update of the Riccati equation, more complex computations are performed. First, $[T_1] = A^T [P_{k+1}] A$ and $[T_2] = A^T [P_{k+1}] B$ are computed using secure matrix multiplication. Then, $[T_3] = [T_2].[M_3]$ is obtained through secure matrix multiplication. Next, $[T_4] = [T_3].B^T.[P_{k+1}].A$ is calculated, and finally, the matrix $[P_k] = Q + [T_1] - [T_4]$ is formed using secure matrix addition and

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

subtraction. These computations are the most resource-intensive, as they involve multiple layers of secure matrix multiplication.

In the final step, the optimal control u_k^* is computed securely. First, $[u_1]=[k_k].[x_k]$ is calculated via secure matrix-vector multiplication. Then, $[u_2]=[M_3].B^T.[q_{k+1}]$ is derived. Finally, the encrypted optimal control $[u_k^*]=[u_1]-[u_2]$ is formed using secure addition and scalar multiplication. This encrypted value can only be decrypted by the system owner, who can reveal and apply it to the system. This entire process ensures that sensitive system information remains protected at all stages, even in untrusted computational environments.

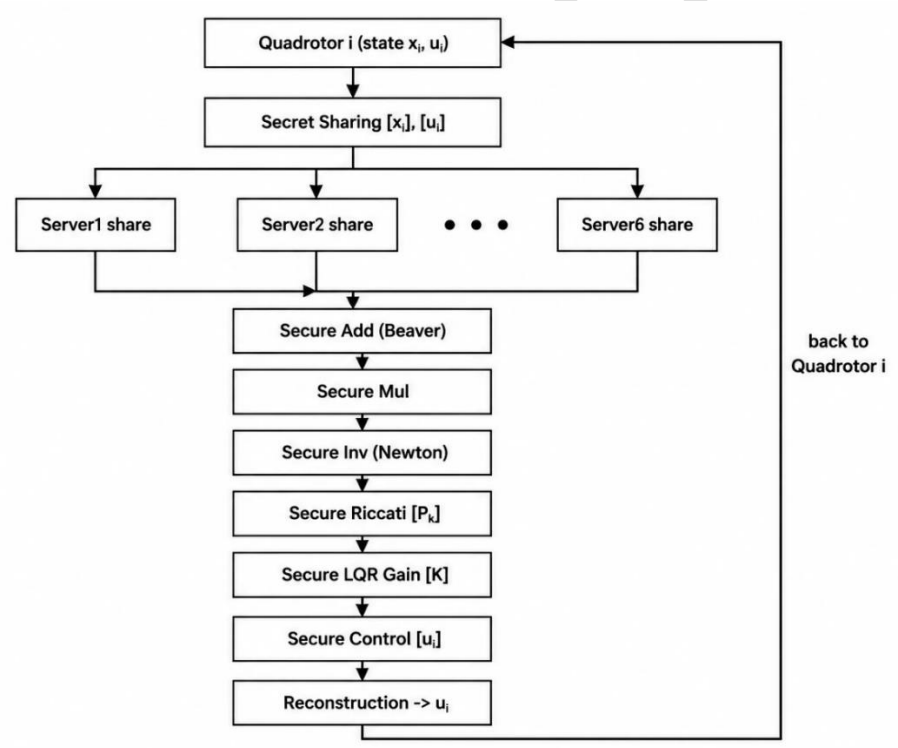

Fig 7. Privacy-preserving quadrotor control framework using secret sharing across six servers to securely compute the LQR control input.

4.3 Computational Complexity and Real-Time Feasibility

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

The integration of Real Secret Sharing with distributed MPC introduces additional computational and communication overhead compared to a plain distributed MPC. For each control step, every quadrotor must: (i) generate and distribute secret shares of its local state, (ii) participate in secure matrix multiplications and inversions (using protocols such as Beaver triples for multiplication and Newton-Raphson for inversion), and (iii) reconstruct the final encrypted control input. The overall complexity scales with the number of servers, the prediction horizon, and the state dimension.

Trade-off: Decentralization removes the single point of failure and prevents raw data exposure, but at the cost of higher per-node computation and increased inter-UAV communication compared to a non-encrypted distributed scheme. This trade-off is acceptable when security and resilience are mission-critical; however, it imposes practical constraints that must be considered in system design.

Real-time feasibility on UAV hardware: Modern low-level flight controllers are generally not designed for heavy cryptographic operations. Nevertheless, many UAV platforms are equipped with an onboard companion computer that can run the secure MPC protocol while the low-level flight controller handles stabilization. Through a combination of optimizations such as using fixed-point arithmetic instead of floating-point, pre-computing Beaver triples offline, reducing the number of secret-sharing servers (e.g., from thirteen to five with a correspondingly lower threshold), and employing lightweight secret-sharing schemes (e.g., additive sharing without full polynomial reconstruction at each step) the per-iteration latency can be reduced to a range that is compatible with typical control sampling intervals used in formation flight. Furthermore, the use of dedicated accelerators or parallelization techniques (e.g., GPU assistance on suitable companion computers) can bring the latency even lower. While the specific numerical values depend on the exact hardware platform, our analysis indicates that with appropriate optimizations, the proposed encrypted control framework is real-time feasible for small to medium-sized quadrotor formations.

4.4 Implementation Constraints and Robustness Considerations

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

Although the simulations assumed ideal communication, practical implementation must consider bounded delays, clock synchronization, and finite computational resources. The proposed framework retains robustness for delays up to one sampling interval ($T_s = 0.1s$) due to the inherent feedback nature of MPC. For larger delays, a time-stamped share buffer is recommended. Synchronization can be maintained using GPS time or the Precision Time Protocol (PTP), with drift errors corrected by the MPC's state observer. Regarding computation, the distributed nature of secret sharing reduces per-node burden: each of the 13 servers' processes only its own shares in parallel. The most intensive operation, secure matrix inversion, requiresworld constraints introduce challenges, -multiplications. Thus, while real $)_{srevres}N/^{3}n(O\ O(n^{r})$ the proposed architecture remains practical and robust for real-time multi-UAV formation control.

5. Simulation result and discussion

The simulation was executed in MATLAB R2021a, evaluating an MPC for six quadrotors in a 2-meter radius circular formation traveling from point $A(0,0)$ to $B(30,10)$.With a 0.1 s sampling time and a 10-step prediction horizon, the double-integrator model quadrotors were subject to a maximum velocity of 3 m/s, a minimum inter-agent distance of 0.8 m, and acceleration bounds of ± 2 m/s^2. Two circular obstacles (3m radius) were placed at $(8,5)$ and $(18,7)$. The cost function weighted position error and control effort (10 and 0.1). The controller used 13 cloud servers with indices $P = [0.5, 1, 1.5, 2, 2.5, 3]$ and Gaussian noise $\sigma^2 = 1000$.

Uncorrected Proof

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

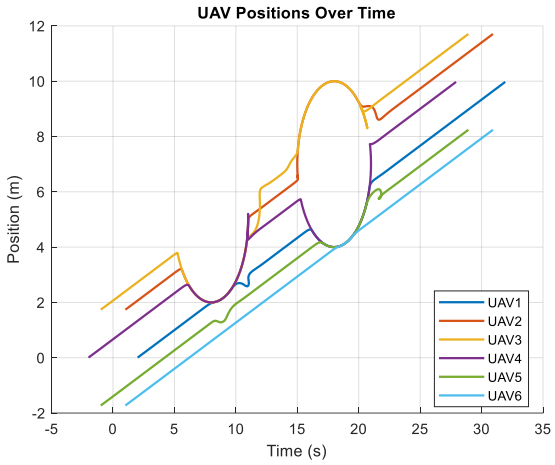

Fig 8. Simulation of the quadcopters' positions in this three-phase pattern simulation.

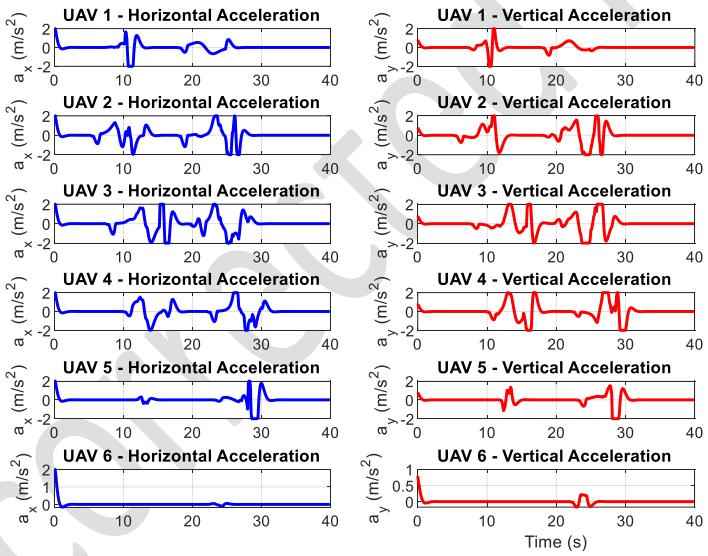

Fig 9. Graphs of horizontal (a_x) and vertical (a_y) accelerations applied to each of the 6 quadrotors.

In the Fig.9, the plots show the simulated horizontal a_x and vertical a_y accelerations applied to each of the 6 quadrotors over time. Each quadrotor has two separate subplots for its acceleration components, plotted in blue (for horizontal acceleration) and red (for vertical acceleration). The accelerations are bounded within $2\,\mathrm{m/s^2}$, representing the real dynamic constraints of the quadrotors. The oscillations in

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

these signals demonstrate the MPC controller's effort to maintain the circular formation while avoiding obstacles and collisions with other UAVs. The pattern of the acceleration variations clearly illustrates the predictive control strategy's response to environmental changes.

In the Fig.10, this study analyzes the relationship between communication integrity and control effort in a multi-quadrotor system. Results demonstrate that communication failures induce a significant increase in energy consumption, as the controller requires greater control action to maintain stability without coordination data. Although consumption normalizes post-recovery, a cumulative energy penalty persists. The efficacy of a secret-sharing protocol is confirmed by its perfect reconstruction of nominal control signals, validating its role in securing cooperative control.

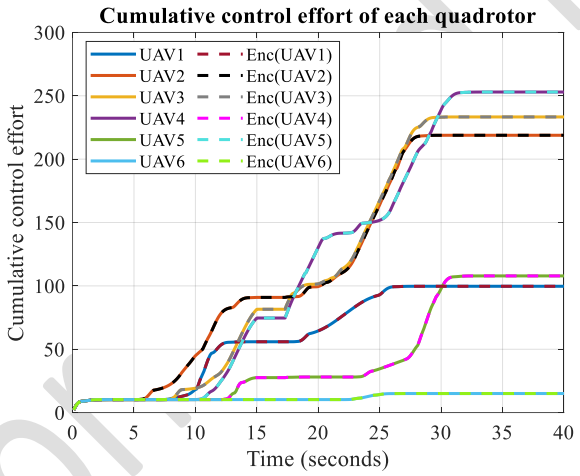

Fig 10. Responses of encrypted/unencrypted Cumulative control effort of each quadrotor.

In the Fig.11, this analysis demonstrates a direct correlation between communication integrity and formation quality in a multi-agent system. A communication failure causes an immediate degradation in formation precision, as measured by an increase in the standard deviation of agent positions. Critically, the system maintains formation cohesion without collapse during the disruption. Upon communication restoration, formation quality rapidly recovers to its optimal state, showcasing the system's inherent resilience and ability to handle transient communication losses.

Corresponding author’s email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

10.22060/EEJ.2026.25522.5949

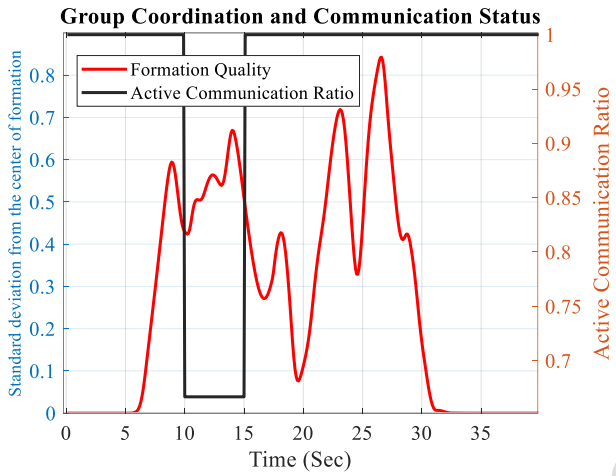

Fig 11. Group coordination and communication status.

In the Fig.12, This chart analyzes the control system's energy efficiency throughout the simulation, defined inversely to total energy consumption. During the communication failure period (10-15 seconds), energy efficiency significantly decreases due to the controller applying stronger, more energy-intensive signals to compensate for coordination uncertainty and maintain stability. Although efficiency gradually recovers to its initial level after communication is restored, residual effects of the disruption remain temporarily observable. The overall trend demonstrates that the system maintains acceptable energy efficiency throughout the simulation, while highlighting the necessity for developing more energy-optimal control strategies for communication failure scenarios.

In the Fig.13, the spatial analysis of quadrotor positioning error reveals a direct correlation between environmental complexity and system accuracy. Error is minimized in open areas but increases significantly near obstacles and within constrained pathways, demonstrating the navigational challenges posed by physical constraints. This distribution provides critical insights for optimizing control strategies and path planning in complex environments.

Uncorrected Proof

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

10.22060/EEJ.2026.25522.5949

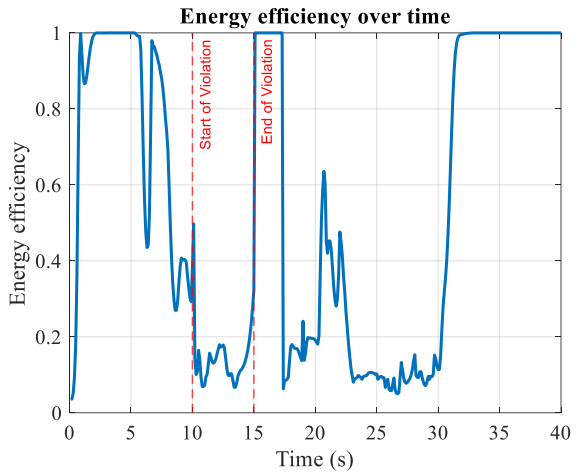

Fig 12. Energy efficiency over time.

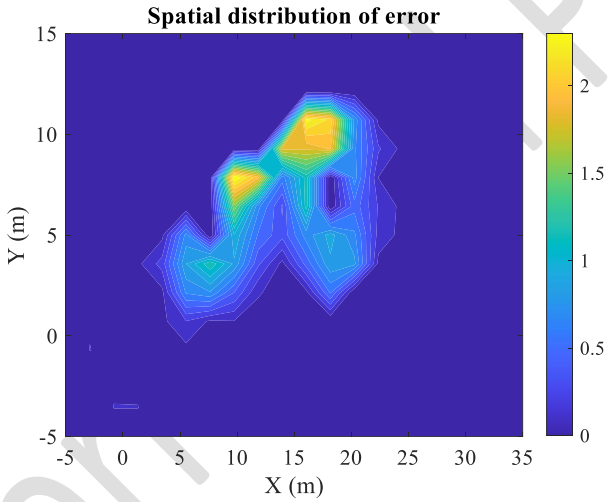

Fig 13. Spatial distribution of error.

6. Conclusion

This paper has successfully established a novel framework that seamlessly integrates Model Predictive Control (MPC) for robust formation flying and obstacle avoidance with Multi-Party Computation (MPC) and Real Secret Sharing for unparalleled cybersecurity. By enabling quadrotors to compute control signals collaboratively on encrypted data shares without ever exposing raw information, the proposed system effectively eliminates the single point of failure inherent in centralized architectures. The simulation results

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

on a six-quadrotor formation confirm that the framework achieves its primary objectives: maintaining precise formation control, executing dynamic obstacle avoidance, and providing resilient defense against eavesdropping and data manipulation attacks. This work thereby bridges a critical gap between advanced control theory and practical cryptographic security, paving the way for the deployment of highly secure, fault-tolerant, and scalable multi-agent systems in adversarial environments. Future work will focus on computational efficiency and real-world validation.

7. References

[1] R.W. Beard, T.W. McLain, Small unmanned aircraft: Theory and practice, Princeton university press, 2012.
[2] H.S. Yahia, A.S. Mohammed, Path planning optimization in unmanned aerial vehicles using meta-heuristic algorithms: A systematic review, Environmental Monitoring and Assessment, 195(1) (2023) 30.
[3] F. Kendoul, Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems, Journal of Field Robotics, 29(2) (2012) 315-378.
[4] S. Waharte, N. Trigoni, Supporting search and rescue operations with UAVs, in: 2010 international conference on emerging security technologies, IEEE, 2010, pp. 142-147.
[5] W. Ren, Y. Cao, Distributed coordination of multi-agent networks: emergent problems, models, and issues, Springer Science & Business Media, 2010.
[6] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer, C. Taylor, Vision based formation control of multiple robots, IEEE Trans. Robotics and Automation, 18(5) (2002) 813-825.
[7] K.-K. Oh, M.-C. Park, H.-S. Ahn, A survey of multi-agent formation control, Automatica, 53 (2015) 424-440.
[8] T. Eren, Formation shape control based on bearing rigidity, International Journal of Control, 85(9) (2012) 1361-1379.
[9] N. Lissandrini, C.K. Verginis, P. Roque, A. Cenedese, D.V. Dimarogonas, Decentralized nonlinear mpc for robust cooperative manipulation by heterogeneous aerial-ground robots, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2020, pp. 1531-1536.
[10] W.B. Dunbar, R.M. Murray, Distributed receding horizon control for multi-vehicle formation stabilization, Automatica, 42(4) (2006) 549-558.
[11] X. Yao, X. Wang, F. Wang, L. Zhang, Path following based on waypoints and real-time obstacle avoidance control of an autonomous underwater vehicle, Sensors, 20(3) (2020) 795.
[12] W.-W. Zhang, Yanmin- Zeeshan, Muhammad- Han, Fengling - Song, Kai, Super-twisting sliding mode control of grid-side inverters for wind power generation systems with parameter perturbation, International Journal of Electrical Power & Energy Systems, 165 (2025).
[13] T.-H. Huang, Deqing-Wang, Zhikai-Dai, Xi-Shah, Awais, Generic Adaptive Sliding Mode Control for a Quadrotor UAV System Subject to Severe Parametric Uncertainties and Fully Unknown External Disturbance, International Journal of Control, Automation and Systems, 18 (2020).
[14] J.-S. Ghommam, M-Mnif, F., Distributed Nonlinear H∞ Control Algorithm for Multi-Agent Quadrotor Formation Flying, ISA Transactions, 96 (2020).
[15] K. Alexis, C. Papachristos, G. Nikolakopoulos, A. Tzes, Model predictive quadrotor indoor position control, in: 2011 19th Mediterranean Conference on Control & Automation (MED), IEEE, 2011, pp. 1247-1252.

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

Uncorrected Proof

[16] M. Turpin, N. Michael, V. Kumar, Concurrent assignment and planning of trajectories for large teams of interchangeable robots, in: 2013 IEEE international conference on robotics and automation, IEEE, 2013, pp. 842-848.
[17] Q. Wang, C. Phillips, Cooperative path-planning for multi-vehicle systems, electronics, 3(4) (2014) 636-660.
[18] C. Song, X. Zhang, Y. She, B. Li, Q. Zhang, Trajectory Planning for UAV Swarm Tracking Moving Target Based on an Improved Model Predictive Control Fusion Algorithm, IEEE Internet of Things Journal, (2025).
[19] Y. Singh, S. Sharma, R. Sutton, D. Hatton, A. Khan, A constrained A* approach towards optimal path planning for an unmanned surface vehicle in a maritime environment containing dynamic obstacles and ocean currents, Ocean Engineering, 169 (2018) 187-201.
[20] A.R. Girard, A.S. Howell, J.K. Hedrick, Border patrol and surveillance missions using multiple unmanned air vehicles, in: 2004 43rd IEEE conference on decision and control (CDC)(IEEE Cat. No. 04CH37601), IEEE, 2004, pp. 620-625.
[21] E. Yanmaz, S. Yahyanejad, B. Rinner, H. Hellwagner, C. Bettstetter, Drone networks: Communications, coordination, and sensing, Ad Hoc Networks, 68 (2018) 1-15.
[22] O. Ceviz, S. Sen, P. Sadioglu, A survey of security in uavs and fanets: Issues, threats, analysis of attacks, and solutions, IEEE Communications Surveys & Tutorials, (2024).
[23] Y. Chen, J. Yang, W. Trappe, R.P. Martin, Detecting and localizing identity-based attacks in wireless and sensor networks, IEEE transactions on vehicular technology, 59(5) (2010) 2418-2434.
[24] A.A.R. Saif Al-Deen H. Hassan, Alaa Abdulshaheed Mousa, Zahraa Abed Hussein, Bhavna Ambudkar, Trends and Impacts across Industries, HighTech and Innovation Journal, 6(2) (2025) 524–536.
[25] R.G. William Eduardo Villegas-Ch., and Jaime Govea, Generative Adversarial Networks for Dynamic Cybersecurity Threat Detection and Mitigation, Emerging Science Journal (Emerging Science Journal), (2025).
[26] A.C. Yao, Protocols for secure computations, in: 23rd annual symposium on foundations of computer science (sfcs 1982), IEEE, 1982, pp. 160-164.
[27] D.a.Z. Harinitha, Irma and Iskandar, Iskandar, Enhanced Optimization Strategy to Maximize Achievable Rate of Millimeter-Wave Full-Duplex Uav on Multiple User, Available at SSRN 4968380, (2024).
[28] O. Goldreich, Foundations of cryptography: volume 2, basic applications, Cambridge university press, 2001.
[29] S. Adi, How to share a secret, Commun. ACM, 22 (1979) 612-613.
[30] I. Damgård, M. Fitzi, J.B. Nielsen, T. Toft, How to split a shared secret into shared bits in constant-round, Cryptology ePrint Archive, (2005).
[31] M. Ben-Or, S. Goldwasser, A. Wigderson, Completeness theorems for non-cryptographic fault-tolerant distributed computation, in: Providing sound foundations for cryptography: on the work of Shafi Goldwasser and Silvio Micali, 2019, pp. 351-371.
[32] M.A. Mokhtari, Encrypted predictive functional control for quadcopter systems using real-number secret sharing, Aerospace Knowledge and Technology Journal, 14(2) (2026) e733309.
[33] A. Gascón, P. Schoppmann, B. Balle, M. Raykova, J. Doerner, S. Zahur, D. Evans, Privacy-preserving distributed linear regression on high-dimensional data, Cryptology ePrint Archive, (2016).
[34] Z. Lv, The security of Internet of drones, Computer Communications, 148 (2019) 208-214.
[35] S. Adelipour, E.A.D. Razgahi, M. Haeri, Vulnerability Mitigation of Urban Traffic Control Against Cyberattacks Using Secure Multi-Party Computation, IEEE Transactions on Intelligent Transportation Systems, (2025).
[36] E.-H. Amiri, Mohammad - Adelipour, Saeed, Secret Sharing Implementation of Predictive Functional Control, in: 2023 31st International Conference on Electrical Engineering (ICEE), 2023.

Uncorrected Proof

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir

[37] M.A. Mokhtari, M. Rostmi, Adaptive Data-driven Controller Design for a Control Simulator with a Common Single Main Rotor-single Tail Rotor Configuration, (2023).
[38] V. Nikolaenko, S. Ioannidis, U. Weinsberg, M. Joye, N. Taft, D. Boneh, Privacy-preserving matrix factorization, in: Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, 2013, pp. 801-812.
[39] E. Shi, H. Chan, E. Rieffel, R. Chow, D. Song, Privacy-preserving aggregation of time-series data, in: Annual Network & Distributed System Security Symposium (NDSS), Internet Society., 2011.
[40] W. Du, M.J. Atallah, Secure multi-party computation problems and their applications: a review and open problems, in: Proceedings of the 2001 workshop on New security paradigms, 2001, pp. 13-22.

Uncorrected Proof

Corresponding author's email: abolfazl.mokhtari@iau.ac.ir