Predictive Control over LAWN: Joint Trajectory Design and Resource Allocation

Haijia Jin, Jun Wu, Weijie Yuan, *Senior Member, IEEE*, Ruizhi Ruan, Jiacheng Wang, *Member, IEEE*, Dusit Niyato, *Fellow, IEEE*, Dong In Kim, *Life Fellow, IEEE*, and Abbas Jamalipour, *Fellow, IEEE*

arXiv:2507.02374v1 [eess.SP] 3 Jul 2025

***Abstract*—Low-altitude wireless networks (LAWNs) have been envisioned as flexible and transformative platforms for enabling delay-sensitive control applications in Internet of Things (IoT) systems. In this work, we investigate the real-time wireless control over a LAWN system, where an aerial drone is employed to serve multiple mobile automated guided vehicles (AGVs) via finite blocklength (FBL) transmission. Toward this end, we adopt the model predictive control (MPC) to ensure accurate trajectory tracking, while we analyze the communication reliability using the outage probability. Subsequently, we formulate an optimization problem to jointly determine control policy, transmit power allocation, and drone trajectory by accounting for the maximum travel distance and control input constraints. To address the resultant non-convex optimization problem, we first derive the closed-form expression of the outage probability under FBL transmission. Based on this, we reformulate the original problem as a quadratic programming (QP) problem, followed by developing an alternating optimization (AO) framework. Specifically, we employ the projected gradient descent (PGD) method and the successive convex approximation (SCA) technique to achieve computationally efficient sub-optimal solutions. Furthermore, we thoroughly analyze the convergence and computational complexity of the proposed algorithm. Extensive simulations and AirSim-based experiments are conducted to validate the superiority of our proposed approach compared to the baseline schemes in terms of control performance.**

***Index Terms*—Low-altitude wireless network, aerial drone, automated guided vehicle, finite blocklength, model predictive control, outage probability.**

I. Introduction

The rapid evolution of the Internet of Things (IoT) has driven the deployment of large-scale, distributed systems that require robust, low-latency, and adaptive wireless connectivity. To meet these demands, low-altitude wireless networks (LAWNs) have emerged as a promising paradigm for near-ground IoT applications such as smart factories, urban mobility, and intelligent logistics [1]. Thanks to their close proximity to ground-level devices, LAWN architectures offer several inherent advantages over conventional terrestrial or satellite networks, including flexible deployment, enhanced spatial resolution, and the ability to maintain line-of-sight (LoS) links in complex or dynamic environments. Furthermore, the low-altitude operational domain enables LAWNs to accommodate dense device connectivity and facilitate swift topological adjustments, rendering them ideal for time-critical and mission-sensitive IoT services [2], [3].

Within LAWN systems, drone-assisted communication has emerged as a cornerstone technology for enabling reliable and flexible wireless connectivity in dynamic, large-scale, or infrastructure-limited environments. Drones can exploit their aerial mobility and LoS advantages to dynamically establish high-quality links with ground nodes, thereby supporting low-latency and resilient data transmission [4]–[6]. To fully harness these benefits, extensive studies have been conducted on optimizing drone deployment, user association, and spectrum allocation. For example, Mozaffari *et al.* [7] formulated a joint optimization framework for drone mobility, device association, and power control in IoT networks. In addition, adaptive drone trajectory designs have been proposed to improve link quality and enhance security, particularly in mobile or adversarial environments [8], [9]. These efforts have significantly advanced the capabilities of drone-enabled communication services in LAWN scenarios.

Complementary to the communication layer, real-time control of ground agents like automated guided vehicles (AGVs) constitutes another critical function within LAWN applications, particularly in domains such as industrial automation, logistics, and smart transportation [10], [11]. Among various control strategies, model predictive control (MPC) stands out for its ability to anticipate future trajectories and optimize control actions with physical and operational constraints in a receding-horizon manner. Compared with classical approaches such as proportional-integral-derivative (PID) and linear quadratic regulator (LQR) controllers [12], [13], MPC offers greater adaptability and performance robustness in complex and dynamic environments. It has been widely applied in AGV trajectory tracking, multi-agent coordination, and collision avoidance tasks [14]–[16]. However, in most existing studies, communication and control are treated as two independent subsystems. Communication is responsible for ensuring data delivery, while control focuses on generating optimal motion commands, without explicitly modeling their mutual dependency, leading to suboptimal system behavior, especially when time delays, packet losses, or limited bandwidth affect control execution.

H. Jin, J. Wu, R. Ruan, and W. Yuan are with the School of Automation and Intelligent Manufacturing, Southern University of Science and Technology, Shenzhen 518055, China (e-mails: {jinhj2024, wuj2021, ruanrz2024}@mail.sustech.edu.cn; yuanwj@sustech.edu.cn).

Jiacheng Wang and Dusit Niyato are with the College of Computing and Data Science, Nanyang Technological University, Singapore 639798 (e mails: jiacheng.wang@ntu.edu.sg; dniyato@ntu.edu.sg).

D. I. Kim is with the Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon 16419, South Korea (e-mail: dongin@skku.edu).

A. Jamalipour is with the School of Electrical and Information Engineering, the University of Sydney, Sydney, NSW 2006, Australia (e-mail: a.jamalipour@ieee.org).

To address the growing need for real-time responsiveness and reliability in networked control systems, recent research has increasingly focused on the joint optimization of communication and control, giving rise to the concept of wireless networked control systems (WNCSs). Unlike conventional approaches that design communication protocols and control strategies separately, WNCS aims to coordinate both subsystems in a unified manner to enhance closed-loop performance under stringent bandwidth, latency, and energy constraints [17]–[20]. The integrated perspective stems from the recognition that control performance is fundamentally influenced by the quality and timeliness of information exchange, particularly in time-sensitive applications such as trajectory tracking and multi-agent coordination. While existing joint design studies focus on trajectory planning and resource allocation at the *outer loop* level, they typically overlook how communication unreliability affects the stability of *inner loop* control mechanisms, such as drone attitude regulation and real-time feedback control in LAWNs [21]. Toward this end, Kostina *et al.* [22] established a fundamental lower bound on the minimum communication rate required to achieve a specified LQR cost, providing an information-theoretic benchmark for control-aware communication design. Building on this, subsequent works have formulated cross-layer optimization problems that jointly consider control accuracy and communication throughput, as exemplified by recent efforts in integrated sensing-communication-computing-control ($\mathbf{SC^3}$) architectures for satellite-drone systems [23]. Meanwhile, algorithmic advances have sought to address practical challenges in wireless environments. For instance, Cao *et al.* [24] developed a robust MPC framework that dynamically adapts to variable delays, measurement noise, and packet drops, thus ensuring reliable AGV tracking even under fluctuating network conditions. Despite these developments, most existing approaches implicitly assume the availability of large-packet or infinite blocklength (IBL) transmissions, which do not accurately reflect the stringent latency and reliability requirements encountered in real-time control systems. In such systems, operational periods are typically segmented into numerous short time slots to support agile communication and timely control decisions [25], making short-packet transmission not only common but also essential.

To address this mismatch, finite blocklength (FBL) transmission has been adopted as a more realistic communication model in real-time control settings. Compared to IBL transmission, where the achievable rate follows Shannon capacity, FBL introduces a more complex rate expression that depends on the signal-to-noise ratio (SNR), block error rate (BLER), and blocklength [26], which poses significant challenges for system optimization, as the rate function under FBL is typically non-convex and highly sensitive to system parameters, thereby making resource allocation and control-communication co-design considerably more difficult than in IBL-based systems [27]. Recent studies have applied FBL theory in various drone communication scenarios. For instance, the secrecy and covertness performance of FBL-based drone communication was investigated in [28] and [29], while Ruat *et al.* [30] analyzed reliability optimization in a drone-assisted nonlinear energy harvesting full-duplex network under both IBL and FBL regimes. However, to the best of our knowledge, most existing studies have primarily focused on either optimizing control strategies under communication constraints or enhancing wireless resource allocation to support control performance, while often overlooking the intricate interplay between communication reliability and control actions, particularly in latency-critical real-time control scenarios. Motivated by the above, this paper investigates the interaction between predictive control and wireless communication in the context of LAWN. Specifically, we consider a real-time control scenario where an aerial drone serves multiple mobile AGVs via FBL transmissions. To enable accurate and delay-sensitive trajectory tracking, we adopt an MPC strategy and jointly optimize the control policy, transmit power, and drone trajectory. Under this framework, our goal is to minimize the control cost while ensuring communication reliability and satisfying physical constraints such as mobility and power limits. The main contributions of this work are summarized as follows.

1) We develop a joint communication-control framework for the drone-assisted LAWN system, enabling real-time remote control of multiple AGVs under FBL transmission. We adopt the MPC method to guide the drone's real-time decision-making. Moreover, we analyze the communication reliability using the outage probability to ensure reliable wireless transmission and support accurate, low-latency trajectory tracking.

2) To accomplish the control task, we formulate a joint optimization problem to minimize the total control cost of real-time AGV operations by jointly optimizing control inputs, power allocation, and drone trajectory, subject to constraints on maximum transmit power, actuation limits, and drone mobility and flight region.

3) We first derive the closed-form expression of the outage probability under FBL transmission. Based on this, we recast the formulated non-convex problem into a quadratic programming (QP) structure and decomposed it into a sequence of tractable sub-problems. We then develop an alternating optimization (AO) framework to iteratively update control actions, power allocation, and drone trajectory by leveraging projected gradient descent (PGD) and successive convex approximation (SCA). We rigorously analyze the convergence properties and computational complexity of the proposed algorithm to validate its practicality and scalability.

4) Extensive simulations confirm that the proposed algorithm enables the drone to robustly track AGVs even under abrupt trajectory variations and consistently outperforms existing benchmarks in control performance. Its effectiveness is further validated on the AirSim platform [31], offering practical insights into the design of drone-assisted wireless control systems in LAWN.

The remainder of this article is organized as follows. Section II introduces the drone-assisted LAWN control network. Section III formulates the joint optimization of the MPC-based control input, power allocation, and drone trajectory, which is subsequently addressed in Section IV. Simulation and experimental results are presented in Section V, followed by the conclusions in Section VI.

Table I: List of Key Symbols

Symbol	Description	Symbol	Description
$\mathbf{q}[n]$	Horizontal coordinate of aerial drone	H	Fixed altitude of aerial drone
K	Number of AGVs	$\mathcal{K}$	Set of AGVs
$\mathbf{c}_k[n]$	Horizontal position of the k-th AGV	T	Total time duration
N	Number of discrete time slots	$\Delta\tau$	Duration of each time slot
a_{LoS}, b_{LoS}	LoS model parameters	$\theta_k[n]$	Elevation angle
$P_{k,\text{LoS}}[n]$	LoS probability	X_{NLoS}	Small-scale fading
$d_k[n]$	Distance between drone and AGV	$h_k[n]$	Channel between drone and AGV
α_0	Reference channel gain at 1 m	σ_k^2	Noise power
$p_k[n]$	Transmit power	$\Gamma_k[n]$, Γ_k^{th}	SNR and SNR threshold
B	Channel bandwidth	ϵ	BLER
l_k	Blocklength of transmitted signal	$R_k[n]$, R_k^{th}	Achievable rate and rate threshold
$P_k[n]$	Outage probability	$V_k[n]$	Channel dispersion
$\mathbf{x}_k[n]$	AGV's state vector	$\mathbf{u}_k[n]$	Control input
$\mathbf{A}_k$, $\mathbf{B}_k$	State transition and input matrices	$\tilde{\mathbf{x}}_k[n]$	Augmented AGV state
$\tilde{\mathbf{A}}_k$, $\tilde{\mathbf{B}}_k$	Augmented system matrices	$\Delta\mathbf{u}_k[n]$	Augmented control input
$\tilde{\mathbf{x}}_{k,\text{ref}}[n]$	Reference AGV state	$\mathbf{e}_k[n]$	Predicted deviation
N_p	Prediction horizon of MPC	$\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{Q}_3$	Weighting matrices of control cost
J_n	Control cost function	$\Delta\boldsymbol{\mu}_k[n]$	Control input sequence over horizon
$\boldsymbol{\mathcal{U}}_k$	Predefined control set	f_{QoS}	Communication QoS
$P_{\max}$	Maximum transmit power	$V_{\max}$	Maximum drone speed
$\mathcal{D}$	Aerial drone flight region	$\boldsymbol{\chi}_k[n]$	Predicted control state vector
$\boldsymbol{\chi}_{k,\text{ref}}[n]$	Reference state over horizon	$\hat{J}_n$	Control cost of reformulated QP
$\mathbf{Q}_4, \mathbf{Q}_5$	Weighting matrices of reformulated QP	$\mathbf{G}_k[n]$, $\mathbf{H}_k[n]$, $M_k[n]$, $L_k[n]$, $S_k[n]$, $W_k[n]$	Coefficients of reformulated QP

Notations: Unless otherwise specified, bold lowercase and uppercase letters, e.g., $\mathbf{m}$ and $\mathbf{M}$, represent vectors and matrices, respectively. $\mathbb{R}^M$ denotes the space of M-dimensional real-valued column vectors. $\mathbb{E}(\cdot)$, $\|\cdot\|$, and $(\cdot)^T$ denote the expectation, ℓ_2 norm, and matrix transposition, respectively. $\mathcal{N}(\mu, \sigma^2)$ denotes the Gaussian distribution with mean μ and variance σ^2. $\text{Exp}(\lambda)$ denotes the exponential distribution with rate λ. $\Pr(\cdot)$ represents the probability of an event. In addition, $\mathbf{I}_M$ denotes the $M \times M$ identity matrix, and the function $Q^{-1}(\cdot)$ is the inverse of the Gaussian Q-function, defined as $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp(-\frac{t^2}{2})\text{d}t$. Moreover, to facilitate understanding of the mathematical derivations, the key symbols used throughout the system and optimization models are summarized in Table I.

II. System Model

As shown in Fig. 1, we investigate a LAWN system, in which a single-antenna drone remotely controls K AGVs for real-time trajectory tracking. The time-varying horizontal coordinate of the drone is denoted as $\mathbf{q}[n] = [x[n], y[n]]^{\text{T}}$ with a constant altitude H.[1] The K AGVs are indexed by $\mathcal{K} = \{1, 2, \ldots, K\}$, and the location of k-th AGV is represented as $\mathbf{c}_k[n] = [x_k[n], y_k[n]]^{\text{T}}, k \in \mathcal{K}$, assuming zero altitude. To ensure reliable and dynamic trajectory tracking, the drone, functioning as a remote controller, generates optimal control actions based on the AGVs' current states and transmits the corresponding commands to the AGVs over the wireless channel [24].[2] To facilitate system design, we consider a total duration T divided into N discrete time slots, each with a duration $\Delta\tau = T/N$. Moreover, $\Delta\tau$ is chosen to be sufficiently small to ensure that the states of the AGVs remain approximately constant during each time slot.

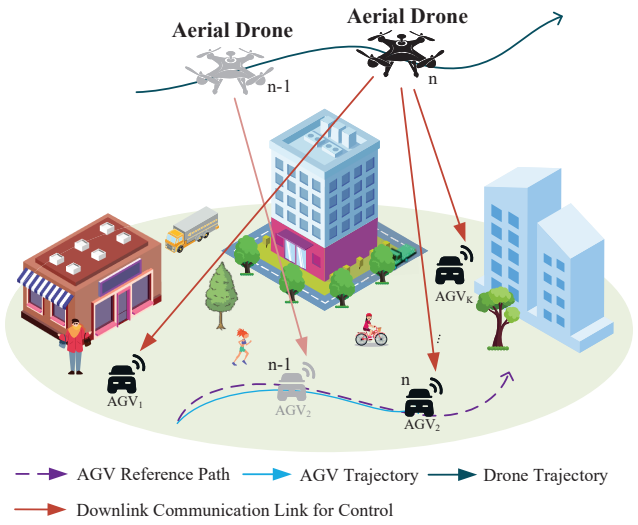


Fig. 1. The considered LAWN comprises a mobile drone enabling wireless control for AGV path tracking.

A. Communication Model

At each time slot, the drone transmits the control commands to the AGVs over the frequency orthogonal channel. Denote

[1]The drone altitude is fixed to ensure safe operation by preventing potential collisions with obstacles. Moreover, in the absence of specific altitude constraints, a lower altitude naturally leads to improved channel quality, thereby eliminating the need for further altitude optimization [19].

[2]In this study, we assume ideal uplink channels to enable perfect state estimation of all AGVs, focusing on the impact of downlink communication on control performance, which has also been considered in prior studies such as [23], [32].

the real-time distance between the drone and the k-th AGV as

$$d_k[n] = \sqrt{\| \mathbf{q}[n] - \mathbf{c}_k[n] \|^2 + H^2}. \quad (1)$$

In the considered system, the air-to-ground channel is dominated by a probabilistic LoS model, which accounts for both LoS and non-line-of-sight (NLoS) links, as commonly adopted in the literature, e.g., [33], [34]. Therefore, the wireless channel between the drone and the k-th AGV is modeled as

$$h_k[n] = P_{k,\text{LoS}}[n] \sqrt{\frac{\alpha_0}{d_k^2[n]}} + (1 - P_{k,\text{LoS}}[n]) \sqrt{\frac{\alpha_0}{d_k^2[n]}} X_{\text{NLoS}}, \quad (2)$$

where

$$P_{k,\text{LoS}}[n] = \frac{1}{1 + a_{\text{LoS}} \cdot \exp\left(-b_{\text{LoS}}\left(\theta_k[n] - a_{\text{LoS}}\right)\right)}, \quad (3)$$

denotes the LoS probability with a_{LoS} and b_{LoS} being the model parameters, and $\theta_k[n]$ representing the elevation angle, given by

$$\theta_k[n] = \arctan\left(\frac{H}{\| \mathbf{q}[n] - \mathbf{c}_k[n] \|}\right). \quad (4)$$

Moreover, α_0 denotes the channel gain at the reference distance $d_k[n] = 1$ m and $X_{\text{NLoS}} \sim \mathcal{CN}(0,1)$ represents the small-scale fading [35]. Accordingly, the SNR at the k-th AGV is denoted as

$$\Gamma_k[n] = \frac{|h_k[n]|^2 p_k[n]}{\sigma_k^2}, \quad (5)$$

where $p_k[n]$ denotes the transmit power allocated to the k-th AGV, and σ_k^2 represents the noise power at the receiver. Furthermore, the achievable communication rate from the drone to the k-th AGV is $R_k[n] = B \log_2(1 + \Gamma_k[n])$ with B denoting the sub-channel bandwidth. In order to achieve delay-aware control tasks in LAWN systems, this work adopts FBL transmission to meet stringent latency requirements. The achievable communication rate under FBL depends on the SNR, the BLER, and the blocklength. Consequently, the data throughput between the drone and the k-th AGV is given by [26]

$$R_k[n] = B\left(\log_2(1 + \Gamma_k[n]) - \sqrt{\frac{V_k[n]}{l_k}} Q^{-1}(\epsilon)\right), \quad (6)$$

where l_k denotes the blocklength of the transmitted signal, ϵ represents the BLER, and $V_k[n]$ is the channel dispersion, which is defined as

$$V_k[n] = 1 - (1 + \Gamma_k[n])^{-2}. \quad (7)$$

B. Control Model

As for the control part, we model the aerial drone and each AGV as a linear control system. The discrete-time dynamics of the k-th control system are described by[3]

$$\mathbf{x}_k[n+1] = \mathbf{A}_k \mathbf{x}_k[n] + \mathbf{B}_k \mathbf{u}_k[n], \quad (8)$$

[3]For analytical tractability, we neglect the control system noise at the AGVs in this study, as our primary focus is to evaluate the impact of wireless communication on control performance [36].

where $\mathbf{x}_k[n] = [x_k[n], y_k[n], v_{x,k}[n], v_{y,k}[n]]^{\mathrm{T}} \in \mathbb{R}^4$ denotes the system state vector of the k-th AGV with $[v_{x,k}[n], v_{y,k}[n]]$ representing the AGV's velocity during the time interval $(n\Delta\tau, (n+1)\Delta\tau]$ and $\mathbf{u}_k[n] = [a_{x,k}[n], a_{y,k}[n]]^{\mathrm{T}} \in \mathbb{R}^2$ denotes the control input vector at time slot n, i.e., the acceleration vector of the k-th AGV. $\mathbf{A}_k \in \mathbb{R}^{4\times4}$ and $\mathbf{B}_k \in \mathbb{R}^{4\times2}$ represent the fixed state transition matrix and input matrix, respectively, defined as

$$\mathbf{A}_k = \begin{bmatrix} 1 & 0 & \Delta\tau & 0 \\ 0 & 1 & 0 & \Delta\tau \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{B}_k = \begin{bmatrix} \frac{\Delta\tau^2}{2} & 0 \\ 0 & \frac{\Delta\tau^2}{2} \\ \Delta\tau & 0 \\ 0 & \Delta\tau \end{bmatrix}. \quad (9)$$

To incorporate the mechanical constraints of AGVs [32], the original system state equation (8) is reformulated by introducing the acceleration increment $\Delta\mathbf{u}_k[n] = \mathbf{u}_k[n] - \mathbf{u}_k[n-1]$ as the new control input, while the original state $\mathbf{x}_k[n]$ together with the previous control input $\mathbf{u}_k[n-1]$ are stacked into an augmented state vector, i.e.,

$$\tilde{\mathbf{x}}_k[n] = \begin{bmatrix} \mathbf{x}_k[n] \\ \mathbf{u}_k[n-1] \end{bmatrix}. \quad (10)$$

The resulting augmented discrete-time system evolves as

$$\tilde{\mathbf{x}}_k[n+1] = \tilde{\mathbf{A}}_k \tilde{\mathbf{x}}_k[n] + \tilde{\mathbf{B}}_k \Delta\mathbf{u}_k[n], \quad (11)$$

where $\tilde{\mathbf{A}}_k \in \mathbb{R}^{6\times6}$ and $\tilde{\mathbf{B}}_k \in \mathbb{R}^{6\times2}$ represent the updated state transition and input matrices, respectively, and are defined as

$$\tilde{\mathbf{A}}_k = \begin{bmatrix} \mathbf{A}_k & \mathbf{B}_k \\ \mathbf{0}_{2\times4} & \mathbf{I}_{2\times2} \end{bmatrix}, \tilde{\mathbf{B}}_k = \begin{bmatrix} \mathbf{B}_k \\ \mathbf{I}_{2\times2} \end{bmatrix}. \quad (12)$$

III. Problem Formulation

In this section, we first present a general MPC framework that aims to minimize the control cost over a finite horizon subject to communication-induced constraints, followed by extending this formulation to account for the impact of transmission outage on control effectiveness.

A. General Framework

To ensure precise trajectory tracking under dynamic conditions, this work adopts an MPC framework. The goal of the MPC framework is to generate an optimal sequence of control inputs that minimizes the deviation between predicted system states and reference trajectories within a finite prediction horizon of length N_p. The associated control cost function is given by (13) [32], as shown at the top of the next page, where

$$\mathbf{e}_k[n+i \mid n] = \tilde{\mathbf{x}}_k[n+i \mid n] - \tilde{\mathbf{x}}_{k,\text{ref}}[n+i], \quad (14)$$

denotes the predicted deviation from the reference state for the k-th AGV at time step $n+i$ and $\Delta\mathbf{u}_k[n+i \mid n]$ represents the predicted control input. $\mathbf{Q}_1 \in \mathbb{R}^{6\times6}$ and $\mathbf{Q}_2 \in \mathbb{R}^{2\times2}$ are positive semi-definite weighting matrices used to balance the trade-off between tracking accuracy and control energy consumption. The term $\mathbf{e}_k^{\mathrm{T}}[n+N_p \mid n]\mathbf{Q}_3\mathbf{e}_k[n+N_p \mid n]$ is introduced to enhance long-term control performance, with

$$J_n = \sum_{k\in\mathcal{K}} \sum_{i=0}^{N_p-1} \mathbf{e}_k^{\mathrm{T}}[n+i\mid n]\,\mathbf{Q}_1\mathbf{e}_k[n+i\mid n] + \Delta\mathbf{u}_k^{\mathrm{T}}[n+i\mid n]\,\mathbf{Q}_2\Delta\mathbf{u}_k[n+i\mid n] + \mathbf{e}_k^{\mathrm{T}}[n+N_p\mid n]\,\mathbf{Q}_3\mathbf{e}_k[n+N_p\mid n]. \quad (13)$$

$\mathbf{Q}_3 \in \mathbb{R}^{6\times 6}$ being the solution of a discrete Riccati equation [14]

$$\begin{aligned}\mathbf{Q}_3 = \mathbf{Q}_1 + \tilde{\mathbf{A}}_k^{\mathrm{T}}\mathbf{Q}_3\tilde{\mathbf{A}}_k - \tilde{\mathbf{A}}_k^{\mathrm{T}}\mathbf{Q}_3\tilde{\mathbf{B}}_k \\ \times\left(\mathbf{Q}_2 + \tilde{\mathbf{B}}_k^{\mathrm{T}}\mathbf{Q}_3\tilde{\mathbf{B}}_k\right)^{-1}\tilde{\mathbf{B}}_k^{\mathrm{T}}\mathbf{Q}_3\tilde{\mathbf{A}}_k.\end{aligned} \quad (15)$$

To facilitate system design, all predictive control input increments for the k-th AGV over the prediction horizon are organized into a vector $\Delta\boldsymbol{\mu}_k[n] \in \mathbb{R}^{2N_p}$, which compactly represents its acceleration decisions at each future time step and is defined as

$$\Delta\boldsymbol{\mu}_k[n] = \begin{bmatrix} \Delta\mathbf{u}_k[n\mid n] \\ \Delta\mathbf{u}_k[n+1\mid n] \\ \vdots \\ \Delta\mathbf{u}_k[n+N_p-1\mid n]\end{bmatrix}. \quad (16)$$

In the considered scenario, control commands are transmitted over wireless links, and thus, communication reliability has a significant impact on overall control performance. To compromise, the control input optimization must explicitly incorporate communication quality-of-service (QoS) constraints, such that the general form of the optimization problem can be formulated as

$$\begin{aligned}\min_{\{\Delta\boldsymbol{\mu}_k[n]\},\mathbf{p}[n],\mathbf{q}[n]} \quad & J_n && (17a)\\ \text{s.t.} \quad & \Delta\boldsymbol{\mu}_k[n]\in\boldsymbol{\mathcal{U}}_k, \forall k\in\mathcal{K}, && (17b)\\ & f_{\mathrm{QoS}}\in\mathcal{C}, && (17c)\end{aligned}$$

where $\mathbf{p}[n] = [p_1[n],\ldots,p_K[n]]^{\mathrm{T}}$ denotes the power allocation vector, $\boldsymbol{\mathcal{U}}_k$ is the predefined control set corresponding to the k-th AGV, and f_{QoS} represents a communication quality metric that must meet $\mathcal{C}$ to ensure the required QoS performance.

B. Outage Probability-Based Formulation

To explicitly characterize how communication reliability influences control, we model the real-time transmission reliability in terms of outage probability. A communication outage is defined as when the real-time communication rate $R_k[n]$ falls below the QoS threshold, preventing the successful delivery of control commands to the AGVs. As a result, during the outage periods, the AGV system evolves based solely on its intrinsic physical dynamics in the absence of external control input. Additionally, assuming ideal channel state information (CSI) and perfect time-frequency synchronization at the AGVs, the outage probability of drone-to-AGV transmission is given by

$$P_k[n] = \Pr\left(R_k[n] < R_k^{\mathrm{th}}\right), \quad (18)$$

where R_k^{th} denotes the minimum required data rate to satisfy the QoS constraint for the k-th AGV. To account for the influence of wireless communication, the discrete-time control system is accordingly reformulated as

$$\tilde{\mathbf{x}}_k[n+1] = \begin{cases}\tilde{\mathbf{A}}_k\tilde{\mathbf{x}}_k[n] + \tilde{\mathbf{B}}_k\Delta\mathbf{u}_k[n] & \text{with a prob. of } (1-P_k[n]),\\ \tilde{\mathbf{A}}_k\tilde{\mathbf{x}}_k[n] & \text{with a prob. of } P_k[n].\end{cases} \quad (19)$$

Accordingly, the expected system state at the next time instant is formulated as

$$\mathbb{E}[\tilde{\mathbf{x}}_k[n+1]] = \tilde{\mathbf{A}}_k\tilde{\mathbf{x}}_k[n] + (1-P_k[n])\,\tilde{\mathbf{B}}_k\Delta\mathbf{u}_k[n]. \quad (20)$$

Substituting the expected dynamics into the original cost yields a modified objective that incorporates the impact of transmission unreliability. The resultant optimization problem is formulated as

$$\begin{aligned}\min_{\{\Delta\boldsymbol{\mu}_k[n]\},\mathbf{p}[n],\mathbf{q}[n]} \quad & \mathbb{E}(J_n) && (21a)\\ \text{s.t.} \quad & \sum_{k\in\mathcal{K}} p_k[n] \le P_{\max}, && (21b)\\ & p_k[n]\ge 0, \forall k\in\mathcal{K}, && (21c)\\ & \Delta\boldsymbol{\mu}_k[n]\in\boldsymbol{\mathcal{U}}_k, \forall k\in\mathcal{K}, && (21d)\\ & \parallel \mathbf{q}[n]-\mathbf{q}[n-1]\parallel \le V_{\max}\Delta\tau, && (21e)\\ & \mathbf{q}[n]\in\mathcal{D}, && (21f)\\ & (20), && (21g)\end{aligned}$$

where constraints (21b) and (21c) indicate the transmit power budget, with $P_{\max}$ being the maximum transmit power of the network. Constraints (21e) and (21f) represent the drone's maximum flight speed and flight boundary, with $V_{\max}$ and $\mathcal{D}$ denoting the maximum allowable speed and the feasible flight region, respectively. Solving the optimization problem yields the optimal power allocation and drone trajectory at time step n, along with control inputs over the prediction horizon. However, it is worth mentioning that only the control input for the current slot is adopted. It is evident that problem (21) is non-convex with coupled variables, making it difficult to achieve a globally optimal solution. To address this challenge, we develop an efficient iterative AO algorithm in the sequel.

IV. Proposed Solutions

In this section, we first derive the closed-form expression of the outage probability function related to the transmit power and drone trajectory. Then, we reformulated the problem (21) into a QP problem, which is next decomposed into three subproblems and solved using an AO framework.

Proposition 1. *The closed-form expression for $P_k[n]$ is given as*

$$P_k[n] = 1 - \exp\left(\frac{d_k^2[n]\,\sigma_k^2\Gamma_k^{\mathrm{th}} - \alpha_0 P_{k,\mathrm{LoS}}^2[n]}{\alpha_0\left(1-P_{k,\mathrm{LoS}}[n]\right)^2 p_k[n]}\right), \quad (22)$$

$$
\begin{aligned}
\tilde{\mathbf{x}}_k\left[n \mid n\right] &= \tilde{\mathbf{x}}_k\left[n\right], \\
\mathbb{E}\left[\tilde{\mathbf{x}}_k\left[n+1 \mid n\right]\right] &= \tilde{\mathbf{A}}_k \tilde{\mathbf{x}}_k\left[n\right] + \left(1-P_k\left[n\right]\right)\tilde{\mathbf{B}}_k \Delta \mathbf{u}_k\left[n \mid n\right], \\
\mathbb{E}\left[\tilde{\mathbf{x}}_k\left[n+2 \mid n\right]\right] &= \tilde{\mathbf{A}}_k \mathbb{E}\left[\tilde{\mathbf{x}}_k\left[n+1 \mid n\right]\right] + \left(1-P_k\left[n\right]\right)\tilde{\mathbf{B}}_k \Delta \mathbf{u}_k\left[n+1 \mid n\right], \\
&= \tilde{\mathbf{A}}_k^2 \tilde{\mathbf{x}}_k\left[n\right] + \tilde{\mathbf{A}}_k\left(1-P_k\left[n\right]\right)\tilde{\mathbf{B}}_k \Delta \mathbf{u}_k\left[n \mid n\right] + \tilde{\mathbf{B}}_k\left(1-P_k\left[n\right]\right)\Delta \mathbf{u}_k\left[n+1 \mid n\right], \\
&\vdots \\
\mathbb{E}\left[\tilde{\mathbf{x}}_k\left[n+N_p \mid n\right]\right] &= \tilde{\mathbf{A}}_k^{N_p} \tilde{\mathbf{x}}_k\left[n\right] + \tilde{\mathbf{A}}_k^{N_p-1}\left(1-P_k\left[n\right]\right)\tilde{\mathbf{B}}_k \Delta \mathbf{u}_k\left[n \mid n\right] + \ldots + \left(1-P_k\left[n\right]\right)\tilde{\mathbf{B}}_k \Delta \mathbf{u}_k\left[n+N_p-1 \mid n\right].
\end{aligned} \tag{23}
$$

$$
\hat{J}_n = \sum_{k\in\mathcal{K}} \mathbb{E}\left[\left(\boldsymbol{\chi}_k\left[n\right]-\boldsymbol{\chi}_{k,\text{ref}}\left[n\right]\right)^{\mathrm{T}} \mathbf{Q}_4 \left(\boldsymbol{\chi}_k\left[n\right]-\boldsymbol{\chi}_{k,\text{ref}}\left[n\right]\right) + \Delta\boldsymbol{\mu}_k\left[n\right]^{\mathrm{T}} \mathbf{Q}_5 \Delta\boldsymbol{\mu}_k\left[n\right]\right]. \tag{28}
$$

where $\Gamma_k^{\text{th}} > 0$ denotes the minimum SNR threshold required to achieve the target rate R_k^{th} for the k-th AGV.

Proof: See Appendix A. ∎

To facilitate tractable optimization, we proceed by transforming problem (21) into an equivalent QP formulation. Given the system state $\tilde{\mathbf{x}}_k[n]$, the predicted control states over the horizon are expressed as (23), as shown at the top of this page. Next, we introduce the predicted control state vector

$$
\boldsymbol{\chi}_k\left[n\right] = \begin{bmatrix} \tilde{\mathbf{x}}_k\left[n \mid n\right] \\ \tilde{\mathbf{x}}_k\left[n+1 \mid n\right] \\ \vdots \\ \tilde{\mathbf{x}}_k\left[n+N_p \mid n\right] \end{bmatrix} \in \mathbb{R}^{6N_p+6}. \tag{24}
$$

Accordingly, (23) can be rewritten in a matrix form as

$$
\mathbb{E}\left[\boldsymbol{\chi}_k\left[n\right]\right] = \mathbf{C}_k \tilde{\mathbf{x}}_k\left[n\right] + \left(1-P_k\left[n\right]\right)\mathbf{F}_k \Delta\boldsymbol{\mu}_k\left[n\right], \tag{25}
$$

where $\mathbf{C}_k \in \mathbb{R}^{(6N_p+6)\times 6}$ and $\mathbf{F}_k \in \mathbb{R}^{(6N_p+6)\times 2N_p}$ denote the augmented state transition and input matrices, respectively, with detailed expressions provided as

$$
\mathbf{C}_k = \begin{bmatrix} \mathbf{I}_{6\times 6} \\ \tilde{\mathbf{A}}_k \\ \vdots \\ \tilde{\mathbf{A}}_k^{N_p} \end{bmatrix}, \mathbf{F}_k = \begin{bmatrix} 0 & \cdots & 0 \\ \tilde{\mathbf{B}}_k & \cdots & 0 \\ \tilde{\mathbf{A}}_k\tilde{\mathbf{B}}_k & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \tilde{\mathbf{A}}_k^{N_p-1}\tilde{\mathbf{B}}_k & \cdots & \tilde{\mathbf{B}}_k \end{bmatrix}. \tag{26}
$$

Similarly, we define that

$$
\boldsymbol{\chi}_{k,\text{ref}}\left[n\right] = \begin{bmatrix} \tilde{\mathbf{x}}_{k,\text{ref}}\left[n\right] \\ \tilde{\mathbf{x}}_{k,\text{ref}}\left[n+1\right] \\ \vdots \\ \tilde{\mathbf{x}}_{k,\text{ref}}\left[n+N_p\right] \end{bmatrix} \in \mathbb{R}^{6N_p+6}, \tag{27}
$$

represents the reference state trajectory for the k-th AGV over the prediction horizon. The cost function in problem (21) is equivalently rewritten in the matrix form as in (28), shown at the top of this page, where $\mathbf{Q}_4 = \text{diag}\left(\mathbf{Q}_1, \ldots, \mathbf{Q}_1, \mathbf{Q}_3\right)$ and $\mathbf{Q}_5 = \text{diag}\left(\mathbf{Q}_2, \ldots, \mathbf{Q}_2\right)$. By substituting (25) into (28), the cost function is rewritten as

$$
\hat{J}_n = \sum_{k\in\mathcal{K}} \Delta\boldsymbol{\mu}_k^{\mathrm{T}}\left[n\right]\mathbf{G}_k\left[n\right]\Delta\boldsymbol{\mu}_k\left[n\right] + \mathbf{H}_k\left[n\right]\Delta\boldsymbol{\mu}_k\left[n\right] + M_k\left[n\right], \tag{29}
$$

where $\mathbf{G}_k\left[n\right]$, $\mathbf{H}_k\left[n\right]$, $M_k\left[n\right]$ are constant coefficient, which are calculated as

$$
\mathbf{G}_k\left[n\right] = \left(1-P_k\left[n\right]\right)^2 \mathbf{F}_k^{\mathrm{T}} \mathbf{Q}_4 \mathbf{F}_k + \mathbf{Q}_5, \tag{30}
$$

$$
\mathbf{H}_k\left[n\right] = 2\left(1-P_k\left[n\right]\right)\left(\tilde{\mathbf{x}}_k\left[n\right]^{\mathrm{T}} \mathbf{C}_k^{\mathrm{T}} - \boldsymbol{\chi}_{k,\text{ref}}^{\mathrm{T}}\left[n\right]\right)\mathbf{Q}_4 \mathbf{F}_k, \tag{31}
$$

$$
\begin{aligned}
M_k\left[n\right] =& \tilde{\mathbf{x}}_k^{\mathrm{T}}\left[n\right]\mathbf{C}_k^{\mathrm{T}}\mathbf{Q}_4\mathbf{C}_k\tilde{\mathbf{x}}_k\left[n\right] - 2\tilde{\mathbf{x}}_k^{\mathrm{T}}\left[n\right]\mathbf{C}_k^{\mathrm{T}}\mathbf{Q}_4\boldsymbol{\chi}_{k,\text{ref}}\left[n\right] \\
&+ \boldsymbol{\chi}_{k,\text{ref}}^{\mathrm{T}}\left[n\right]\mathbf{Q}_4\boldsymbol{\chi}_{k,\text{ref}}\left[n\right].
\end{aligned} \tag{32}
$$

Consequently, problem (21) is reformulated as

$$
\begin{aligned}
\min_{\{\Delta\boldsymbol{\mu}_k[n]\},\mathbf{p}[n],\mathbf{q}[n]} \quad & \hat{J}_n\left(\{\Delta\boldsymbol{\mu}_k\left[n\right]\}, \mathbf{p}\left[n\right], \mathbf{q}\left[n\right]\right) && \text{(33a)} \\
\text{s.t.} \quad & \text{(21b), (21c), (21d), (21e), (21f).} && \text{(33b)}
\end{aligned}
$$

Although problem (33) is cast as a QP problem, it remains challenging to solve due to the interdependence among control inputs, power allocation, and drone trajectories. To overcome this challenge, we decompose problem (33) into three sub-problems: AGV path tracking, power allocation, and drone trajectory planning. These sub-problems are then solved in an alternating manner using the AO method.

A. Control Strategy Design

In this subsection, we focus on the control strategy design for AGV path tracking, assuming that the power allocation and drone trajectory are predetermined. The corresponding optimization problem is formulated as

$$
\begin{aligned}
\min_{\{\Delta\boldsymbol{\mu}_k[n]\}} \quad & \hat{J}_n\left(\{\Delta\boldsymbol{\mu}_k\left[n\right]\}\right) && \text{(34a)} \\
\text{s.t.} \quad & \text{(21d).} && \text{(34b)}
\end{aligned}
$$

Given the objective function $\hat{J}_n\left(\{\Delta\boldsymbol{\mu}_k\left[n\right]\}\right)$ is quadratic and involves a positive definite weighting matrix $\mathbf{G}_k\left[n\right]$, problem (34) can be solved as a standard QP problem using convex optimization techniques [37], [38]. Furthermore, since only the control policy of the current interval is selected and transmitted to AGVs, the optimal control input $\Delta\mathbf{u}_k^{\star}\left[n\right]$ is obtained as the first two entries of the optimized control sequence $\Delta\boldsymbol{\mu}_k^{\star}\left[n\right]$.

B. Optimization of Power Allocation

With the given drone trajectory and the updated control action, problem (33) is equivalently written as

$$\min_{\mathbf{p}[n]} \quad \hat{J}_n\left(\mathbf{p}\left[n\right]\right) \tag{35a}$$

$$\text{s.t.} \quad \text{(21b)}, \text{(21c)}, \tag{35b}$$

where $\hat{J}_n\left(\mathbf{p}\left[n\right]\right)$ denotes the control cost related to the power allocation variable expressed as

$$\begin{aligned}\hat{J}_n\left(\mathbf{p}\left[n\right]\right) = &\sum_{k\in\mathcal{K}} L_k\left[n\right]\left(g\left(p_k\left[n\right]\right)\right)^2 \\ &+ 2S_k\left[n\right] g\left(p_k\left[n\right]\right) + W_k\left[n\right],\end{aligned} \tag{36}$$

with

$$g\left(p_k\left[n\right]\right) = \exp\left(\frac{d_k^2\left[n\right]\sigma_k^2\Gamma_k^{\text{th}} - \alpha_0 P_{k,\text{LoS}}^2\left[n\right]}{\alpha_0\left(1-P_{k,\text{LoS}}\left[n\right]\right)^2 p_k\left[n\right]}\right), \tag{37}$$

$$L_k\left[n\right] = \Delta\boldsymbol{\mu}_k^{\text{T}}\left[n\right]\mathbf{F}_k^{\text{T}}\mathbf{Q}_4\mathbf{F}_k\Delta\boldsymbol{\mu}_k\left[n\right], \tag{38}$$

$$S_k\left[n\right] = \left(\tilde{\mathbf{x}}_k\left[n\right]^{\text{T}}\mathbf{C}_k^{\text{T}} - \boldsymbol{\chi}_{k,\text{ref}}^{\text{T}}\left[n\right]\right)\mathbf{Q}_4\mathbf{F}_k\Delta\boldsymbol{\mu}_k\left[n\right], \tag{39}$$

$$W_k\left[n\right] = \Delta\boldsymbol{\mu}_k^{\text{T}}\left[n\right]\mathbf{Q}_5\Delta\boldsymbol{\mu}_k\left[n\right] + M_k[n]. \tag{40}$$

Since problem (35) is a linearly constrained optimization problem with a non-convex objective function, it can be solved through the PGD method [39]. The detailed procedure of the PGD-based algorithm for power allocation is summarized in Algorithm 1. In Algorithm 1, the process starts by initializing a feasible starting point, assuming equal power allocation among all AGVs. The initial step size ρ, Armijo parameter β, backtracking coefficient $\hat{\rho}$, and convergence tolerance ε are also specified. At each iteration, the search direction is determined by computing the gradient of $\hat{J}_n\left(\mathbf{p}\left[n\right]\right)$ with respect to $\mathbf{p}\left[n\right]$, i.e.,

$$\boldsymbol{\nabla}_{\mathbf{p}[n]}\hat{J}_n\left(\mathbf{p}\left[n\right]\right) = \left[\frac{\partial\hat{J}_n\left(\mathbf{p}\left[n\right]\right)}{\partial p_1\left[n\right]},\ldots,\frac{\partial\hat{J}_n\left(\mathbf{p}\left[n\right]\right)}{\partial p_K\left[n\right]}\right]^{\text{T}}, \tag{41}$$

In (41), $\frac{\partial\hat{J}_n(\mathbf{p}[n])}{\partial p_k[n]}$ is expressed as

$$\begin{aligned}\frac{\partial\hat{J}_n\left(\mathbf{p}\left[n\right]\right)}{\partial p_k\left[n\right]} =& 2L_k\left[n\right]g\left(p_k\left[n\right]\right) \\ &\times\frac{\partial g\left(p_k\left[n\right]\right)}{\partial p_k\left[n\right]} + 2S_k\left[n\right]\frac{\partial g\left(p_k\left[n\right]\right)}{\partial p_k\left[n\right]},\end{aligned} \tag{42}$$

where $\frac{\partial g(p_k[n])}{\partial p_k[n]}$ is particularized as

$$\begin{aligned}\frac{\partial g\left(p_k\left[n\right]\right)}{\partial p_k\left[n\right]} =& \frac{d_k^2\left[n\right]\sigma_k^2\Gamma_k^{\text{th}} - \alpha_0 P_{k,\text{LoS}}^2\left[n\right]}{\alpha_0\left(1-P_{k,\text{LoS}}\left[n\right]\right)^2\left(p_k\left[n\right]\right)^2} \\ &\times\exp\left(\frac{d_k^2\left[n\right]\sigma_k^2\Gamma_k^{\text{th}} - \alpha_0 P_{k,\text{LoS}}^2\left[n\right]}{\alpha_0\left(1-P_{k,\text{LoS}}\left[n\right]\right)^2 p_k\left[n\right]}\right).\end{aligned} \tag{43}$$

Then, $\mathbf{p}\left[n\right]$ is updated by projecting onto the feasible domain, i.e, $\Psi_{\mathcal{P}}\left(\mathbf{p}'\left[n\right]-\rho\boldsymbol{\Delta}\right) : \mathbb{R}^K \rightarrow \mathcal{P}$, where $\mathcal{P}$ denotes the feasible set defined as

$$\mathcal{P} = \left\{\mathbf{p}\left[n\right] \mid \sum_{k\in\mathcal{K}} p_k\left[n\right] \le P_{\max}, p_k\left[n\right] \ge 0\right\}, \tag{44}$$

Algorithm 1 PGD-Based Algorithm for Power Allocation

1: **Initialize** Starting feasible point $\mathbf{p}\left[n\right] \leftarrow \left[\frac{P_{\max}}{K},\ldots,\frac{P_{\max}}{K}\right]$, initial step size ρ, Armijo parameter β, backtracking coefficient $\hat{\rho}$, and tolerance ε.
2: **repeat**
3: Save the previous direction vector $\mathbf{p}'\left[n\right] \leftarrow \mathbf{p}\left[n\right]$.
4: Determine a search direction $\boldsymbol{\Delta} \triangleq \boldsymbol{\nabla}_{\mathbf{p}[n]}\hat{J}_n\left(\mathbf{p}\left[n\right]\right)$.
5: **repeat**
5: Choose a step size $\rho \leftarrow \frac{\rho}{1+\hat{\rho}}$.
6: **until** $\hat{J}_n\left(\Psi_{\mathcal{P}}\left(\mathbf{p}'[n]-\rho\boldsymbol{\Delta}\right)\right) \le \hat{J}_n\left(\mathbf{p}'[n]\right) - \beta\rho\left\|\boldsymbol{\Delta}\right\|^2$.
6: Update $\mathbf{p}\left[n\right] \leftarrow \Psi_{\mathcal{P}}\left(\mathbf{p}'\left[n\right]-\rho\boldsymbol{\Delta}\right)$.
7: **until** $\left\|\mathbf{p}\left[n\right]-\mathbf{p}'\left[n\right]\right\| < \varepsilon$
8: **Output** $\mathbf{p}^\star\left[n\right] \leftarrow \mathbf{p}\left[n\right]$.

and $\Psi_{\mathcal{P}}\left(\mathbf{p}'\left[n\right]-\rho\boldsymbol{\Delta}\right)$ is obtained via solving problem

$$\min_{\mathbf{p}[n]\in\mathcal{P}} \quad \left\|\mathbf{p}'\left[n\right]-\rho\boldsymbol{\Delta}-\mathbf{p}\left[n\right]\right\|. \tag{45a}$$

The step size ρ is further adaptively selected via a backtracking line search to ensure sufficient descent [40]. Specifically, starting from an initial value, ρ is iteratively reduced by a factor of $1/(1+\hat{\rho})$ until the following Armijo-type condition is satisfied

$$\hat{J}_n\left(\Psi_{\mathcal{P}}\left(\mathbf{p}'[n]-\rho\boldsymbol{\Delta}\right)\right) \le \hat{J}_n\left(\mathbf{p}'[n]\right) - \beta\rho\left\|\boldsymbol{\Delta}\right\|^2. \tag{46}$$

C. Drone Trajectory Design and Overall Algorithm

In this subsection, we focus on optimizing the drone trajectory given the control strategy and power allocation obtained from previous steps. The corresponding optimization is reduced to

$$\min_{\mathbf{q}[n]} \quad \hat{J}_n\left(\mathbf{q}\left[n\right]\right) \tag{47a}$$

$$\text{s.t.} \quad \text{(21e)}, \text{(21f)}. \tag{47b}$$

Due to the non-convexity of the objective function in problem (47), the SCA method is adopted, which linearizes the non-convex terms using first-order Taylor expansions [41]. In particular, $\hat{J}_n\left(\mathbf{q}\left[n\right]\right)$ is approximated as

$$\phi = \hat{J}_n\left(\mathbf{q}^{\iota_1}\left[n\right]\right) + \left[\boldsymbol{\nabla}_{\mathbf{q}[n]}\hat{J}_n\left(\mathbf{q}^{\iota_1}\left[n\right]\right)\right]^{\text{T}}\left(\mathbf{q}\left[n\right]-\mathbf{q}^{\iota_1}\left[n\right]\right), \tag{48}$$

where $\mathbf{q}^{\iota_1}\left[n\right]$ denotes the drone's trajectory at the n-th time slot in the ι_1-th iteration and $\boldsymbol{\nabla}_{\mathbf{q}[n]}\hat{J}_n\left(\mathbf{q}^{\iota_1}\left[n\right]\right)$ is given by (49), as shown at the top of the next page, with $\partial g_k\left(\mathbf{q}^{\iota_1}\left[n\right]\right)/\partial\mathbf{q}^{\iota_1}\left[n\right]$ representing the first-order derivation of function $g_k\left(\mathbf{q}^{\iota_1}\left[n\right]\right)$. As a result, problem (47) is approximated by a sequence of convex optimization problems

$$\min_{\mathbf{q}[n]} \quad \phi \tag{50a}$$

$$\text{s.t.} \quad \text{(21e)}, \text{(21f)}, \tag{50b}$$

which can be solved with standard convex optimization tools such as CVX [42].

Consequently, we propose an AO-based algorithm that alternately optimizes the control strategy, power allocation, and drone trajectory. The pseudocode of the AO-based algorithm

$$\nabla_{\mathbf{q}^{\iota_1}[n]}\hat{J}_n\left(\mathbf{q}^{\iota_1}[n]\right)=\sum_{k\in\mathcal{K}}2L_k[n]\,g_k\left(\mathbf{q}^{\iota_1}[n]\right)\frac{\partial g_k\left(\mathbf{q}^{\iota_1}[n]\right)}{\partial\mathbf{q}^{\iota_1}[n]}+2S_k[n]\frac{\partial g_k\left(\mathbf{q}^{\iota_1}[n]\right)}{\partial\mathbf{q}^{\iota_1}[n]}. \tag{49}$$

$$\begin{aligned}\eta\left(\left\{\Delta\boldsymbol{\mu}_k^{\iota-1}[n]\right\},\mathbf{p}^{\iota-1}[n],\mathbf{q}^{\iota-1}[n]\right)&=\eta_1\left(\left\{\Delta\boldsymbol{\mu}_k^{\iota-1}[n]\right\},\mathbf{p}^{\iota-1}[n],\mathbf{q}^{\iota-1}[n]\right)\overset{1}{\geq}\eta_1\left(\left\{\Delta\boldsymbol{\mu}_k^{\iota}[n]\right\},\mathbf{p}^{\iota-1}[n],\mathbf{q}^{\iota-1}[n]\right)\\&\overset{2}{\geq}\eta_2\left(\left\{\Delta\boldsymbol{\mu}_k^{\iota}[n]\right\},\mathbf{p}^{\iota}[n],\mathbf{q}^{\iota-1}[n]\right)\overset{3}{\geq}\eta_3\left(\left\{\Delta\boldsymbol{\mu}_k^{\iota}[n]\right\},\mathbf{p}^{\iota}[n],\mathbf{q}^{\iota}[n]\right)\\&=\eta\left(\left\{\Delta\boldsymbol{\mu}_k^{\iota}[n]\right\},\mathbf{p}^{\iota}[n],\mathbf{q}^{\iota}[n]\right).\end{aligned} \tag{51}$$

is given in Algorithm 2. Specifically, the algorithm begins by initializing $\mathbf{p}[n]$, $\mathbf{q}[n]$, and $\tilde{\mathbf{x}}_k[n]$, as well as the auxiliary parameters including the iteration index ι, the convergence tolerance ε, and the maximum number of inner iterations $\iota_{\max}$. The optimization variables of problem (33) are then partitioned into three blocks, i.e., $\{\Delta\boldsymbol{\mu}_k[n]\}$, $\mathbf{p}[n]$, and $\mathbf{q}[n]$. We optimize the three partitioned variables alternately by solving problems (34), (35), and (47), respectively, while keeping the other variables fixed. It is important to note that the solutions to problems (35) and (47) are only near-optimal approximations to the original problem. Therefore, it is essential to analyze the convergence of the AO-based problem. For convergence, let $\eta\left(\{\Delta\boldsymbol{\mu}_k[n]\},\mathbf{p}[n],\mathbf{q}[n]\right)$, $\eta_1\left(\{\Delta\boldsymbol{\mu}_k[n]\},\mathbf{p}[n],\mathbf{q}[n]\right)$, $\eta_2\left(\{\Delta\boldsymbol{\mu}_k[n]\},\mathbf{p}[n],\mathbf{q}[n]\right)$, and $\eta_3\left(\{\Delta\boldsymbol{\mu}_k[n]\},\mathbf{p}[n],\mathbf{q}[n]\right)$ denote the value of the objective function of problem (33), (34), (35), and (50), respectively. Accordingly, we have the following relationships (51), as shown at the top of this page, where the inequalities 1, 2, and 3 hold as the solutions obtained by solving these three problems sequentially. Equation (51) shows that the objective function of problem (33) is non-increasing with each iteration. Moreover, as the objective function is lower bounded, Algorithm 2 is guaranteed to converge to a sub-optimal solution.

We further analyze the computational complexity of the proposed AO-based algorithm. Problem (34) is addressed using a standard interior-point method, with a computational complexity of $C_1=\mathcal{O}\left((2N_pK)^{3.5}\right)$ [41]. The PGD-based algorithm incurs complexity mainly from gradient computations of $\hat{J}_n\left(\mathbf{p}[n]\right)$ with respect to $\mathbf{p}[n]$ at each iteration, yielding a total complexity of $C_2=\mathcal{O}\left(\log\left(1/\varepsilon^2\right)K^2\right)$ [43]. The drone trajectory is optimized via the interior-point method combined with the SCA approach, which introduces a complexity of $C_3=\mathcal{O}\left(L_3\left(3M\right)^{3.5}\right)$, where L_3 denotes the number of SCA iterations. Therefore, the overall computational complexity of Algorithm 2 is given $C_{tot}=\mathcal{O}\left(L\left(C_1+C_2+C_3\right)\right)$, with L denoting the total number of alternating iterations.

Algorithm 2 AO-based Algorithm for Solving Original Problem (33)

Initialize $n\leftarrow 1$, $\iota\leftarrow 1$, $\mathbf{p}^{\iota}[n]$, $\mathbf{q}^{\iota}[n]$, $\tilde{\mathbf{x}}_k[n]$, tolerance ε, and maximum number of iterations $\iota_{\max}$.
repeat
  repeat
    $\iota\leftarrow\iota+1$.
    Solve problem (34) with given $\tilde{\mathbf{x}}_k[n]$, $\mathbf{p}^{\iota-1}[n]$, and $\mathbf{q}^{\iota-1}[n]$ to obtain $\{\Delta\boldsymbol{\mu}_k^{\iota}[n]\}$.
    Solve problem (35) using Algoithm 1, $\tilde{\mathbf{x}}_k[n]$, $\{\Delta\boldsymbol{\mu}_k^{\iota}[n]\}$, and $\mathbf{q}^{\iota-1}[n]$ to obtain $\mathbf{p}^{\iota}[n]$.
    Solve problem (50) using SCA, $\tilde{\mathbf{x}}_k[n]$, $\{\Delta\boldsymbol{\mu}_k^{\iota}[n]\}$, and $\mathbf{p}^{\iota}[n]$ to obtain $\mathbf{q}^{\iota}[n]$.
  until The fractional increase of the objective value is below ε or $\iota>\iota_{\max}$.
  Output $\Delta\mathbf{u}_k^{\star}[n]\leftarrow\left(\Delta\boldsymbol{\mu}_k^{\iota}[n]\right)_1$, $\mathbf{p}^{\star}[n]\leftarrow\mathbf{p}^{\iota}[n]$, and $\mathbf{q}^{\star}[n]\leftarrow\mathbf{q}^{\iota}[n]$.
  Use $\tilde{\mathbf{x}}_k[n]$ and $\Delta\mathbf{u}_k^{\star}[n]$ to obtain $\tilde{\mathbf{x}}_k[n+1]$.
  $n\leftarrow n+1$.
until $n>N$.

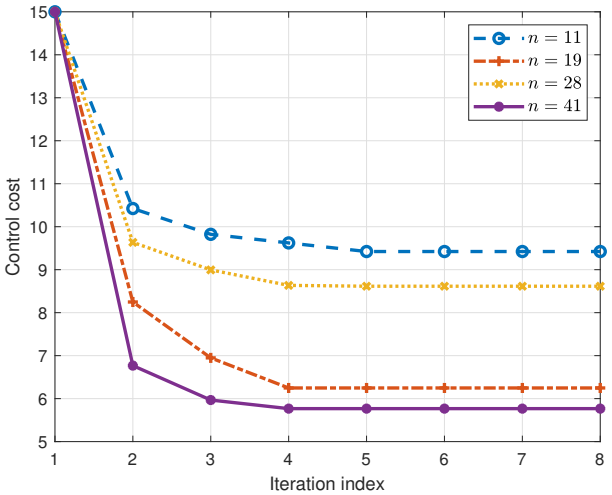


Fig. 2. Convergence of the proposed algorithm at different time slots with $P_{\max}=0$ dBW, $R_k^{\text{th}}=1$Mbps, and $l_k=1024$ bits.

V. Results Discussion

A. Simulation Results

In this subsection, we present simulation results for the proposed algorithm, which jointly optimizes the control strategy, power allocation, and drone trajectory in a LAWN wireless control system for real-time AGV path tracking. The total simulation duration is set to 60 s, divided into 120 discrete time slots. Four AGVs are assumed to move within a predefined

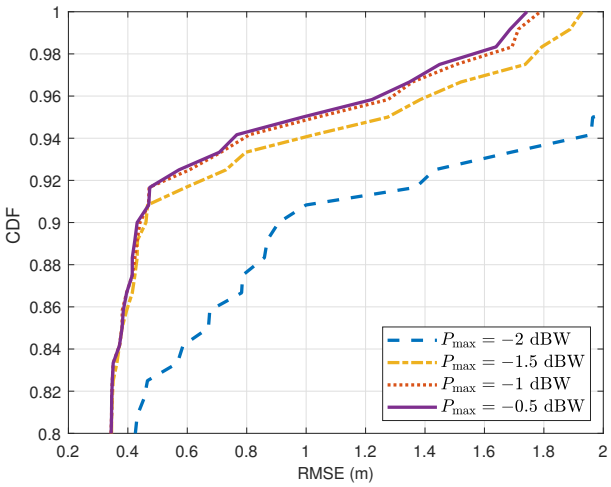


Fig. 3. CDF of tracking performance under different power budgets with $R_k^{\text{th}} = 1$Mbps and $l_k = 1024$ bits.

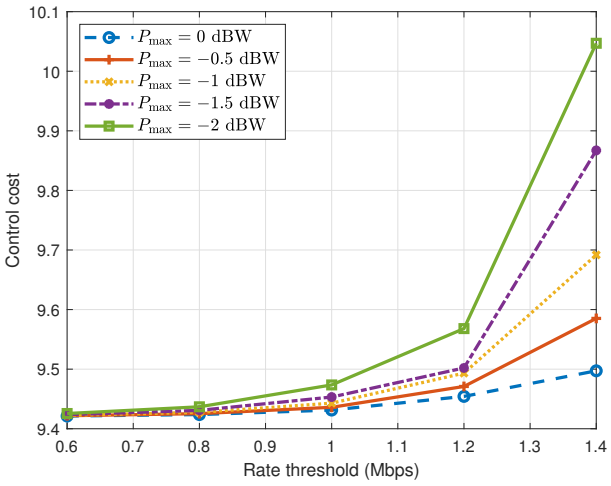


Fig. 4. Control cost versus the rate threshold under different power budgets with $n = 11$ and $l_k = 1024$ bits.

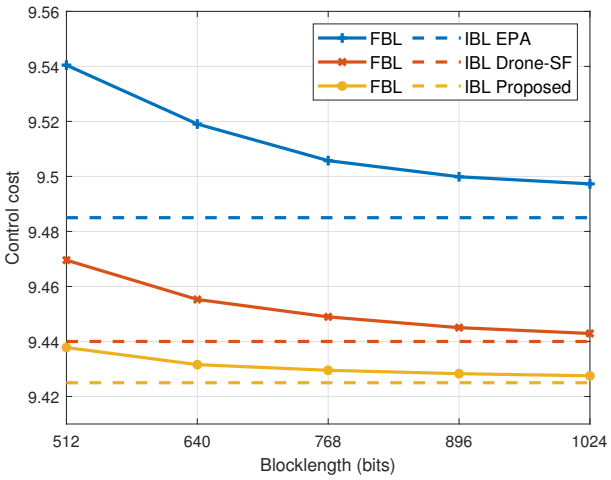


Fig. 5. Control cost versus bloclength of command signals under different scenarios with $n = 11$, $P_{\max} = 0$ dBW, and $R_k^{\text{th}} = 1$ Mbps.

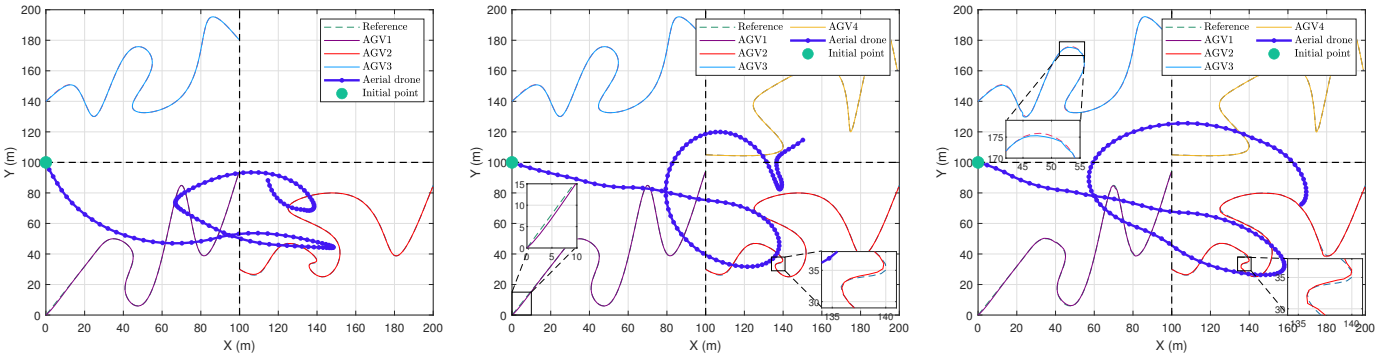
rectangular area of size 200×200 m^2. The drone flies within the same area at a fixed altitude of 50 m. The initial drone position is set as $\mathbf{q}[1] = [0, 100]^{\text{T}}$ m. Regarding the communication parameters, the reference channel gain is set to $\alpha_0 = -50$ dB and the sub-channel bandwidth B is set to 1 MHz. The noise power is configured as $\sigma_k^2 = -100$ dBm and BLER is set to $\epsilon = 10^{-6}$ [44], [45]. For the control settings, the weighting matrices $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are both set to identity matrices, and the prediction horizon is set to $N_p = 10$ [32], [36]. To evaluate the performance of the proposed algorithm, we further introduce two benchmark schemes for comparison. The first is the equal power allocation (EPA) scheme, in which the total power budget $P_{\max}$ is equally distributed among all AGVs without dynamic power adjustment. The second is the drone-straight flight (SF) scheme, where the drone follows a pre-defined straight-line trajectory with a randomly selected constant speed, without trajectory optimization. Throughout the simulations, we analyze the convergence behavior, tracking performance, and drone trajectory under varying power budgets, rate thresholds, and command signal blocklengths.

To evaluate the convergence behavior of the proposed algorithm, Fig. 2 shows the evolution of control cost across iterations for different time slots. Specifically, four representative time slots, i.e., $n = 11, 19, 28, 41$, are selected to comprehensively assess the stability and convergence properties of the algorithm. As shown in the figure, the control cost in all cases rapidly decreases and converges to an optimal value within a few iterations, demonstrating both the rapid convergence rate and the effectiveness of the proposed optimization framework.

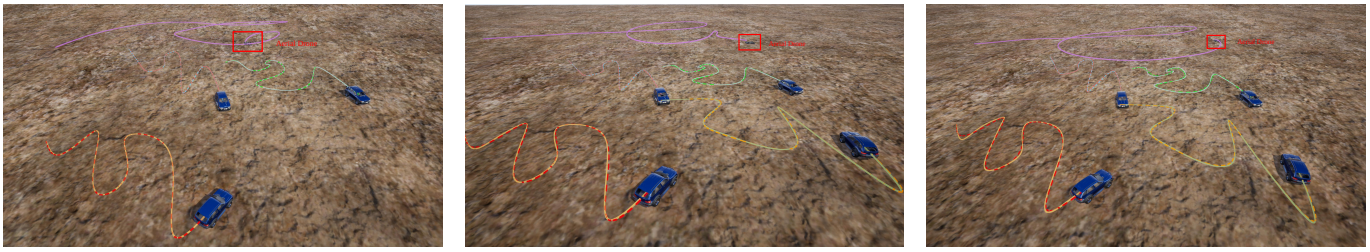
Fig. 3 presents the cumulative distribution function (CDF) of the tracking root mean square error (RMSE) under various transmission power levels. The curves indicate that higher power budgets lead to improved control accuracy, as evidenced by smaller RMSE values and sharper CDF transitions. As the power budget decreases, performance degradation becomes more pronounced, especially in low-reliability scenarios, which underscores the strong interplay between wireless communication quality and control performance in real-time systems. The finding emphasizes that insufficient communication resources can significantly impair control accuracy, highlighting the need to jointly consider communication constraints when designing wireless control systems.

To further investigate the impact of wireless communication on control performance, Fig. 4 shows the control cost at a selected time slot under different rate thresholds and power budgets. As the rate threshold increases, implying more stringent communication requirements, the control cost rises accordingly, which is because a higher rate threshold leads to a higher outage probability under limited power, reducing the reliability of command transmission and ultimately increasing the control cost. Moreover, the increase in control cost is more pronounced under lower power budgets since limited transmission power makes the system more sensitive to rate constraints, resulting in a sharper increase in outage probability and, consequently, higher control cost.

Fig. 5 illustrates the relationship between control cost and the blocklength of command signals under different bench-



(a) Simulation with 3 AGVs, Proposed. **(b)** Simulation with 4 AGVs, Proposed. **(c)** Simulation with 4 AGVs, EPA.



(d) Validation with 3 AGVs, Proposed. **(e)** Validation with 4 AGVs, Proposed. **(f)** Validation with 4 AGVs, EPA.

Fig. 6. The simulation and AirSim-based validation of drone trajectory and AGVs tracking paths under different scenarios.

mark scenarios. As expected, the control cost consistently decreases with increasing blocklength and gradually converges to the IBL case. This trend can be attributed to the improved communication reliability and reduced outage probability enabled by longer transmissions, which in turn enhance the quality of real-time wireless control. It is also evident that the proposed algorithm significantly outperforms the EPA scheme. This performance gain stems from the proposed algorithm's ability to dynamically adjust power allocation according to channel conditions and control demands, while the EPA baseline statically divides total power equally among AGVs, disregarding their heterogeneous needs and communication states. Likewise, the proposed method achieves better control performance than the drone SF scenario, where the drone follows a fixed flight path, limiting its ability to maintain reliable and timely communication.

Fig. 6 compares the drone trajectories and AGV tracking results under three different scenarios. Fig. 6(a) presents the result of the proposed method when only three AGVs are involved. In this case, the overall control task is relatively simple, allowing the drone to maintain a shorter trajectory while providing sufficient wireless support. As a result, the AGVs exhibit high tracking accuracy throughout the entire path, even around sharp turns or directional changes, demonstrating that with reduced task load, the system can allocate resources more effectively and ensure precise trajectory following. Fig. 6(b) corresponds to the proposed joint optimization framework with four AGVs. As the number of AGVs increases, the control complexity becomes higher, especially in regions where multiple AGVs perform sharp turns or frequent directional changes. The drone trajectory is skewed toward the lower region, where AGV2 and AGV3 follow more complex paths, reflecting the algorithm's ability to prioritize areas with greater control demands and dynamically allocate more communication resources where needed. Despite increased complexity, the tracking performance remains satisfactory, although slight deviations occur near the initial stage and in high-curvature regions. In contrast, Fig. 6(c) shows the result under the EPA baseline with four AGVs. Without spatial prioritization, the drone follows a significantly longer trajectory and fails to focus on critical areas with high tracking requirements. Consequently, the tracking performance is notably worse, especially in sharp-turn regions where precise control is needed. The AGVs deviate considerably from their reference paths, indicating limited communication support under uniform power allocation.

B. AirSim Platform-Based Experiments

Table II: AirSim Simulation Parameters

Parameter	Value	Parameter	Value
Rain	20%	Dust	None
Road Wetness	30%	Fog	15%
Snow	20%	Wind Speed x-axis	2 m/s
Road Snow	None	Wind Speed y-axis	2.5 m/s
Falling Leaves	None	Wind Speed z-axis	None

To further validate the effectiveness of the proposed method, we implement the LAWN-assisted wireless control system within the AirSim simulation platform [31], as illustrated in Fig. 6(d)-Fig. 6(f). To better emulate real-world disturbances, we configured the AirSim environment with multiple weather parameters, as summarized in Table II. Specifically, we introduced moderate environmental dynamics by setting the rain

and snow intensities to 20%, road wetness to 30%, and fog to 15%, while disabling dust, road snow, and falling leaves. In addition, wind effects were introduced by configuring directional wind speeds to 2 m/s, 2.5 m/s, and 0 m/s along the X, Y, and Z axes, respectively. In this setup, AGVs are deployed with predefined nonlinear trajectories featuring diverse motion characteristics, including sharp turns and variable speeds. The drone is tasked with providing wireless control support while dynamically adjusting its trajectory and power allocation according to the proposed joint optimization algorithm. Compared to simulation, the AirSim validation exhibits more noticeable tracking deviations, particularly around sharp turns and abrupt motion changes, mainly due to the incorporation of realistic physical constraints such as actuator delays, vehicle dynamics, and limited control frequency. Under these conditions, both the AGVs and the drone encounter greater challenges in maintaining high-precision trajectory following. Nevertheless, the overall trends remain consistent with the simulation results, where the drone still tends to hover around regions of high control complexity and adaptively supports AGVs with greater tracking demands, further demonstrating the robustness and practicality of the proposed method in more physically grounded environments.

VI. Conclusion

This paper has investigated a real-time wireless control framework for drone-assisted LAWN systems, with a particular focus on delay-sensitive AGV trajectory tracking under FBL transmission. We have developed the joint communication-control system, in which we first derived a closed-form expression for the outage probability under FBL transmission, and then we incorporated it into an MPC-based control strategy. Based on this, we have formulated the joint optimization problem to optimize control inputs, power allocation, and drone trajectory, subject to transmit power budgets and drone mobility constraints. We have transformed the inherently non-convex problem into a QP model and solved it via an AO framework, which leverages PGD and SCA techniques. Rigorous analysis was conducted to evaluate the convergence behavior and computational complexity of the proposed solution. Extensive simulation results demonstrate that the proposed algorithm consistently outperforms baseline methods in control performance. Furthermore, its effectiveness is validated through additional experiments conducted on the AirSim platform, confirming its practical applicability in LAWN-enabled IoT scenarios.

Appendix A
Proof of Proposition 1

The first-order derivation of function $R_k\left(\Gamma_k[n]\right)$ is calculated as

$$R_k'\left(\Gamma_k[n]\right) = \frac{1}{1+\Gamma_k[n]} \times \left(1 - \frac{Q^{-1}\left(\epsilon\right)}{\sqrt{l_k}\left(1+\Gamma_k[n]\right)\sqrt{\left(1+\Gamma_k[n]\right)^2-1}}\right). \tag{52}$$

Let $R_k'\left(\Gamma_k[n]\right)=0$, i.e.,

$$\left(1+\Gamma_k[n]\right)\sqrt{\left(1+\Gamma_k[n]\right)^2-1} = \frac{Q^{-1}\left(\epsilon\right)}{\sqrt{l_k}}. \tag{53}$$

The corresponding solution is given by

$$\Gamma_{k,0}[n] = \sqrt{\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{\left(Q^{-1}\left(\epsilon\right)\right)^2}{l_k}}} - 1. \tag{54}$$

Note that $\left(1+\Gamma_k[n]\right)\sqrt{\left(1+\Gamma_k[n]\right)^2-1}$ is monotonically increasing with respect to $\Gamma_k[n]$. Therefore, $R_k\left(\Gamma_k[n]\right)$ is monotonically decreasing over the interval $\left(0,\Gamma_{k,0}[n]\right)$, and monotonically increasing over $\left(\Gamma_{k,0}[n],+\infty\right)$. Moreover, since $R_k\left(0\right)=0$, it follows that $R_k\left(\Gamma_{k,0}[n]\right)<0$. Hence, $R_k\left(\Gamma_k[n]\right)$ is a monotonically increasing function of $\Gamma_k[n]$ within the feasible domain. According to [46], the analytical solution to $R_k\left(\Gamma_k[n]\right)=R_k^{\text{th}}$ is $\Gamma_k[n]=2^{R_k^{\text{th}}+\frac{\kappa^*}{2}}-1$, where

$$\kappa^* = \mathcal{W}\left(2^{\frac{2Q^{-1}(\epsilon)}{\sqrt{l_k}}}, -2^{\frac{2Q^{-1}(\epsilon)}{\sqrt{l_k}}}; -4\times 2^{-2R_k^{\text{th}}}\frac{\left(Q^{-1}\left(\epsilon\right)\right)^2}{l_k}\right), \tag{55}$$

with $\mathcal{W}\left(^{\omega_1,\omega_2};\mu\right)$ defined as

$$\mathcal{W}\left(^{\omega_1,\omega_2};\mu\right) = \omega_1 - \sum_{i=1}^{+\infty}\frac{1}{i\times i!} \times \left(\frac{2^{-\omega_1}\mu i}{\omega_2-\omega_1}\right)\mathcal{B}_{i-1}\left(\frac{-2}{i\left(\omega_2-\omega_1\right)}\right), \tag{56}$$

and

$$\mathcal{B}_i\left(z\right) = \sum_{j=0}^{i}\frac{(i+j)!}{j!\,(i-j)!}\left(\frac{z}{2}\right)^j. \tag{57}$$

Thus, (18) can be equivalently expressed in terms of the SNR as

$$P_k\left[n\right] = \Pr\left(\Gamma_k\left[n\right] < 2^{R_k^{\text{th}}+\frac{\kappa^*}{2}}-1\right). \tag{58}$$

For convenience, let $\Gamma_k^{\text{th}} = 2^{R_k^{\text{th}}+\frac{\kappa^*}{2}}-1$ represent the corresponding minimum SNR. Then, (58) can be rewritten as

$$P_k\left[n\right] = \Pr\left(\Gamma_k\left[n\right] < \Gamma_k^{\text{th}}\right). \tag{59}$$

Substituting (5) into (59) yields the outage probability expression

$$P_k\left[n\right] = \Pr\left(|\,X_{\text{NLoS}}\,|^2 < \frac{d_k^2\left[n\right]\sigma_k^2\Gamma_k^{\text{th}} - \alpha_0 P_{k,\text{LoS}}^2\left[n\right]}{\alpha_0\left(1-P_{k,\text{LoS}}\left[n\right]\right)^2 p_k\left[n\right]}\right). \tag{60}$$

Considering $X_{\text{NLoS}}\sim\mathcal{CN}\left(0,1\right)$, which implies $|\,X_{\text{NLoS}}\,|^2\sim$ Exp (1), the real-time outage probability is calculated as

$$P_k\left[n\right] = 1 - \exp\left(\frac{d_k^2\left[n\right]\sigma_k^2\Gamma_k^{\text{th}} - \alpha_0 P_{k,\text{LoS}}^2\left[n\right]}{\alpha_0\left(1-P_{k,\text{LoS}}\left[n\right]\right)^2 p_k\left[n\right]}\right), \tag{61}$$

which completes the proof.

REFERENCES

[1] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives," *IEEE Internet Things J.*, vol. 3, pp. 899–922, Dec. 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:17791330

[2] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-based IoT platform: A crowd surveillance use case," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 128–134, Feb. 2017.

[3] W. Yuan, Y. Cui, J. Wang, F. Liu, G. Sun, T. Xiang, J. Xu, S. Jin, D. Niyato, S. Coleri *et al.*, "From ground to sky: Architectures, applications, and challenges shaping low-altitude wireless networks," *arXiv preprint arXiv:2506.12308*, 2025.

[4] C. Zhao, J. Wang, R. Zhang, D. Niyato, G. Sun, H. Du, D. I. Kim, and A. Jamalipour, "Generative AI-enabled wireless communications for robust low-altitude economy networking," *arXiv preprint arXiv:2502.18118*, 2025.

[5] J. Wu, W. Yuan, and L. Bai, "On the interplay between sensing and communications for UAV trajectory design," *IEEE Internet Things J.*, vol. 10, no. 23, pp. 20 383–20 395, Dec. 2023.

[6] J. Wu, W. Yuan, and L. Hanzo, "When UAVs meet ISAC: Real-time trajectory design for secure communications," *IEEE Trans. Veh. Technol.*, vol. 72, no. 12, pp. 16 766–16 771, Dec. 2023.

[7] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient internet of things communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, Nov. 2017.

[8] Y. Cai, Z. Wei, R. Li, D. W. K. Ng, and J. Yuan, "Joint trajectory and resource allocation design for energy-efficient secure UAV communication systems," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4536–4553, Jul. 2020.

[9] P. K. Sharma and D. I. Kim, "Random 3D mobile UAV networks: Mobility modeling and coverage probability," *IEEE Trans. Wireless Commun.*, vol. 18, no. 5, pp. 2527–2538, May 2019.

[10] I. Vlachos, R. M. Pascazzi, M. Ntotis, K. Spanaki, S. Despoudi, and P. Repoussis, "Smart and flexible manufacturing systems using autonomous guided vehicles and the internet of things," *Int. J. Prod. Res.*, vol. 62, no. 15, pp. 5574–5595, Oct. 2022.

[11] C.-L. Hwang, C.-C. Yang, and J. Y. Hung, "Path tracking of an autonomous ground vehicle with different payloads by hierarchical improved fuzzy dynamic sliding-mode control," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 899–914, Apr. 2018.

[12] V. K, M. Ambalal Sheta, and V. Gumtapure, "A comparative study of stanley, LQR and MPC controllers for path tracking application (ADAS/AD)," in *2019 IEEE International Conference on Intelligent Systems and Green Technology (ICISGT)*, 2019, pp. 67–674.

[13] J. E. Normey-Rico, I. Alcalá, J. Gómez-Ortega, and E. F. Camacho, "Mobile robot path tracking using a robust PID controller," *Control Eng. Pract.*, vol. 9, no. 11, pp. 1209–1214, 2001.

[14] G. V. Raffo, G. K. Gomes, J. E. Normey-Rico, C. R. Kelber, and L. B. Becker, "A predictive controller for autonomous vehicle path tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 92–102, 2009.

[15] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 952–964, 2016.

[16] H. Wang, B. Liu, X. Ping, and Q. An, "Path tracking control for autonomous vehicles based on an improved MPC," *IEEE access*, vol. 7, pp. 161 064–161 073, 2019.

[17] A. A. Hussien, M. J. Marie, and K. S. Gaeid, "Effect of fuzzy PID controller on feedback control systems based on wireless sensor network," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 3, p. 2416, Jun. 2020.

[18] H.-C. Yi, C.-J. An, and J.-Y. Choi, "Compensation of time-varying delay in networked control system over Wi-Fi network," *International Journal of Computers Communications & Control*, vol. 12, no. 3, pp. 415–428, 2017.

[19] H. Jin, J. Wu, W. Yuan, F. Liu, and Y. Cui, "Co-design of sensing, communications, and control for low-altitude wireless networks," *IEEE Trans. Mob. Comput.*, pp. 1–13, Jun. 2025.

[20] H. Li and Y. Shi, "Networked min–max model predictive control of constrained nonlinear systems with delays and packet dropouts," *Int. J. Control*, vol. 86, no. 4, pp. 610–624, Nov. 2013.

[21] J. Wu, W. Yuan, Q. Cheng, and H. Jin, "Towards dual-functional LAWN: Control-aware system design for aerodynamics-aided UAV formations," *submitted to IEEE J. Sel. Areas Commun.*, 2025.

[22] V. Kostina and B. Hassibi, "Rate-cost tradeoffs in control," *IEEE Trans. Autom. Control*, vol. 64, no. 11, pp. 4525–4540, Nov. 2019.

[23] C. Lei, W. Feng, J. Wang, S. Jin, and N. Ge, "Control-oriented power allocation for integrated satellite-UAV networks," *IEEE Wireless Commun. Lett.*, vol. 12, no. 5, pp. 883–887, May 2023.

[24] J. Cao, S. Khan, W. Liu, Y. Li, and B. Vucetic, "Remote UGV control via practical wireless channels: A model predictive control approach," *IEEE Trans. Intell. Veh.*, pp. 1–13, Oct. 2024.

[25] B. Chang, L. Zhang, L. Li, G. Zhao, and Z. Chen, "Optimizing resource allocation in urllc for real-time wireless control systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 8916–8927, 2019.

[26] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[27] J. Zhu, H. Jin, Y. He, F. Fang, W. Huang, and Z. Zhang, "Joint optimization of user scheduling, rate allocation, and beamforming for RSMA finite blocklength transmission," *IEEE Internet Things J.*, vol. 11, no. 17, pp. 27 904–27 915, Sep. 2024.

[28] P. Liu, Z. Li, J. Si, N. Al-Dhahir, and Y. Gao, "Joint information-theoretic secrecy and covertness for UAV-assisted wireless transmission with finite blocklength," *IEEE Trans. Veh. Technol.*, vol. 72, no. 8, pp. 10 187–10 199, Aug. 2023.

[29] Y. Wang, X. Zhou, Z. Zhuang, L. Sun, Y. Qian, J. Lu, and F. Shu, "UAV-Enabled secure communication with finite blocklength," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 16 309–16 313, Dec. 2020.

[30] P. Raut, K. Singh, C.-P. Li, M.-S. Alouini, and W.-J. Huang, "Nonlinear EH-based UAV-assisted FD IoT networks: Infinite and finite blocklength analysis," *IEEE Internet Things J.*, vol. 8, no. 24, pp. 17 655–17 668, Dec. 2021.

[31] L. Turco, J. Zhao, Y. Xu, and A. Tsourdos, "A study on co-simulation digital twin with MATLAB and AirSim for future advanced air mobility," in *2024 IEEE Aerospace Conference*, 2024, pp. 1–18.

[32] H. Wang, B. Liu, X. Ping, and Q. An, "Path tracking control for autonomous vehicles based on an improved MPC," *IEEE Access*, vol. 7, pp. 161 064–161 073, Oct. 2019.

[33] C. Zhao, R. Zhang, J. Wang, D. Niyato, G. Sun, H. Du, Z. Li, A. Jamalipour, and D. I. Kim, "Temporal spectrum cartography in low-altitude economy networks: A generative AI framework with multi-agent learning," *arXiv preprint arXiv:2505.15571*, 2025.

[34] M. Kim and J. Lee, "Outage probability of UAV communications in the presence of interference," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[35] S. He, J. Yuan, Z. An, W. Huang, Y. Huang, and Y. Zhang, "Joint user scheduling and beamforming design for multiuser MISO downlink systems," *IEEE Trans. Wireless Commun.*, vol. 22, no. 5, pp. 2975–2988, Oct. 2022.

[36] K. Ishii, "MPC based power allocation for reliable wireless networked control systems," *IEEE Access*, vol. 9, pp. 60 913–60 922, Mar. 2021.

[37] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 1999.

[38] S. Boyd, "Convex optimization," *Cambridge UP*, 2004.

[39] P. H. Calamai and J. J. Moré, "Projected gradient methods for linearly constrained problems," *Math. Program.*, vol. 39, pp. 93–116, Sep. 1987. [Online]. Available: https://doi.org/10.1007/BF02592073

[40] H. Zhao, P. Poupart, and G. J. Gordon, "A unified approach for learning the parameters of sum-product networks," *Advances in neural information processing systems*, vol. 29, 2016.

[41] Y. Huang, W. Mei, J. Xu, L. Qiu, and R. Zhang, "Cognitive UAV communication via joint maneuver and power control," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 7872–7888, Jul. 2019.

[42] M. Grant, S. Boyd, and Y. Y. CVX, "Matlab software for disciplined convex programming, version 1.21," *CVX Research*, 2011.

[43] L. Salaün, M. Coupechoux, and C. S. Chen, "Joint subcarrier and power allocation in NOMA: Optimal and approximate algorithms," *IEEE Trans. Signal Process.*, vol. 68, pp. 2215–2230, Mar. 2020.

[44] X. Jing, F. Liu, C. Masouros, and Y. Zeng, "ISAC from the sky: UAV trajectory design for joint communication and target localization," *IEEE Trans. Wireless Commun.*, vol. 23, no. 10, pp. 12 857–12 872, Oct. 2024.

[45] G. J. Sutton, J. Zeng, R. P. Liu, W. Ni, D. N. Nguyen, B. A. Jayawickrama, X. Huang, M. Abolhasan, Z. Zhang, E. Dutkiewicz *et al.*, "Enabling technologies for ultra-reliable and low latency communications: From phy and mac layer perspectives," *IEEE Commun. Surv. Tutorials*, vol. 21, no. 3, pp. 2488–2524, Feb. 2019.

[46] S. He, Z. An, J. Zhu, J. Zhang, Y. Huang, and Y. Zhang, "Beamforming design for multiuser urllc with finite blocklength transmission," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 8096–8109, Dec. 2021.