SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2026, Vol. 69, Iss. 1, 112305:1–112305:16
https://doi.org/10.1007/s11432-025-4606-2

Robust joint optimization framework for highly reliable low-latency communication services under traffic uncertainties

Yilin REN, Xinchen LYU, Xiaofeng TAO* & Keda CHEN

National Engineering Research Center of Mobile Network Technologies, Beijing University of Posts and Telecommunications, Beijing 100876, China

Received 14 April 2025/Revised 11 July 2025/Accepted 19 September 2025/Published online 4 January 2026

Abstract Highly reliable low-latency communication (HRLLC), a key paradigm in 6G networks, aims to meet stringent requirements of ultra-low latency and extreme reliability, making it ideal for enabling delay-critical services. However, HRLLC faces challenges due to traffic uncertainties—unpredictable fluctuations that degrade quality of service (QoS) in dynamic environments. Existing optimization methods often rely on simplified network assumptions (precise knowledge of traffic arrivals) or suffer from prohibitive computational complexity (e.g., conditional value at risk, CVaR). This paper proposes a novel robust optimization framework that integrates routing, resource provisioning, and admission control for delay-critical services under uncertain network conditions. While leveraging the Bernstein approximation to handle traffic uncertainty, the proposed framework integrates robust routing, resource provisioning, and admission control into a unified architecture for large-scale HRLLC scenarios. This holistic design enables scalable and delay-aware decision-making beyond the scope of conventional Bernstein-based methods. The framework decomposes the robust optimization problem into subproblems, deriving closed-form solutions for resource allocation and employing a quantized dynamic programming algorithm to achieve an $[\mathcal{O}(\epsilon), \mathcal{O}(1/\epsilon^{N+E})]$-trade-off between optimality loss and computational complexity. Comprehensive packet-level simulations validate the effectiveness of our proposed framework, which outperforms existing methods such as OSPF and greedy admission control with respect to reliability, resource utilization, and admission control. By utilizing Bernstein approximation for resource allocation, the framework achieves a 99.99% delay guarantee and improves the success rate of service routing and admission control by up to 33.3%, with a 25.0% enhancement over benchmark approaches in resource utilization, demonstrating the potential of robust optimization to maintain QoS in dynamic, mission-critical applications.

Keywords 6G, HRLLC, delay-critical, routing, resource provisioning, admission control, traffic uncertainties

Citation Ren Y L, Lyu X C, Tao X F, et al. Robust joint optimization framework for highly reliable low-latency communication services under traffic uncertainties. Sci China Inf Sci, 2026, 69(1): 112305, https://doi.org/10.1007/s11432-025-4606-2

1 Introduction

Highly reliable low-latency communication (HRLLC) is a cornerstone of 6G networks, advancing beyond 5G's ultra-reliable low-latency communication (URLLC) to meet stringent latency (sub-1 ms) and reliability (99.9999%) requirements [1–4]. As 6G evolves toward hyperconnected ecosystems, HRLLC enables mission-critical applications such as autonomous driving, industrial automation, and remote surgery, where real-time responsiveness and zero-failure tolerance are non-negotiable [5, 6]. Its ability to adapt to dynamic environments makes it indispensable for future smart cities and digital twins [7].

Resource allocation and routing are pivotal in achieving HRLLC's dual mandates of ultra-low latency and extreme reliability. Efficient resource allocation ensures optimal bandwidth and power distribution, minimizing contention and maximizing throughput [8]. Meanwhile, adaptive routing determines the shortest-latency paths while avoiding congested or degraded links [9]. Together, these mechanisms directly influence end-to-end delay, packet loss, and service availability. However, their effectiveness hinges on addressing network dynamics and uncertainties—a challenge unmet by conventional approaches that assume static or predictable conditions [10].

Existing approaches for delay-critical services—spanning routing protocols [10–14], resource allocation [15–17], and admission control [18–21]—often rely on deterministic network assumptions or oversimplified models, neglecting real-world stochasticity. While robust optimization (RO) has been applied to uncertainty management in other domains [22], its integration with delay-bounded routing remains largely unexplored. Furthermore, traditional

* Corresponding author (email: taoxf@bupt.edu.cn)

methods face prohibitive computational complexity as network scale and service diversity grow, limiting their practicality [23].

This gap underscores the motivation behind our work. HRLLC services—such as autonomous driving, industrial control, and remote surgery—demand strict QoS guarantees even under unpredictable traffic conditions and resource fluctuations. Traditional optimization frameworks often fail to uphold latency and reliability constraints when faced with such real-world uncertainties. To address these challenges, we propose a robust joint optimization framework that integrates routing, admission control, and resource provisioning under traffic uncertainty. By employing Bernstein approximation, we can manage distributional uncertainty in a tractable manner. To further reduce computational complexity, we design a quantized dynamic programming (QDP) algorithm that balances solution optimality with runtime efficiency.

To the best of our knowledge, this paper is the first to integrate routing, resource allocation, and admission control into a unified optimization framework that is resilient to traffic uncertainty and computationally scalable for large-scale HRLLC networks.

The key contributions of this paper are summarized as follows.

• We introduce a holistic framework that jointly optimizes routing, admission control, and resource provisioning under network uncertainty, targeting highly reliable low-latency communication services. While Bernstein approximation is employed to address chance constraints, the core novelty lies in how it is systematically embedded into a scalable decomposition-based framework that enables closed-form resource decisions and discrete path selection under latency and reliability guarantees.

• We propose to decompose the formulated robust joint optimization problem into two dependent subproblems. The resource allocation subproblem admits a closed-form solution, while the remaining admission problem is transformed into an integer program efficiently solvable by the proposed quantized dynamic programming algorithm. We theoretically prove that the QDP algorithm achieves an $[\mathcal{O}(\epsilon), \mathcal{O}(1/\epsilon^{N+E})]$ trade-off between optimality and time complexity.

• We validate the performance of the proposed framework through packet-level simulations. Resource allocation based on Bernstein approximation ensures reliable distribution with a 99.99% delay guarantee. Compared to existing baselines, our algorithm improves the success rate of service routing and admission control by up to 33.3%, and enhances resource utilization by 25.0%.

The remainder of this paper is organized as follows. Section 2 provides a brief overview of the related work. Section 3 presents our system model. Section 4 formulates and reformulates the problem. Section 5 details the proposed robust joint optimization approach. Section 6 presents simulation results, followed by the conclusion in Section 7.

2 Related work

This section reviews optimization approaches for delay-critical services and robust optimization techniques for handling traffic uncertainties. By comparing existing studies, we highlight our study's novelty.

2.1 Optimization for delay-critical services

Significant advancements in network routing and resource allocation have improved delay-sensitive services, particularly through time-sensitive networking (TSN) (e.g., CBS, CQF, TAS) and software-defined traffic engineering, such as dynamic programming, network calculus, and reinforcement learning [16–24].

Routing and admission control have been jointly optimized in various studies. Ramdhani et al. [16] designed an algorithm maximizing network throughput while minimizing latency. Liu et al. [17] proposed a greedy routing algorithm for MCCQF. Maile et al. [21] employed CBS-based routing with network Calculus to ensure deadline guarantees. Zhu et al. [22] and Budhiraja et al. [24] explored SDN-based heuristic and joint optimization strategies for video transmission. Ji et al. [25] introduced a delay-sensitive user association strategy that improves load balancing by periodic reassociation, but the induced randomness in queueing and access delays limits its applicability to latency-bounded traffic. Recent advances in URLLC have also focused on physical-layer optimization in the context of massive MIMO systems [13, 14]. However, these studies do not account for traffic uncertainty, which is critical in practical URLLC scenarios.

Despite these advancements, existing approaches fail to address traffic uncertainties, such as sudden traffic fluctuations and latency jitter, which significantly affect service reliability and resource provisioning. This limitation necessitates robust optimization techniques capable of adapting to dynamic network conditions.

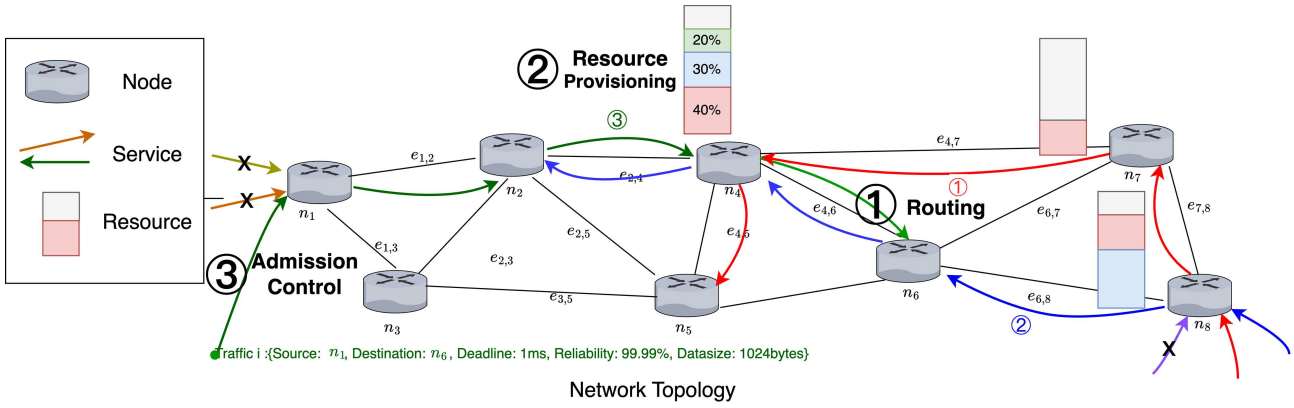

Figure 1 (Color online) The resource provisioning network with routing of providers and K different services.

2.2 Robust optimization

Robust optimization enhances network resilience against uncertainties and has been applied in areas such as computation offloading [26], green communications [27], and fog networks [28]. Despite its effectiveness, CVaR-based formulations often incur high computational complexity, typically requiring Monte Carlo sampling [29] or semi-definite programming [30]. Rao et al. [31] analyzed delay under traffic uncertainty via shortest path routing but relied on precise packet length distributions, which are impractical in real-world systems. Our previous work [15] proposed a Bernstein-based method to efficiently handle traffic uncertainty with limited knowledge, offering scalability in delay-aware service provisioning. However, the joint integration of robust optimization and routing for delay-critical services remains underexplored, motivating this study.

To the best of our knowledge, this paper is the first to consider traffic uncertainties in the routing problem of delay-critical services. Moreover, unlike previous studies, our algorithm achieves joint optimization across admission control, routing decisions, and resource allocation, while addressing the unpredictability of packet sizes. We initially modeled this as a chance-constrained problem and then applied distributed robust optimization with the Bernstein approximation to efficiently find feasible solutions that meet constraints. This method not only improves computational efficiency but also enhances the adaptability of our approach, enabling it to better handle complex requirements under uncertain network conditions.

Remark: While URLLC often requires modeling the physical-layer performance under the finite blocklength (FBL) regime, such as using the normal approximation formula $R^*(n,\epsilon) \approx C - \sqrt{\frac{V}{n}}Q^{-1}(\epsilon)$, our work focuses on network-level orchestration for HRLLC. Therefore, we adopt a capacity-based abstraction, which is appropriate for IP-level packet transmission and allows tractable optimization of routing and provisioning decisions under end-to-end QoS guarantees.

3 System model

Figure 1 illustrates a network with N nodes and E edges, where delay-critical services with uncertain traffic arrivals require end-to-end data delivery within strict deadlines. To accommodate these delay-critical services, the network must preemptively reserve computational resources at nodes and communication resources on edges, ensuring the necessary infrastructure is in place before traffic arrives.

3.1 Service model with traffic uncertainty

Let $\mathbf{K} = \{1, 2, \ldots, K\}$ represent the set of services within the network. Each service k is defined by a tuple $(n^k_{src}, n^k_{dst}, d^k, \tau^k, \epsilon^k, \mathbf{r}^k)$ [15]. Here, n^k_{src} and n^k_{dst} represent the source and destination nodes of service k, respectively. d^k, τ^k, and ϵ^k correspond to the traffic size, service deadline, and reliability requirement, respectively. The resource density for processing per unit traffic of service k is defined as $\mathbf{r}^k = \{r^k_c, r^k_f\}$, where r^k_c and r^k_f refer to the number of CPU processing cycles and the size of transmitted data required to handle the traffic, respectively.

In practice, the system performs traffic prediction in advance of traffic arrival [32], yielding an estimated traffic size $\hat{d}^k$ [33]. Due to traffic uncertainty, however, this estimate may differ from the actual traffic size d^k by an offset

Δd^k, as given by

$$d^k = \hat{d}^k + \Delta d^k. \tag{1}$$

Various techniques, such as long short-term memory (LSTM) models and spatial-temporal cross-domain neural network (STCNet) [34], can be employed to predict $\hat{d}^k$ [15], aiming to minimize Δd^k and enhance prediction accuracy. However, the estimation error Δd^k cannot be entirely eliminated given the nature of AI techniques, and moreover, obtaining the exact distribution of Δd^k, defined as $\mathbb{P}$, is challenging in practice.

Instead of relying on the exact distribution, we can leverage other available statistical properties, such as the first and second-order moments—specifically, the mean value and covariance—of the distribution $\mathbb{P}$ [35]. We define the set of distributions $\mathcal{P}$ as those satisfying the aforementioned range constraints, which can be given by

$$\mathcal{P} = \left\{ \mathbb{P} : \begin{array}{l} \|\mathbb{E}_{\mathbb{P}}(\Delta d)\|_1 \leqslant \mu, \\ \mathbb{E}_{\mathbb{P}}[(\Delta d - \mu)(\Delta d - \mu)^T] \preceq \Sigma \end{array} \right\}. \tag{2}$$

This ambiguity set $\mathcal{P}$ is defined via moment constraints (mean and covariance) without assuming any specific probability distribution, which enhances robustness under traffic uncertainty. Compared to AI-based predictors, our method provides stronger worst-case guarantees under unpredictable variations, where $\mu = [\mu_1, \ldots, \mu_K]$ denotes the upper bound of the mean vector of Δd and $\Sigma = diag(\sigma_1, \ldots, \sigma_K)$ denotes the upper bound of the covariance interval.

3.2 Network topology model and routing decision

The network topology consists of nodes $n \in \mathbf{N}$ and links $e \in \mathbf{E}$ that interconnect these nodes. This topology is encapsulated in an $N \times N$ adjacency matrix $\theta = \{\theta_{u,v} | u, v \in N\}$, where each element $\theta_{u,v} \in \{0, 1\}$ specifies the presence or absence of a direct link between nodes u and v. The routing decision for service k is represented by the vector $\boldsymbol{\varphi}^k = \{\varphi_1^k, \varphi_2^k, \ldots, \varphi_N^k, \varphi_{N+1}^k, \varphi_{N+2}^k, \ldots, \varphi_{N+E}^k\}$, where $\varphi_u^k \in \{0, 1\}$ indicates whether service k traverses node n or link e, $\boldsymbol{\varphi^k} \in \Phi$. The routing decisions are subject to the following constraints [36]:

$$\sum_{\substack{u \in \mathbf{N} \cup \mathbf{E}, \\ u \neq v}} \varphi_u^k \theta_{u,v} \varphi_v^k - \sum_{\substack{v \in \mathbf{N} \cup \mathbf{E}, \\ u \neq v}} \varphi_u^k \theta_{u,v} \varphi_v^k = \begin{cases} 1, & \text{if } u = n_{src}^k, \\ -1, & \text{if } v = n_{dst}^k, \\ 0, & \text{if others.} \end{cases} \tag{3}$$

Eq. (3) is the typical feasibility constraint of flow in graph theory, ensuring the data traverse the network without creating any bottlenecks or disconnections at intermediate nodes. In particular, at the source node n_{src}^k, it mandates that there is exactly one outgoing link, indicating the start of the service's path. Conversely, at the destination node n_{dst}^k, there should be exactly one incoming link, signifying the end of the service's route. For all other nodes within the network, the equation ensures that the number of incoming links equals the number of outgoing links, maintaining the conservation of flow.

Note that physical-layer transmission conditions (e.g., distance, path loss, fading) are not explicitly modeled; instead, their effects are abstracted into the end-to-end latency and reliability constraints discussed in Section 3.3. This abstraction enables our framework to accommodate heterogeneous link qualities in both wired and wireless segments.

3.3 Service admission and resource provisioning model

Let $\mathbf{f}^k = \{f_n^k\}_{n \in \mathcal{N}}$ and $\mathbf{c}^k = \{c_e^k\}_{e \in \mathcal{E}}$ represent the computation and communication resource provisioning decisions for service k, respectively. The binary variable $s^k \in \{0, 1\}$ denotes whether service k is admitted. If $s^k = 1$, service k is admitted into the system; otherwise, it is rejected.

Given a service's traffic size d^k and its routing decision ϕ^k, the total end-to-end delay consists of two components:

$$T^k = T_c^k + T_t^k = \sum_{e \in \mathcal{E}} \frac{r_c^k d^k}{c_e^k} \phi_{N+e}^k + \sum_{n \in \mathcal{N}} \frac{r_f^k d^k}{f_n^k} \phi_n^k,$$

where r_c^k and r_f^k denote the computation and communication resource requirements per unit traffic of service k.

To guarantee that the service meets its deadline τ^k, we enforce the following constraint:

$$s^k T^k \leqslant \tau^k. \tag{4}$$

Under uncertainty in traffic size ($d^k = \hat{d}^k + \Delta d^k$), the above constraint is relaxed to a probabilistic constraint ensuring that the deadline is satisfied with high probability:

$$\mathbb{P}\left\{ s^k d^k \left(\sum_{e\in\mathcal{E}} \frac{r_c^k}{c_e^k}\phi_{N+e}^k + \sum_{n\in\mathcal{N}} \frac{r_f^k}{f_n^k}\phi_n^k \right) \geqslant \tau^k \right\} \leqslant \epsilon^k, \quad \forall k. \tag{5}$$

Eq. (7) enforces the capacity constraints on each communication link and computation node:

$$\sum_{k=1}^{K} s^k \phi_{N+e}^k c_e^k \leqslant C_e, \quad \forall e \in \mathcal{E}, \qquad \sum_{k=1}^{K} s^k \phi_n^k f_n^k \leqslant F_n, \quad \forall n \in \mathcal{N}. \tag{6}$$

These constraints ensure that the total allocated communication and computation resources do not exceed the available capacity on each link C_e or node F_n. Here, ϕ_{N+e}^k (respectively, ϕ_n^k) indicates whether service k uses link e (respectively, node n), and c_e^k (respectively, f_n^k) is the corresponding resource allocated. This abstraction allows the model to accommodate heterogeneous infrastructure while enforcing hard resource limits.

Note that our model abstracts the physical medium and focuses on end-to-end latency and reliability, enabling unified scheduling across both wired and wireless links without relying on specific physical-layer details.

4 Problem formulation and reformulation

This section first presents the problem formulation for jointly optimizing routing, admission, and resource provisioning decisions. For tractability, we reformulate the problem to reduce the search space of the routing decisions.

4.1 Problem formulation

Let p^k denote the revenue generated from successfully admitting flow k. The actual profit for each admitted flow is obtained by subtracting the associated computing and communication costs. The optimization problem can be formulated as follows:

$$\begin{aligned} \mathbf{P}: \max_{f,c,s^k,\boldsymbol{\varphi}^k} & \sum_{k\in K} s^k \left(p^k - \sum_{n\in N} \mathcal{C}_n f_n^k - \sum_{e\in E} \mathcal{C}_e c_e^k \right) \\ \text{s.t.} \quad & (1)\text{–}(3), (6), (7). \end{aligned}$$

4.2 Problem reformulation

Motivated by the fact that many routing decisions are unprofitable or suboptimal, we propose reformulating the routing decision space—originally defined over all possible paths—into a limited set of candidate paths, defined as $\boldsymbol{\varphi}^k$, subject to the following selection conditions.

Condition 1: Profitable selection. For a path P^{k_i} associated with service k, the profit is given by $p^k - \sum_{n^k\in N^{k_i}} \mathcal{C}_n f_n^k - \sum_{e^k\in E^{k_i}} \mathcal{C}_e c_e^k$, where $\mathcal{C}_n$ and $\mathcal{C}_e$ represent the unit costs for computing and communication resources, respectively.

Condition 2: Shortest-path selection. Among the profitable paths with positive profit, we further restrict the selection to the x shortest paths in terms of cost. Let P_x^k denote the x-th shortest path for service k.

Note that Conditions 1 and 2 are not strict system constraints but rather practical simplifications introduced to reduce the routing search space and balance computational complexity with optimality. By adjusting the parameter X, the approximation gap introduced by the candidate path reduction can be effectively controlled, allowing the proposed framework to scale while maintaining near-optimal performance.

According to Conditions 1 and 2, the possible routing paths $\hat{\phi}^k$ can be given by $\hat{\phi}^k = \{P_x^k, \forall x \leqslant X | r(P_x^k) \geqslant 0\}$. Note that Condition 1 only excludes the paths with negative revenues and does not compromise the optimality. Condition 2 prioritizes the paths with large revenues to further reduce the searching space. By meticulously configuring the value of X (the number of paths for selection), we can achieve a balance between optimality loss and computational complexity. Here, $\psi_i^k \in \{0, 1\}$ indicates whether candidate path P_i^k is selected to deliver service k. The problem $\mathbf{P}$ can thus be reformulated as follows:

$$\mathbf{P}': \max_{f,c,s,\boldsymbol{\psi}} \sum_{k\in K} \sum_{i=1}^{X} s^k p^k \psi_i^k, \quad P_i^k \in \hat{\boldsymbol{\varphi}}^k$$

$$\text{s.t. } \mathbf{C1:}\ \mathbb{P}\left\{s^k d^k\left(\sum_{e\in P_i^k}\frac{r_c^k}{c_e^k}+\sum_{n\in P_i^k}\frac{r_f^k}{f_n^k}\right)\leqslant\tau^k\right\}\geqslant 1-\epsilon^k,\ \forall k,i,\ \mathbf{C2:}\ \sum_{k\in K}\sum_{i=1}^{X}s^k\psi_i^k c_e^k\leqslant C_e,\quad \forall e\in E,$$

$$\mathbf{C3:}\ \sum_{k\in K}\sum_{i=1}^{X}s^k\psi_i^k f_n^k\leqslant F_n,\quad \forall n\in N,\ \mathbf{C4:}\ \sum_{k\in K}s^k\leqslant K,$$

$$\mathbf{C5:}\ \sum_{i=1}^{X}\psi_i^k\leqslant 1,\quad \forall k\in K,\ \mathbf{C6:}\ s^k\in\{0,1\},\quad \psi_i^k\in\{0,1\},\quad \forall k\in K,\forall i\in\{1,\ldots,X\}.$$

5 Proposed robust joint optimization for delay-critical services

To efficiently solve the above MINLP problem $\mathbf{P}'$, we adopt a two-stage decomposition strategy. Specifically, we decompose $\mathbf{P}'$ into two subproblems: **P1**, a resource allocation subproblem under fixed routing and admission decisions that can be solved analytically; and **P2**, an integer programming problem that optimizes routing and admission based on the resource cost derived from **P1**. To address **P2** efficiently, we employ a QDP algorithm, which allows a trade-off between computational complexity and solution accuracy by adjusting the quantization interval.

By decoupling $\mathbf{P}'$ into two subproblems, we isolate resource allocation into subproblem **P1**, which can be efficiently approximated using a closed-form solution under fixed admission and routing decisions. The remaining discrete decision problem is formulated as subproblem **P2**, an integer program with a significantly reduced decision space. This decomposition not only reduces computational complexity but also better meets the stringent latency requirements of HRLLC services by avoiding costly dual updates.

Specifically, in problem $\mathbf{P}'$, the routing and admission decisions s^k are represented by binary variables $\boldsymbol{\psi}^k$, while the resource allocations f and c are continuous. This results in a typical MINLP, which is NP-hard in general. To improve tractability, we proceed with the two-stage decomposition described below.

P1: To derive **P1**, we fix the routing and admission variables (s^k,ψ_i^k) in $\mathbf{P}'$, which effectively determines the selected path for each admitted service. This allows us to isolate the continuous resource variables (f_n^k,c_e^k) and transform the original chance constraint **C1**—which enforces delay and reliability guarantees—into a deterministic inequality using the Bernstein approximation. The resulting problem **P1** becomes a convex resource minimization problem under tractable constraints, allowing closed-form solutions while preserving QoS guarantees.

$$\mathbf{P1}:\min_{f,c}\sum_{n_i^k\in N^k}\mathcal{C}_n f_n^k+\sum_{e_i^k\in E^k}\mathcal{C}_e c_e^k\quad\text{s.t. }\mathbf{C1},$$

where the objective is to minimize the allocated resources under the feasibility constraint **C1**. Note that the objective is equivalent to the Lagrangian function by incorporating constraints **C2** and **C3** into the objective, i.e.,

$$\sum_{k\in K}s^k p^k\psi_i^k+\Upsilon_e\left(C_e-\sum_{k=1}^{K}s^k c_e^k\right)+\Psi_n\left(F_n-\sum_{k=1}^{K}s^k f_n^k\right),P_i^k\in\hat{\phi}^k.$$

Notably, given the admission decisions $\mathbf{s}$, the profit term $\sum s_k p_k\psi_i^k$ is fixed, so problem **P1** focuses on minimizing resource allocation under the QoS constraint in **C1**.

P2: **P1** is a basic linear programming problem (except that the probability constraint in **C1**). In the following, we will show that **P1** can be solved in a closed-form manner. By substituting the closed-form solution of **P1** into the original problem $\mathbf{P}'$. Problem $\mathbf{P}'$ is reduced to an integer programming problem **P2** to optimize the admission and routing decisions. Problem **P2** can be given by

$$\mathbf{P2}:\max_{s,\psi}\sum_{k\in K}s^k p^k\psi_i^k,\quad P_i^k\in\hat{\phi}^k\quad\text{s.t. }\mathbf{C2}\text{–}\mathbf{C6}.$$

In the following, we first solve **P1** to find a closed-form solution for resource allocation in Section 5.1, and then solve the integer programming to optimize the routing decisions in Section 5.2.

5.1 Robust resource provisioning

The objective of P1 is to minimize resource consumption while satisfying the QoS requirements under traffic uncertainty. Instead of treating chance constraints in isolation, we embed the Bernstein approximation as part of a

broader optimization framework that decouples resource provisioning from discrete admission and routing decisions. This integration not only enables tractable optimization but also facilitates a closed-form solution that supports real-time decision making in HRLLC scenarios.

Then, the key to solving **P1** is to satisfy the chance constraint **C1** to remove the *Prob* term. To address this challenge, we introduce two auxiliary variables:

$$\sum_{e_i \in E_k} \frac{r_c^k}{c_{e_i}} \leqslant \hat{c}^k, \ \sum_{n_i \in N_k} \frac{r_f^k}{f_{n_i}} \leqslant \hat{f}^k. \tag{7}$$

Using Bernstein's inequality, the transformation of **C1** is as follows [15]:

$$(\hat{c}^k + \hat{f}^k)(\mu^k + \hat{d}^k) + \sqrt{2\ln\frac{1}{\epsilon^k}}\sqrt{{\sigma^k}^2(\hat{c}^k + \hat{f}^k)^2} \leqslant \tau^k, \quad \forall k. \tag{8}$$

By substituting (11) to replace **C1**, problem **P1** can be reformulated as

$$\begin{aligned} \mathbf{P1}': &\min_{f,c} \sum_{n_i^k \in N^k} \mathcal{C}_n f_n^k + \sum_{e_i^k \in E^k} \mathcal{C}_e r_e^k \\ &\text{s.t. } \mathbf{(12)}\text{–}\mathbf{(14)}. \end{aligned}$$

P1′ is a convex problem that satisfies Slater's condition, ensuring strong duality [37]. We can derive a closed-form solution for robust service provisioning decisions by analyzing its Lagrangian dual. The Lagrangian dual of **P1′** is formulated by introducing a vector of Lagrange multipliers for each constraint:

$$\begin{aligned} L(F,c,\Lambda,\Xi,\Phi) = &\sum_{k\in K}\sum_{n_i\in N} \mathcal{C}_n f_{n_i}^k + \sum_{k\in K}\sum_{e_i\in E} \mathcal{C}_e c_{e_i}^k + \sum_{k\in K} \mathbf{\Lambda}^k \left(\left(\hat{c}^k + \hat{f}^k\right)\left(\mu^k + \hat{d}^k + \sigma^k\sqrt{2\ln\frac{1}{\epsilon^k}}\right) - \tau^k \right) \\ &+ \sum_{k\in K} \mathbf{\Xi}^k \left(\sum_{e_i\in E_k} \frac{r_c^k}{c_{e_i}^k} - \hat{c}^k \right) + \sum_{k\in K} \mathbf{\Phi}^k \left(\sum_{n_i\in N_k} \frac{r_f^k}{f_{n_i}^k} - \hat{f}^k \right). \end{aligned}$$

By setting the partial derivatives of (12) with respect to all variables equal to zero, the closed-form solution can be derived, as given by

$$f_n^{k_i} = \left(\sum_{d\in E} r_c^{k_i} \sqrt{\frac{\mathcal{C}_d r_f^{k_i}}{\mathcal{C}_n r_c^{k_i}}} + \sum_{m\in N} r_f^{k_i} \sqrt{\frac{\mathcal{C}_m}{\mathcal{C}_n}} \right) \mathcal{A}, c_e^{k_i} = \left(\sum_{d\in E} r_c^{k_i} \sqrt{\frac{\mathcal{C}_d}{\mathcal{C}_e}} + \sum_{m\in N} r_f^{k_i} \sqrt{\frac{\mathcal{C}_m r_c^{k_i}}{\mathcal{C}_e r_f^{k_i}}} \right) \mathcal{A}, \tag{9}$$

where $\mathcal{A} = \frac{\hat{d}^k + \mu^k + \sigma^k\sqrt{2\ln\frac{1}{\epsilon^k}}}{\tau^k}$ for notation clarity. We define the optimal resource usage according to (11) for each path P_i^k of service k as $\mathbf{f}_{k_i} = [f_1^{k_i}, f_2^{k_i}, \ldots, f_N^{k_i}]^{\mathrm{T}}$ for nodes, and $\mathbf{c}_{k^i} = [c_1^{k_i}, c_2^{k_i}, \ldots, c_E^{k_i}]^{\mathrm{T}}$ for links.

5.2 Quantized DP algorithm and extensions

By substituting the resource allocations $\mathbf{f}^{k_i}$ and $\mathbf{c}^{k_i}$ in Section 5.1 into the original problem $\mathbf{P}'$, problem **P2** becomes an integer programming problem, i.e.,

$$\begin{aligned} \mathbf{P2}': &\max_{s,\psi} \sum_{k\in K} p^k \psi_i^k, \forall k \in \mathbf{K} \\ &\text{s.t. } \mathbf{C2:} \sum_{k=1}^{K} \psi_i^k c_e^{k_i} \leqslant C_e, e \in \mathbf{E}, \ \mathbf{C3:} \sum_{k=1}^{K} \psi_i^k f_n^{k_i} \leqslant F_n, n \in \mathbf{N}, \ \mathbf{C5:} \sum_{i=1}^{\hat{I}_k} \psi_i^k \leqslant 1, \ \mathbf{C7:} \psi_i^k \in \{0,1\}. \end{aligned}$$

Here, s_k is removed from $\mathbf{P}'$, since $s_k = \sum \phi_i^k$. Note that **P2′** is integer programming with large searching spaces of all the possible routing paths. The time-complexity is still non-polynomial. For solving efficiency, we propose to solve the integer programming problem via dynamic programming and meticulously design the quantization interval to achieve a flexible tradeoff between time-complexity and optimality loss.

Quantized DP algorithm. The solution to **P2′** can be efficiently constructed by leveraging dynamic programming (DP) techniques, which build the overall solution from solutions to its sub-problems. Specifically, the solution

for minimizing the resource usage for the first k services depends on whether service k is admitted or not, and which path the admitted service selects, given the solutions to the sub-problems for the first $(k-1)$ services.

Let $R_k(\boldsymbol{\alpha},\boldsymbol{\beta})$ represent the maximum revenue for the sub-problem defined in $\mathbf{P2}'$, where $\boldsymbol{\alpha}=[\alpha_1,\alpha_2,\ldots,\alpha_N]$ denotes the units of resources allocated to each node, and $\boldsymbol{\beta}=[\beta_1,\beta_2,\ldots,\beta_E]$ represents the resources allocated to each link in the system. The Bellman function is expressed as $R_k(\boldsymbol{\alpha},\boldsymbol{\beta})=\max\limits_{\boldsymbol{\psi_k}}\left\{\sum\limits_{k=1}^{k}p^k\psi_i^k \;\middle|\; \sum_{n=1}^{N}\psi_i^k f_n^{k_i}=\alpha_n,\sum_{e=1}^{E}\psi_i^k c_e^{k_i}=\beta_e,\forall n\in N,\forall e\in E\right\}$.

According to the Bellman equation, $R_k(\boldsymbol{\alpha},\boldsymbol{\beta})$ can be computed recursively from the results of the previous sub-problems, $R_{k-1}(\boldsymbol{\alpha},\boldsymbol{\beta})$, as follows:

$$R_k(\boldsymbol{\alpha},\boldsymbol{\beta})=\max\left\{\max_{i\in I_k}\left\{R_{k-1}\{\boldsymbol{\alpha}-\boldsymbol{f}^{k_i},\boldsymbol{\beta}-\boldsymbol{c}^{k_i}\}+p^k\right\},R_{k-1}(\boldsymbol{\alpha},\boldsymbol{\beta})\right\},\tag{10}$$

where $I_k=|P_x^k|$ denotes the number of paths in the path selection of service k.

The solution for $R_k(\boldsymbol{\alpha},\boldsymbol{\beta})$ is selected between the result for the first $(k-1)$ services, $R_{k-1}(\boldsymbol{\alpha},\boldsymbol{\beta})$, and the task offloading to the available maximum-revenue path i for service k, processed as $R_{k-1}(\boldsymbol{\alpha}-\boldsymbol{f}^k,\boldsymbol{\beta}-\boldsymbol{c}^k)$. The Bellman equation leverages the optimal substructure property, reducing time complexity by solving the sub-problem of the smallest possible size [38].

The path selection parameter of service k is expressed by

$$\psi_{k_i}(\boldsymbol{\alpha},\boldsymbol{\beta})=\begin{cases}1, & \text{if } \phi_k(\boldsymbol{\alpha},\boldsymbol{\beta})=\max\limits_{i\in I_k}\left\{\psi^{k-1}\{\boldsymbol{\alpha}-\boldsymbol{f}^{k_i},\boldsymbol{\beta}-\boldsymbol{c}^{k_i}\}+\ell_i^k\right\},\\ 0, & \text{if others.}\end{cases}\tag{11}$$

The admission control parameter s^k is calculated as $s^k=\sum_i\psi^k(\alpha,\beta)$. For the k-th sub-problem in (16), we observe that $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can assume up to $\prod\limits_{k=1}^{k}(I_k+1)$ discrete values. Enumerating all these possible values can be computationally infeasible, especially when $\mathbf{K}$ is large. To address the computational burden of enumerating all possible discrete values, we propose initially relaxing $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to continuous variables, with bounds $(0\leqslant\boldsymbol{\alpha}\leqslant\mathbf{F},0\leqslant\boldsymbol{\beta}\leqslant\mathbf{C})$.

When $\alpha_n\in[0,F_n]$ and $\beta_e\in[0,C_e]$ are large and continuous, enumerating all possible values of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can result in an infinite number of sub-problems and excessive computational complexity. To limit the number of sub-problems and improve the tractability of (15) and (16), we propose further discretizing the resource allocation and constraining the number of sub-problems to be finite. We optimize the quantization interval to balance optimality loss and computational complexity, achieving a trade-off of $[\mathcal{O}(\epsilon),\mathcal{O}(1/\epsilon^{N+E})]$, as discussed in Section 5. The uniform quantizer for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is given by

$$q_{\boldsymbol{\delta}}([\boldsymbol{\alpha},\boldsymbol{\beta}])=\mathbf{M},\ \text{if } \boldsymbol{\delta}\odot\mathbf{M}<[\boldsymbol{\alpha},\boldsymbol{\beta}]\leqslant\boldsymbol{\delta}\odot(\mathbf{M}+\mathbf{1}).\tag{12}$$

Let $\boldsymbol{\delta}=[\delta_{n_1},\delta_{n_2},\ldots,\delta_{n_N},\delta_{e_1},\delta_{e_2},\ldots,\delta_{e_E}]$ represent the quantization interval for each node n and link e. The resource utilization for each path i of service k is also discretized, where $\alpha^{k_i}n=q\delta_n(f^{k_i}n)$ and $\beta^{k_i}e=q\delta_e(c^{k_i}e)$ denote the quantized resources at each node and link, respectively. Similarly, the available resources of nodes and links are discretized as $\alpha_n=q\delta_n(F_n)-\delta_n$ and $\alpha_e=q\delta_e(C_e)-\delta_e$. Consequently, the upper bound of total resource utilization across nodes and links can be computed using the following equation:

$$[\hat{\boldsymbol{\alpha}_n},\hat{\boldsymbol{\beta}_e}]=\max\left\{\left\lceil\frac{\sum_{k=1}^{K}\psi_i^k[\boldsymbol{f}_n^{k_i},\boldsymbol{c}_e^{k_i}]}{\boldsymbol{\delta}}\right\rceil+K,[\boldsymbol{\alpha}_n,\boldsymbol{\beta}_e]\right\},\tag{13}$$

where $\left\lceil\frac{\sum_{k=1}^{K}\psi_i^k[\boldsymbol{f}_n^{k_i},\boldsymbol{c}_e^{k_i}]}{\boldsymbol{\delta}}\right\rceil$ represents the upper bound of the quantized total resource utilization for each node and link when all services $k\in K$ are admitted. As noted in (18), the proposed quantizer may overestimate resource utilization by no more than δ_n for node n and δ_e for link e, i.e., $0\leqslant\delta_n\alpha_n^{k_i}-f_n^{k_i}\leqslant\delta_n$, $0\leqslant\delta_e\beta_e^{k_i}-c_e^{k_i}\leqslant\delta_e$. Given the constraint on the number of services K, the system cannot admit more than K services. Consequently, the quantization error due to the discretization of $\sum_{k=1}^{K}f_n^{k_i}$ cannot exceed $\delta_n^{k_i}K$, while the error for $\sum_{k=1}^{K}c_e^{k_i}$ cannot exceed $\delta_e^{k_i}K$, respectively.

The total number of sub-problems $\phi_k(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is given by $K \prod_{n=1}^{N} \hat{\alpha_n} \prod_{e=1}^{E} \hat{\beta}_e$, determined by the number of services K, nodes N, links E, and the quantized resources $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. After solving all sub-problems, the optimal solution is given by

$$p^{k^*} = \max_{k=0,\ldots,K} \left\{ p \mid \phi_K(\boldsymbol{\alpha}, \boldsymbol{\beta}),\ \boldsymbol{\delta} \odot [\boldsymbol{\alpha}, \boldsymbol{\beta}] \leqslant [\hat{\alpha_n}, \hat{\beta}_e] \right\},$$

where p^* denotes the maximum achievable revenue. Let K^*, α^*, and β^* represent the number of admitted services and the corresponding resource usage.

Asymptotically optimal quantized interval. The choice of the quantized interval affects the optimality of the final results. In the following, we optimize the quantization interval δ to be implemented in the uniform quantizer in (17). We will prove that, by meticulously designing the quantization interval, we can achieve an $[\mathcal{O}(\epsilon), \mathcal{O}(1/\epsilon^{N+E})]$-tradeoff between the optimality loss and time complexity. Here, ϵ is an adjustable co-efficient, and the optimality loss is the gap between the solution achieved by the proposed algorithm and the optimum by enumerating all the possible path selections.

To further distinguish our approach from traditional Bernstein-based optimization, we rigorously analyze the approximation gap introduced by quantization. The following theorem establishes a performance-complexity trade-off that bounds the suboptimality of the proposed framework, thus ensuring its practical applicability in large-scale HRLLC deployments.

Theorem 1. Given any $\epsilon \geqslant 0$, we can set the quantization intervals as

$$\delta_n = \frac{f_{n,\min}^{k_i}}{p_{\max}^{k_i}} \frac{p_f \epsilon}{K}, \quad \delta_e = \frac{c_{e,\min}^{k_i}}{p_{\max}^{k_i}} \frac{p_f \epsilon}{K}.$$

Given the quantization intervals above, the proposed algorithm can achieve an $[\mathcal{O}(\epsilon), \mathcal{O}(1/\epsilon^{N+E})]$-tradeoff between the optimality loss and time complexity, where p_f is the lower bound of the objective and K is the amount of services.

Proof. Please see Appendix A for details.

To efficiently solve the integer programming problem while maintaining scalability, we propose a QDP algorithm that significantly reduces computational complexity and enables a flexible trade-off between optimality and runtime. By tuning the quantization granularity, the algorithm achieves an $[\mathcal{O}(\epsilon), \mathcal{O}(\infty/\epsilon^{\mathcal{N}+\mathcal{E}})]$ balance, ensuring tractable computation with bounded sub-optimality—an essential feature for large-scale HRLLC scenarios. The trade-off is distinct from traditional iterative methods such as Lagrangian or dual decomposition, as our quantized DP approach achieves near-optimality without iterative updates, offering faster convergence for latency-sensitive HRLLC scenarios.

The QDP algorithm can converge with following features.

(1) Monotonic utility improvement: At every iteration t, the accumulated system profit $U(t) = \sum_{k=1}^{K} s^k(t) p^k$ is non-decreasing, i.e., $U(t+1) \geqslant U(t)$, where $s^k(t)$ denotes whether service k is admitted in iteration t, p^k denotes the profit of service k. The monotonicity is due to the Bellman Equation. In particular, in each step of the QDP algorithm, the update rule for $R_k(\alpha, \beta)$ follows the Bellman recursion:

$$R_k(\alpha, \beta) = \max \left\{ R_{k-1}(\alpha, \beta), \max_{i \in \mathcal{I}_k} \left\{ R_{k-1}(\alpha - f_i^k, \beta - c_i^k) + p^k \right\} \right\}.$$

This formulation ensures that at every stage k, the utility value is non-decreasing compared to the previous stage $k-1$.

(2) Finite termination: Due to the finite quantized resource space, the total number of dynamic programming (DP) states is bounded. Thus, the algorithm converges to a local optimum in finite steps. Quantization discretizes (α, β) into intervals of size ϵ. This proves that our optimization-based method has guaranteed convergence with monotonic utility improvement.

6 Simulation results

We conduct experiments on a custom-developed packet-level simulation platform running on a Linux system with an event-driven Python implementation. The network topology is based on the TOTEM dataset [39].

The service arrivals are modeled as a Poisson process [40], which is a classical and analytically tractable model widely used in communication networks to represent random and memory-less arrivals. Packet sizes are assumed to follow a normal distribution, approximating the aggregated behavior of multiplexed flows in large-scale networks,

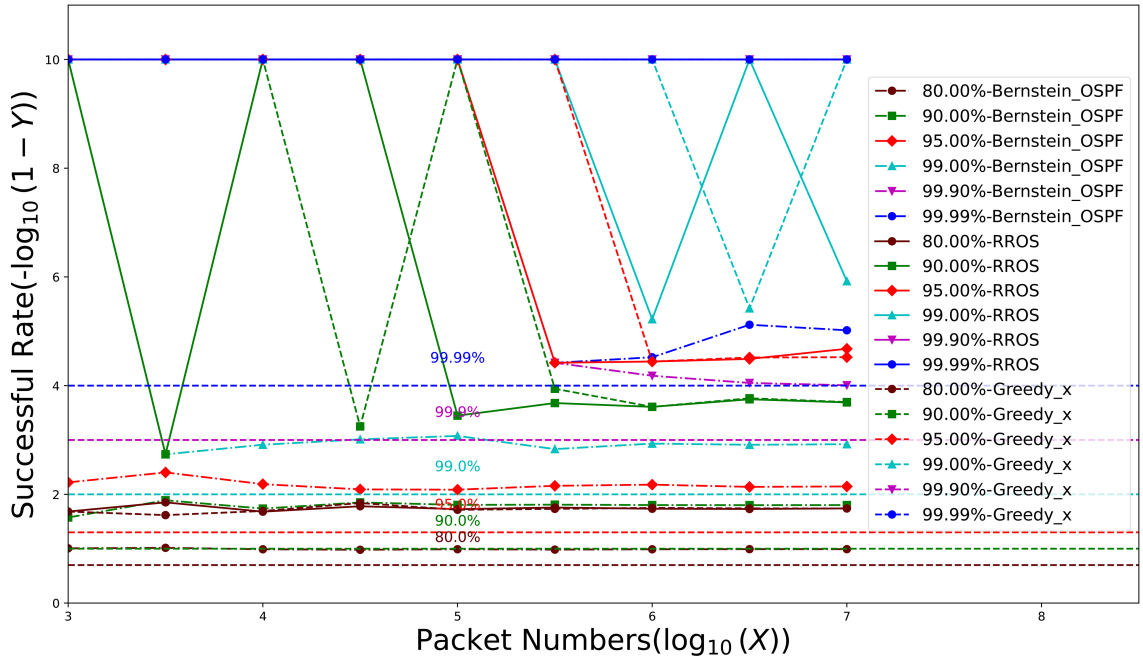

Figure 2 (Color online) The comparison of packet arrival rate as the packet number increases from 10^3 to 10^7.

as validated in prior studies [41, 42]. Origin and target nodes are randomly selected [15]. The traffic consists of six types of flows, each with specific deadlines, reliability demands, and packet size variations [43]. The computational and communication resources per unit data are standardized as $\frac{r_c^k}{r_{c\,ref}^k} = 1$ and $\frac{r_f^k}{r_{f\,ref}^k} = 1$, where $r_{c\,ref}^k$ and $r_{f\,ref}^k$ are reference resource costs. Node and edge usage costs are set to 1 [15].

6.1 Compared algorithms

We compare the proposed RROS scheme with the following baseline algorithms.

- **Bernstein_OSPF** [15]: OSPF-based routing with Bernstein resource provisioning and DP admission control.
- **Greedy_x** [15, 24]: Greedy-based admission control and routing, using the proposed resource allocation method.
- **Greedy_OSPF** [15, 17]: OSPF routing with greedy admission control.
- **Gurobi Solver** [44]: Solves the problem optimally for comparison.

6.2 Service robustness

Figure 2 evaluates six traffic flows with reliability requirements of 80%, 90%, 95%, 99%, 99.9%, and 99.99%. These flows are processed via distributed robust optimization (DRO) using Bernstein approximation for chance constraints. Resources are allocated across multiple paths via DP, Quantized DP, and Quantized Greedy. Admission control and path planning are then performed using DP in our packet-level simulator. The x-axis represents the number of packets (ranging from 10^3 to 10^7), while the y-axis is the negative logarithm (base 10) of the error term ϵ. Higher values indicate greater reliability. Dashed lines denote target reliability requirements. Quantized resource allocation enhances reliability over unquantized methods while meeting reliability constraints. The proposed approach guarantees 99.99% reliability, with Bernstein approximation introducing resource redundancy, enabling 99.9% reliability flows to reach 99.99%. Pre-quantization further boosts reliability.

6.3 Admission control under increasing system capacity

Figures 3 and 4 compare RROS, RROS_OSPF, Greedy_x, and Greedy_OSPF, with $x = 2$. Increasing capacity enables more services to be supported. In Figure 3, when capacity exceeds 2, RROS outperforms RROS_OSPF. This suggests traffic can utilize non-shortest paths, with DP scheduling more services than the greedy approach. At capacity 4, RROS achieves full scheduling, whereas RROS_OSPF, Greedy_x, and Greedy_OSPF require capacity 5, improving resource utilization by 25%. Figure 4 (with 20 services) highlights greater performance differences.

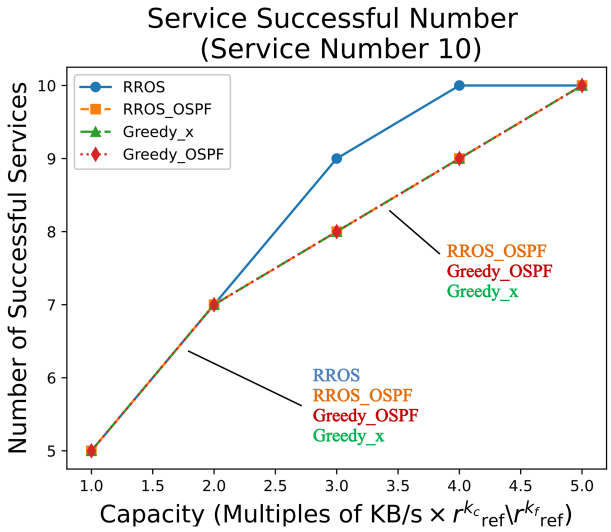

Figure 3 (Color online) Success rate vs. resource capacity ($K = 10, x = 2$).

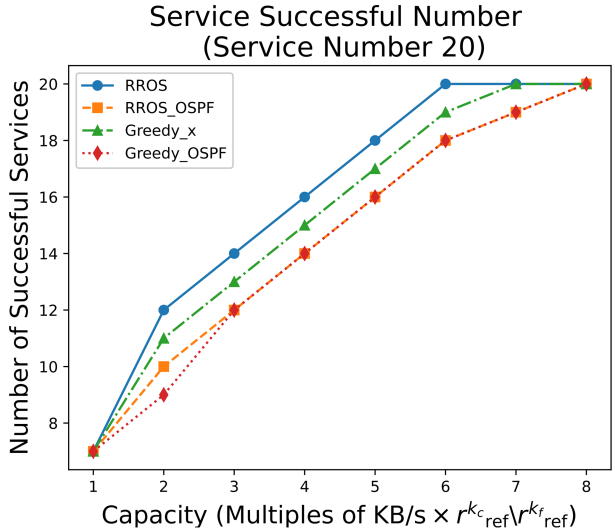

Figure 4 (Color online) Success rate vs. resource capacity ($K = 20, x = 2$).

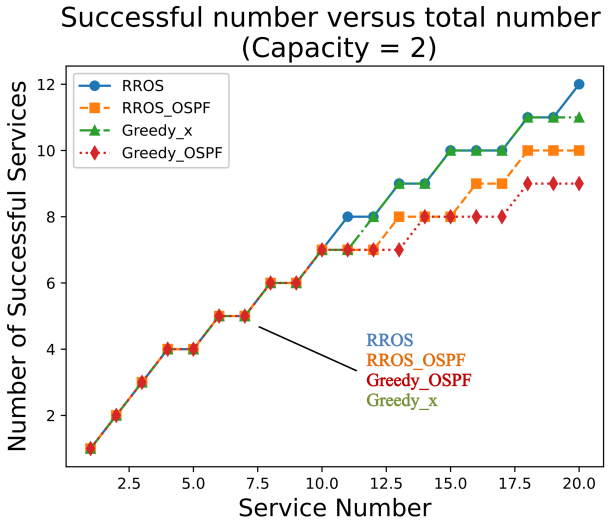

Figure 5 (Color online) Arrival rate vs. service count ($K = 10, x = 2$).

RROS achieves the highest success rate, followed by Greedy_x, RROS_OSPF, and Greedy_OSPF. At capacity 2, RROS improves success rates by 9.1% over Greedy_x, 20% over RROS_OSPF, and 33.3% over Greedy_OSPF. At capacity 6, RROS supports all services, while Greedy_x requires capacity 7, and RROS_OSPF and Greedy_OSPF require capacity 8, yielding 14.3% and 25% higher resource efficiency, respectively. Compared to Figure 3, where 10 services were fully scheduled at capacity 4, only a 0.5× capacity increase enables 2× scheduling capacity, emphasizing improved resource utilization. These results confirm that RROS achieves optimal traffic support while minimizing resource demand.

6.4 Admission control under varying service numbers

Figure 5 evaluates the impact of increasing service numbers on different algorithms, with $x = 2$ and system capacity fixed at 2. As demand grows, all methods schedule more services. When demand exceeds 10, RROS selects non-shortest paths, improving traffic accommodation. At 11 services, RROS schedules 14.3% more traffic than RROS_OSPF, Greedy_x, and Greedy_OSPF. At 15 services, it outperforms both RROS_OSPF and Greedy_OSPF by 25%. When demand reaches 20, RROS schedules 12 flows, compared to 11 (Greedy_x), Bernstein_OSPF, and 9 (Greedy_OSPF). These results highlight RROS's ability to enhance resource utilization by leveraging non-shortest paths and prioritizing overall service count over high-revenue services.

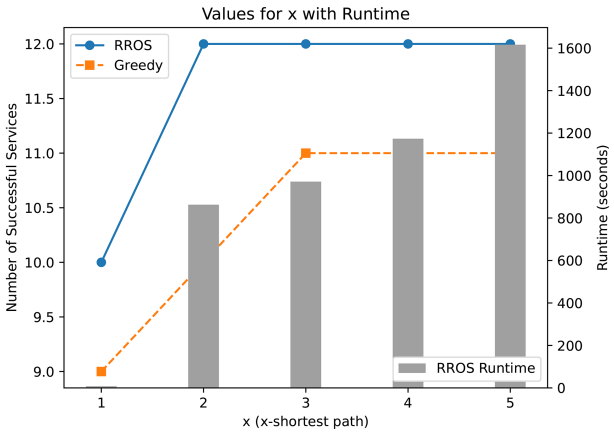

Figure 6 (Color online) Success rate and runtime vs. the x-th shortest path ($K = 20$, capacity=2).

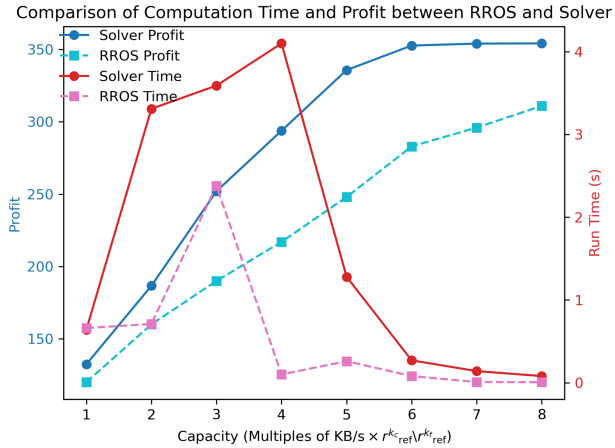

Figure 7 (Color online) Computation time and profit comparison between RROS and Gurobi ($K = 20$, $x = 2$).

6.5 Robustness in delay-critical services

Figure 6 examines the impact of varying x on scheduling efficiency and computation time. With $x = 1$, RROS and RROS_OSPF (likewise Greedy_x and Greedy_OSPF) perform identically, each scheduling 10 flows. As x increases, both RROS and Greedy_x schedule more flows. At $x = 2$, selecting the second-shortest path increases the maximum scheduled services to 12, while $x = 3$ results in 11 scheduled services using a greedy revenue-based approach. While larger x improves scheduling, it also increases computation time. Selecting an optimal x balances scheduling efficiency and runtime.

6.6 Runtime comparison

Figure 7 compares the runtime of RROS and Gurobi as node capacity varies from 1 to 8. At capacity 1, RROS incurs slightly higher latency due to the comparable complexity of solving P2 via QDP, combined with the additional overhead from P1's closed-form resource allocation and the quantization process. In contrast, for larger capacities, RROS consistently outperforms Gurobi in terms of runtime. Specifically, RROS reaches its peak runtime of 2.38 s at capacity 3 and decreases thereafter, while Gurobi peaks at 4.11 s at capacity 4. At capacity 4, RROS achieves a 97.4% reduction in runtime with a 26.1% profit loss; at capacity 1, the runtime reduction is 33.8% with a 14.2% profit loss. These results validate that RROS offers substantial computational savings while maintaining competitive profit, particularly in latency-sensitive scenarios.

7 Conclusion

In this paper, we propose an algorithm for robust optimization of routing, resource allocation, and admission control for delay-sensitive services under traffic uncertainty. Packet-level experiments were conducted to validate the reliability of the algorithm, and the results demonstrate its robustness in meeting delay requirements. Additionally, comparative simulations between the proposed algorithm, OSPF, and greedy scheme show that, under the same resource conditions, the proposed algorithm is able to schedule a greater amount of traffic than the existing method. Furthermore, given a specified traffic volume to be scheduled, the proposed algorithm achieves complete scheduling with lower resource consumption compared to the existing method.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (Grant No. 62327801) and Fundamental Research Funds for the Central Universities (Grant No. 2242022k60006).

References

1 You X H, Wang C X, Huang J, et al. Towards 6G wireless communication networks: vision, enabling technologies, and new paradigm shifts. Sci China Inf Sci, 2021, 64: 110301

2 Cui Q M, You X H, Wei N, et al. Overview of AI and communication for 6G network: fundamentals, challenges, and future research opportunities. Sci China Inf Sci, 2025, 68: 171301

3 Pradhan A, Das S, Piran M J, et al. A survey on physical layer security of ultra/hyper reliable low latency communication in 5G and 6G networks: recent advancements, challenges, and future directions. IEEE Access, 2024, 12: 112320

4 Chang B, Zhang L, Li L, et al. Optimizing resource allocation in URLLC for real-time wireless control systems. IEEE Trans Veh Technol, 2019, 68: 8916–8927

5 Lyu X, Tian H, Ni W, et al. Energy-efficient admission of delay-sensitive tasks for mobile edge computing. IEEE Trans Commun, 2018, 66: 2603–2616

6 Tao T, Wang Y, Li D, et al. 6G hyper reliable and low-latency communication—requirement analysis and proof of concept. In: Proceedings of the IEEE 98th Vehicular Technology Conference (VTC2023-Fall), Hong Kong, 2023. 1–5

7 Haque M E, Tariq F, Khandaker M R A, et al. A survey of scheduling in 5G URLLC and outlook for emerging 6G systems. IEEE Access, 2023, 11: 34372–34396

8 Gharehgoli A, Nouruzi A, Mokari N, et al. AI-based resource allocation in end-to-end network slicing under demand and CSI uncertainties. IEEE Trans Netw Serv Manage, 2023, 20: 3630–3651

9 Yang Y, Wu J, Chen T, et al. Task-oriented 6G native-AI network architecture. IEEE Netw, 2024, 38: 219–227

10 Manias D M, Chouman A, Naoum-Sawaya J, et al. Resilient and robust QoS-preserving post-fault VNF placement. IEEE Netw Lett, 2023, 5: 270–274

11 Lu Z, Shaowei S. Planning method of regional energy syetem with new resources uncertainty. In: Proceedings of the IEEE 14th International Conference on Software Engineering and Service Science (ICSESS), Beijing, 2023. 244–247

12 Shahzad K, Zhou X. Covert wireless communications under quasi-static fading with channel uncertainty. IEEE Trans Inform Forensic Secur, 2021, 16: 1104–1116

13 Peng Q, Ren H, Pan C, et al. Resource allocation for uplink cell-free massive MIMO enabled URLLC in a smart factory. IEEE Trans Commun, 2022, 71: 553–568

14 Peng Q, Ren H, Dong M, et al. Resource allocation for cell-free massive MIMO-aided URLLC systems relying on pilot sharing. IEEE J Sel Areas Commun, 2023, 41: 2193–2207

15 Chen K D, Lyu X C, Ren C S, et al. Robust and reliable resource provisioning for delay-critical services with traffic uncertainty. In: Proceedings of the IEEE Global Communications Conference (GLOBECOM), Kuala Lumpur, 2023. 3475–3480

16 Ramdhani M F, Hertiana S N, Dirgantara B. Multipath routing with load balancing and admission control in software-defined networking (SDN). In: Proceedings of the 4th International Conference on Information and Communication Technology (ICoICT), Malang, 2016. 1–6

17 Liu Y, Zhou D J, Zhan S P, et al. MCCQF: low-latency transmission based on IEEE 802.1 Qch for hierarchical networking. In: Proceedings of the IEEE International Conference on Communications (ICC), Rome, 2023. 6052–6058

18 Li B, Chen L, Yang Z, et al. preDQN-based TAS traffic scheduling in intelligence endogenous networks. IEEE Syst J, 2024, 18: 997–1008

19 Li B, Zhang R, Tian X, et al. Multi-agent and cooperative deep reinforcement learning for scalable network automation in multi-domain SD-EONs. IEEE Trans Netw Serv Manage, 2021, 18: 4801–4813

20 Xu L, Huang Y C, Xue Y, et al. Hierarchical reinforcement learning in multi-domain elastic optical networks to realize joint RMSA. J Lightwave Technol, 2023, 41: 2276–2288

21 Maile L, Hielscher K S J, German R. Delay-guaranteeing admission control for time-sensitive networking using the credit-based shaper. IEEE Open J Commun Soc, 2022, 3: 1834–1852

22 Zhu K J, Ran Y Y, Yang E Z, et al. Joint admission control and routing via neuro-dynamic programming for streaming video over SDN. In: Proceedings of the 13th International Wireless Communications and Mobile Computing Conference (IWCMC), Valencia, 2017. 20–25

23 Chen J, Yang P, Ye Q, et al. Learning-based proactive resource allocation for delay-sensitive packet transmission. IEEE Trans Cogn Commun Netw, 2020, 7: 675–688

24 Budhiraja I, Kumar N, Garg D, et al. Joint traffic admission, resource allocation and mode selection protocol for NOMA-Based D2D users underlaying cellular network for 5G and beyond networks. IEEE Trans Netw Sci Eng, 2024, 11: 4371–4383

25 Ji Q W, Zhu Y X. Delay sensitive user association strategy in massive machine-type communications. Sci China Inf Sci, 2025, 68: 149301

26 Ling Z, Hu F, Zhang Y, et al. Distributionally robust chance-constrained backscatter communication-assisted computation offloading in WBANs. IEEE Trans Commun, 2021, 69: 3395–3408

27 Li L, Shi D, Hou R, et al. Energy-efficient proactive caching for adaptive video streaming via data-driven optimization. IEEE Internet Things J, 2020, 7: 5549–5561

28 Wang J, Liu K, Li B, et al. Delay-sensitive multi-period computation offloading with reliability guarantees in fog networks. IEEE Trans Mobile Comput, 2019, 19: 2062–2075

29 Zhang Y, Li B, Gao F, et al. A robust design for ultra reliable ambient backscatter communication systems. IEEE Internet Things J, 2019, 6: 8989–8999

30 Zymler S, Kuhn D, Rustem B. Distributionally robust joint chance constraints with second-order moment information. Math Program, 2013, 137: 167–198

31 Rao N S V, Batsell S G. On routing algorithms with end-to-end delay guarantees. In: Proceedings of the 7th International Conference on Computer Communications and Networks, Lafayette, 1998. 162–167

32 Luo D H, Yu T, Wu Y W, et al. SPLIT: QoS-aware DNN inference on shared GPU via evenly-sized model splitting. In: Proceedings of the 52nd International Conference on Parallel Processing, Salt Lake City, 2023. 605–614

33 Lyu X, Li Y, He Y, et al. Objective-driven differentiable optimization of traffic prediction and resource allocation for split AI inference edge networks. Trans Mach Learn Comm Netw, 2024, 2: 1178–1192

34 Wang D, Liu Y J, Song B. A credible traffic prediction method based on self-supervised causal discovery. Sci China Inf Sci, 2024, 67: 152303

35 Wang J, Zhang X, Zhang Q, et al. Data-driven spectrum trading with secondary users' differential privacy preservation. IEEE Trans Dependable Secure Comput, 2019, 18: 438–447

36 Gu R T, Qu Y Y, Lian M, et al. Flexible optical network enabled proactive cross-layer restructuring for 5G/B5G backhaul network with machine learning engine. In: Proceedings of the Optical Fiber Communications Conference and Exhibition (OFC), San Diego, 2020. 1–3

37 Nakazato J, Nakamura M, Yu T, et al. Market analysis of MEC-assisted beyond 5G ecosystem. IEEE Access, 2021, 9: 53996–54008

38 Bertsekas D P. Dynamic Programming and Optimal Control. 3rd ed. Belmont: Athena Scientific, 2011

39 Montefiore Institute. TOTEM Dataset. 2024. https://totem.run.montefiore.uliege.be/datatools.html

40 Chen Y, Tang Y, Xu M, et al. Energy-efficient wireless system within average delay and with mixed-erlang-distributed data. IEEE Wireless Commun Lett, 2023, 12: 1239–1243

41 Bertsekas D, Gallager R. Data Networks. Englewood Cliffs: Prentice Hall, 1992

42 Choi B Y, Moon S B, Zhang Z L. Modeling and generating realistic streaming media traffic for network simulation. Comput Netw, 2001, 40: 71–89

43 Xiao Y, Liu J, Wu J, et al. Leveraging deep reinforcement learning for traffic engineering: a survey. IEEE Commun Surv Tutorials, 2021, 23: 2064–2097

44 Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual. 2025. https://www.gurobi.com

Appendix A Proof of Theorem 1

Algorithm 1's energy quantization balances time complexity and optimality loss (Eq. (15)). A smaller quantization interval ϵ reduces loss but increases complexity. This section analyzes this tradeoff by deriving upper and lower bounds for **P2** using its LP relaxation.

$$\begin{aligned}\mathbf{P3}: &\max_{\mathbf{s},\boldsymbol{\omega}} \sum_{k\in K} s_k p_k \psi_i^k, \forall k \in \mathbf{K} \\ &\text{s.t. } \mathbf{C2}, \mathbf{C3}, \mathbf{C4}, \mathbf{C5}, \mathbf{C8}: s_k \in [0,1], \forall k \in \mathbf{K}, \mathbf{C9}: \psi_i^k \in [0,1], \forall k \in \mathbf{K}, \forall i \in \hat{I}_k.\end{aligned}$$

Here, the binary constraint C6 is relaxed to be C8 and C7 is relaxed to C9. Clearly, **P3** gives the upper bound for system revenue.

The Lagrangian problem of **P3** can be written as

$$L(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = \max_{\substack{\mathbf{s}\in C8\\ \boldsymbol{\omega}\in C5,C9}} \sum_{k\in K} s_k p_k \psi_i^k + \gamma\left(K - \sum_{k=1}^{K} s_k\right) + \sum_{e=1}^{E} \lambda_e \left(C_e - \sum_{k=1}^{K} s_k \psi_i^k c_e^{k_i}\right) + \sum_{n=1}^{N} \mu_n \left(F_n - \sum_{k=1}^{K} s_k \psi_i^k f_n^{k_i}\right). \tag{A1}$$

After restructure, the Lagrangian function can be maximized if and only if $L_k(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma)$ is maximized for all $k = 1, 2, \ldots$. The maximization of $L_k(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma)$ can be efficiently solved as

$$s_k^*(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = \begin{cases} 1, & \text{if } \zeta(k,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) > 0, \\ 0, & \text{if } \zeta(k,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) < 0, \end{cases} \tag{A2}$$

where $\zeta(k,i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = \sum_{i=1}^{\hat{I}^k} p_k \omega_{k_i} - \sum_{e=1}^{E} \lambda_e \psi_i^k c_e^{k_i} - \sum_{n=1}^{N} \mu_n \psi_i^k c_e^{k_i} - \gamma$ for notational simplicity. It is also separable, and can be restructured as $\zeta(k,i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = -\gamma + \sum_{i=1}^{\hat{I}_k} \omega_{k_i}(p_k - \sum_{e=1}^{E} \lambda_e c_e^{k_i} - \sum_{n=1}^{N} \mu_n c_e^{k_i})$. In the similar way, the function can be maximized if and only if

$$\psi_i^{k\,*}(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = \begin{cases} 1, & \text{if } \iota(i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) > 0, \\ 0, & \text{if } \iota(i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) < 0, \end{cases} \tag{A3}$$

where $\iota(i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = \ell^{k_i} - \sum_{e=1}^{E} \lambda_e c_e^{k_i} - \sum_{n=1}^{N} \mu_n c_e^{k_i}$ for notation simplicity. Strong duality holds in LP problems[1)]. By substituting (21) and (22) into (20), the dual problem of (20) can be given by

$$\min_{\substack{\boldsymbol{\lambda}>0,\boldsymbol{\mu}>0,\\ \gamma>0,\boldsymbol{\eta}>0}} \sum_{e} \lambda_e C_e + \sum_{n} \mu_n F_n + \gamma K + \sum_{k} \eta_k + \sum_{k\in\mathbf{K}} s_k^* \left(-\gamma + \sum_{i}^{\hat{I}_k} \psi_i^{k\,*} \left(p_k - \sum_{e} \lambda_e c_e^{k_i} - \sum_{n} \mu_n f_n^{k_i} \right) \right), \tag{A4}$$

where the optimal Lagrangian multipliers $\boldsymbol{\lambda}^*$, $\boldsymbol{\mu}^*$ and γ^*, subject to a hyperplane search problem, can be obtained through multidimensional search at a linear complexity of $\mathcal{O}(\prod \hat{I}_k)$.

According to (23) and the optimal Lagrangian multipliers, $\mathbf{K}$ can be divided into three subsets: $\mathbf{K}^+ = \{k \mid \zeta(k,i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) > 0\}$, $\mathbf{K}^0 = \{k \mid \zeta(k,i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = 0\}$ and $\mathbf{K}^- = \{k \mid \zeta(k,i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) < 0\}$. From (36), clearly, $s_k = 1$ for $k \in \mathbf{K}^+$; and $s_k = 0$ for $k \in \mathbf{K}^-$.

*Case*1 : $k \in \mathbf{K}^+$. The function of Lagrangian function for $s_k^*(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma)$ in set $\mathbf{K}^+$ is expressed as follows:

$$\max_{k\in\mathbf{K}^+} \zeta(k,i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma) = -\gamma + \max \sum_{i} \psi_i^k \iota(i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma). \tag{A5}$$

1) Boyd S P. Convex Optimization. Cambridge: Cambridge University Press, 2004.

In this proposition, as $\zeta(k,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma)>0$ and dual variable are positive, and variables ω_{k_i} satisfy C5. The maximization of $L_{k,i}(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma)$ can be efficiently solved as

$$\psi_i^{k*}(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma)=\begin{cases}1, & \text{if } \arg\max_{i\in\hat{I}_k}\iota(i,\boldsymbol{\lambda},\boldsymbol{\mu},\gamma),\\ 0, & \text{if others.}\end{cases} \tag{A6}$$

In this position of $s_k\in\mathbf{K}^+$, if the service is admitted, $s_k=1$, and path i is chosen $\psi_i^k=1$.

*Case*2 : $k\in\mathbf{K}^-$. When $s_k\in\mathbf{K}^-$, if the service is refused, ω_{k_i} satisfy C5. Consequently, $\omega_{k_i}^*=0,\forall i\in\hat{(I_k)}$.

*Case*3 : $k\in\mathbf{K}^0$. When $s_k\in\mathbf{K}^0\neq\emptyset$ and $(\boldsymbol{\lambda},\boldsymbol{\mu},\gamma)=0$, the service k^0 that maximizes revenue, i.e., $k^0=\arg\max_{k^0\in K^0}\left(s_{k^0}\sum_{i\in\hat{I}^k}\psi_i^k p_k\right)$, is partially admitted based on the remaining resources $\left[\mathbf{C}-\sum_{k\in\mathbf{K}^+}\mathbf{c}^{k_i},\mathbf{F}-\sum_{k\in\mathbf{K}^+}\mathbf{f}^{k_i}\right]$. Since the remaining resources cannot fully accommodate any unsatisfied request in $\mathbf{K}^0$, the fraction $s_{k^0}\in(0,1)$ is optimized to maximize total revenue, providing an LP upper bound: $p^{LP}=\sum_{k\in\mathbf{K}^+,i\in\hat{I}^k}p_k+s_{k^0}\psi_0^k p^{k_0}$.

For the lower bound, setting $s_k=0$ for all $k\in\mathbf{K}^+$ results in unused remaining resources, yielding $p_f=\sum_{k\in\mathbf{K}^+}p^k$. Since this integer solution is not optimized, it provides a lower bound for **P1**. The relationship among the lower bound p_f, the optimal solution p^{opt}, and the LP upper bound p^{LP} is established in the following Lemma.

Lemma A1. $p_f\leqslant p^{opt}\leqslant p^{LP}\leqslant 2p_f$.

Proof. Note that p_f is a lower bound for **P1**, while the LP relaxation provides the upper bound ℓ^{LP}. We can obtain that $p_f\leqslant p^{opt}\leqslant p^{LP}$. Besides, $p^{LP}=\sum_{k\in\mathbf{K}^+,i\in\hat{I}^k}p^{k_i}+s_{k^0}\psi_0^k p^{k_0}\leqslant 2\max\sum_{k\in\mathbf{K}^+,i\in\hat{I}^k}p^{k_i}s_{k^0}\psi_0^k p_{k^0}^{k_i}=2p_f$.

Let s_k^{opt} and s_k^* denote the optimal admission decision for **P1** and the decision obtained by the proposed quantized DP algorithm, respectively. Define the original and quantized revenue as $p(s)=\sum_k s_k p^{k_i}(\mathbf{f},\mathbf{c})$ and $x(s)=\sum_k s_k p^{k_i}(\boldsymbol{\alpha},\boldsymbol{\beta})$. The original resource utilization is given by $\rho_n(s)=\sum_k s_k\psi_i^k f_n^{k_i}$ and $\rho_e(s)=\sum_k s_k\psi_i^k c_e^{k_i}$, while the quantized resource utilization is $\chi_n(s)=\sum_k s_k\psi_i^k\alpha_n^{k_i}$ and $\chi_e(s)=\sum_k s_k\psi_i^k\beta_e^{k_i}$. Thus, the optimal revenue is $p^{opt}=p(\mathbf{s}^{opt})$, and the RROS solution is $\overline{p^*}=p(\mathbf{s}^*)$. The resource utilization for the original and quantized problems is $f_n^{opt}=\rho_n(s^{opt})$, $c_e^{opt}=\rho_e(s^{opt})$, $\overline{\alpha_n^*}=\chi_n(s^*)$, and $\overline{\beta_n^*}=\chi_e(s^*)$.

According to [5], we have

$$\delta_e(\beta_e^{k_i}-1)\leqslant c_e^{k_i}\leqslant\delta_e(\beta_e^{k_i}),\delta_n(\alpha_n^{k_i}-1)\leqslant f_n^{k_i}\leqslant\delta_n(\alpha_n^{k_i}). \tag{A7}$$

Hence, we can obtain that $\rho_e(s^{opt})<\delta_e\chi_e(s^{opt})$, $\rho_n(s^{opt})<\delta_n\chi_n(s^{opt})$ and $\rho_e(s^*)\geqslant\delta_e[\chi_e(s^*)-|s^*|]$, $\rho_n(s^*)\geqslant\delta_n[\chi_n(s^*)-|s^*|]$, and therefore, we have

$$\begin{aligned}p^{opt}-\overline{p^*}&=\sum_k s_k^{opt}\sum_i\psi_0^{k\,opt}p^k-\sum_k s^*\sum_i\psi_i^{k*}p^k\leqslant p_{max}^k\left(\sum_k s^{opt}\sum_i\psi_0^{k\,opt}-\sum_k s^*\sum_i\psi_i^{k*}\right)\\&\leqslant p_{i\,max}^k\left(\sum_i\psi_i^{k\,opt}-\sum_i\psi_i^{k*}\right).\end{aligned} \tag{A8}$$

For resource quantization, we can obtain

$$f_n^{opt}-f_n^*<\delta_n[\chi_n(s^{opt}+|s^*|-\chi_n(s^*))],c_e^{opt}-c_e^*<\delta_e[\chi_e(s^{opt}+|s^*|-\chi_e(s^*))]. \tag{A9}$$

For each node or link, the gap between original and quantized resource utilization satisfies

$$f_{n\,min}^{k_i}\left(\sum_k\sum_i\psi_i^{k\,opt}-\sum_k\sum_i\psi_i^{k*}\right)\leqslant\sum_k\sum_i\psi_i^k s_k^{opt}\psi_i^{k\,opt}f_n^{k_i}-\sum_k\sum_i\psi_i^k s_k^*\psi_i^{k*}f_n^{k_i}=f_n^{opt}-f_n^*, \tag{A10}$$

$$c_{e\,min}^{k_i}\left(\sum_k\sum_i\psi_i^{k\,opt}-\sum_k\sum_i\psi_i^{k*}\right)\leqslant\sum_k\sum_i\psi_i^k s_k^{opt}\psi_i^{k\,opt}c_e^{k_i}-\sum_k\sum_i\psi_i^k s_k^*\psi_i^{k*}c_e^{k_i}=c_e^{opt}-c_e^*. \tag{A11}$$

As a result, $p^{opt}-\overline{p^*}<p_{k\,max}\min(\frac{\delta_n[\chi_n(s^{opt}+|s^*|-\chi_n(s^*))]}{f_{n\,min}^{k_i}},\frac{\delta_e[\chi_e(s^{opt}+|s^*|-\chi_e(s^*))]}{c_{e\,min}^{k_i}})$.

As $\delta_n=\frac{f_{n\,min}^{k_i}}{p_{max}^k}\frac{p_f\epsilon}{K}$, $\delta_e=\frac{c_{e\,min}^{k_i}}{p_{max}^k}\frac{p_f\epsilon}{K}$, then we proof the lemma: $p^{opt}-\overline{p^*}<\min_{n,e}\{(p_f\epsilon/K)|s^*|\}\leqslant p_f\epsilon\leqslant p^{opt}\epsilon$. We have $p^{opt}-\overline{p^*}<p^{opt}\epsilon$. Hence, for any $\epsilon>0$, the proposed quantized DP algorithm can achieve $1-\epsilon$-approximation of the optimum, i.e., $\overline{p^*}>(1-\epsilon)p^{opt}$.

In this section, we analyze the complexity.

The following Lemma exhibits the trade-off between the performance and time-complexity of the proposed RROS.

Lemma A2. RROS is able to achieve $(1-\epsilon)$-approximation of the optimum at the complexity of $\mathcal{O}(\frac{K^{N+E}}{\epsilon^{N+E}}\prod_{k=1}^K\hat{I^k})$.

Proof. Recall that $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\beta}}$ and $\hat{I^k}$ are the quantized resource amount of nodes and links, the available links number of service k. The time-complexity of RROS depends on the number of subproblems to be solved. As mentioned in Section 4.2, the number of subproblems is $\mathcal{O}(\prod_{n=1}^N\hat{\boldsymbol{\alpha}^n}\prod_{e=1}^E\hat{\boldsymbol{\beta}^e})$, where the time-complexity for each subproblem using (17) is $\mathcal{O}(\prod_{k=1}^K\hat{I^k})$. The time-complexity of backward induction is $\mathcal{O}(K)$ [28]. Thus, the overall time-complexity of RROS is $\mathcal{O}(\prod_{n=1}^N\hat{\boldsymbol{\alpha}^n}\prod_{e=1}^E\hat{\boldsymbol{\beta}^e}\prod_{k=1}^K\hat{I^k})$.

We can further tighten the upper bound of quantized energy saving in (20) by replacing $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\beta}}$ with the upper bound given by $\hat{\boldsymbol{\alpha}} = \lceil \frac{F_n}{\delta_n} \rceil - 1$ and $\hat{\boldsymbol{\beta}} = \lceil \frac{C_e}{\delta_e} \rceil - 1$. Therefore, we have $\mathcal{O}((\prod_{n=1}^{N} \hat{\boldsymbol{\alpha}^n} \prod_{e=1}^{E} \hat{\boldsymbol{\beta}^e} \prod_{k=1}^{K} \hat{I^k})) = \mathcal{O}(\prod_{n=1}^{N} (\frac{\mathbf{F}_n}{\delta_n}) \prod_{e=1}^{E} (\frac{\mathbf{C}_e}{\delta_e}) \prod_{k=1}^{K} \hat{I^k})$. From Lemma A1, we show that $(1-\epsilon)$-approximation of the optimum can be achieved by using $\delta_n = \frac{f_{n\ min}^{k_i}}{p_{k\,max}} \frac{p_f \epsilon}{K}$, $\delta_e = \frac{c_{e\ min}^{k_i}}{p_{k\,max}} \frac{p_f \epsilon}{K}$. By substituting this into (51), the time-complexity of RROS is

$$\mathcal{O}\left(\left(\prod_{n=1}^{N} \hat{\boldsymbol{\alpha}^n} \prod_{e=1}^{E} \hat{\boldsymbol{\beta}^e} \prod_{k=1}^{K} \hat{I^k}\right)\right) = \mathcal{O}\left(\prod_{n=1}^{N} \frac{F_n p_k K}{f_{n\ min}^{k_i} p_f \epsilon} \prod_{e=1}^{E} \frac{C_e p_k K}{c_{e\ min}^{k_i} p_f \epsilon} \prod_{k=1}^{K} \hat{I^k}\right) = \mathcal{O}\left(\frac{K^{N+E}}{\epsilon^{N+E}} \prod_{k=1}^{K} \hat{I^k}\right).$$

Additional measures can be taken to further reduce the complexity and overhead.

Lemma A2 dictates an $[O(\epsilon), O(1/\epsilon^{N+E})]$-tradeoff between the optimality loss and time-complexity of the proposed quantized DP algorithm. This gives the system an opportunity to reduce the resource consumption of the network by leveraging its hardware capability. In practice, an MEC server can choose the smallest ϵ value based on its capability, thereby attaining the minimum achievable energy consumption of the system.