

Enabling super-resolution as a service via online learning with stochastic queries in mobile edge computing networks[☆]

Ning Chen^{a,b,*,*}, He Huang^{a,**}, Sheng Zhang^b, Jie Wu^c

^a School of Computer Science and Technology, Soochow University, Soochow, 215006, China

^b State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China

^c Center for Networked Computing, Temple University, Philadelphia, PA, 19122, USA

ARTICLE INFO

Keywords:

Edge super-resolution as a service
Online learning
Randomized rounding
Dynamic regret and violation

ABSTRACT

Convolutional networks have recently demonstrated unprecedented performance in on-demand video super-resolution (VSR). VSR enhances the resolution of low-quality videos to meet specific application demands, making it particularly valuable for mobile users. However, enabling VSR as a service at edge presents significant challenges. These include the quality-resource trade-off in model selection, the need for time-coupled decisions regarding VSR query migration, and an unpredictable number of queries stemming from user mobility. To address these issues, we first formulate the problem as a nonlinear integer program with a long-term objective of minimizing inference quality loss. We then design an online learning algorithm comprising two key components: (1) an online learning mechanism employing a virtual queue to quantify long-term constraint violations and decomposing the problem into per-epoch subproblems; and (2) a randomized rounding procedure transforming fractional solutions into integer decisions. We rigorously prove that our algorithm achieves sub-linear dynamic regret regarding optimality loss and sub-linear dynamic violation concerning long-term constraints, under mild conditions on the temporal variation of problem inputs. Extensive evaluations using real-world data confirm the empirical superiority of our approach over state-of-the-art alternatives, demonstrating reductions of at least 23.7% in inference quality loss, 35.1% in dynamic regret, and 41.6% in constraint violations.

1. Introduction

Popular video platforms like YouTube [1], Netflix [2], and iQIYI [3] depend critically on delivering high-quality video to users anytime, anywhere. While cloud-based neural super-resolution has been widely used to reconstruct high-resolution content, it incurs significant transmission overhead when streaming from remote data centers to mobile users. To address this, recent initiatives advocate shifting the VSR process to the network edge via mobile edge computing (MEC) [4–7] infrastructure. Such edge-based VSR minimizes response time to end users, reduces core network traffic, and overcomes the hardware limitations of mobile devices. Unlike on-device methods, it requires neither advanced processors nor extensive battery power, thus broadening accessibility for diverse users.

It is, however, non-trivial for service providers to provision VSR as a service at edge, as it involves the management of pulling SR

models across networks and dispatching users' SR queries over distributed edge nodes, as illustrated in Fig. 1. In particular, managing edge super-resolution in an online manner faces the following key challenges:

First and foremost, loading VSR models from cloud data centers to distributed edge nodes requires dynamically balancing inference quality against resource consumption within the heterogeneous resource and network capacities of the underlying infrastructure. While models delivering higher inference quality typically have larger sizes and demand greater computation during query execution, determining the optimal number of model instances to deploy per edge node per time slot in an online setting is challenging. This stems from a fundamental dilemma: deploying too many instances consumes excessive resources, while deploying too few risks being unable to load additional instances in subsequent slots due to bandwidth constraints. Consequently, current decisions constrain future options, creating a complex temporal coupling that is inherently difficult to manage in online problems.

[☆] This work was supported in part by NSFC (Grant No. 62502331), NSFC (Grant No. 62332013), Natural Science Foundation of Jiangsu province (Grant No. BK20250783), China Postdoctoral Science Foundation, China (Grant No. 2025M771496), State Key Lab. for Novel Software Technology of Nanjing University (Grant No. KFKT2025B24), Jiangsu Funding Program for Excellent Postdoctoral Talent.

* Corresponding author at: School of Computer Science and Technology, Soochow University, Soochow, 215006, China.

** Corresponding author.

E-mail addresses: ningc@suda.edu.cn (N. Chen), huangh@suda.edu.cn (H. Huang), sheng@nju.edu.cn (S. Zhang), jiewu@temple.edu (J. Wu).

Second, user VSR queries are inherently unpredictable due to users' dynamic departures, arrivals, random mobility, and device usage patterns. Compounding this uncertainty is the requirement to pre-deploy machine learning models at edge locations before inference workloads manifest. This obliviousness to future inputs critically complicates online algorithm design. The challenge is further amplified by the need to manage queueing state transitions: migrating decisions of queries in each queue at each edge node must be made prior to the arrival of new queries entering those queues, while simultaneously guaranteeing all queries are served eventually. Intuitively, an online learning approach could be employed. Such an approach would adapt provisioning and workload distribution decisions based on the cost (or penalty) observed after workload arrival and prior decisions. However, designing an effective online learning algorithm capable of achieving queue stability under these complex conditions remains an open and challenging problem.

Existing research falls short of adequately addressing these challenges. While some studies [8–20] focus on optimizing and executing video super-resolution within individual mobile devices, they largely overlook SR optimization across heterogeneous edge nodes, particularly from a service perspective and in online settings. Conversely, other literature [21–26] addresses online service provisioning but falls short of tackling the specific challenges inherent to SR-oriented inference and integrated online decision-making.

In this paper, we study the problem of minimizing the overall inference quality loss for edge VSR services. This optimization must occur over heterogeneous, resource-constrained distributed edge infrastructure while accommodating unpredictable user VSR queries. Our contributions are as follows:

We formulate the edge VSR service provisioning problem as a time-varying non-linear program subject to long-term constraints. The objective is to minimize the cumulative super-resolution loss across multiple time slots, constrained by queueing state transitions, traffic budgets for model loading and query migration, edge node capacities, and service hosting overhead. A key challenge stems from the uncertainty: user queries at each edge node are only revealed after decisions regarding model deployment and workload allocation are made in each time slot. This inherent uncertainty prevents strict constraint satisfaction within individual time slots. Moreover, the problem is NP-hard. Consequently, we design online algorithms that optimize the objective over time while bounding the cumulative violation of the long-term constraints.

We propose a novel polynomial-time online algorithm comprising two key components. First, an online learning module makes per-slot fractional control decisions without observing the current input, utilizing a virtual queue to track and mitigate long-term constraint violations. This module decomposes the problem into per-epoch sub-problems that depend solely on historical inputs. Second, a randomized rounding component converts these fractional decisions into integers. This rounding operates in paired fractions, allowing them to compensate for each other, and ensures that the integer outputs adhere to the constraints in expectation while preserving the fractional solutions' expected values. Furthermore, we provide rigorous formal analysis for our algorithms. Our theoretical results prove that both the dynamic regret, measuring optimality loss against a sequence of instantaneous optimizers, and the dynamic violation, measuring long-term constraint violations, grow sub-linearly over time for our complete online algorithm.

We note that while the high-level algorithmic template (virtual queue-based online learning and dependent rounding) has precedents in the online optimization literature, our work involves several technically non-trivial, VSR-specific elements: (i) the loss-processing-rate ratio (LPR) objective couples model inference quality with processing capacity, capturing a quality-throughput trade-off specific to SR model selection where higher-quality models typically have lower processing capacity; (ii) the queueing state transition constraint explicitly models

Table 1

Summary of notations used for system model.

Symbol	Description
$\mathcal{N}$	Set of heterogeneous edge nodes
$\mathcal{M}$	Set of super-resolution models in remote cloud
$s_{m,t}$	Inference quality loss of model m at time t
c_m	Computational demand per instance of model m
r_m	Traffic demand in bytes of model m
p_m	Processing capability of model m
C_n	Computational capacity of edge node n
O_n	Hosting expense budget of edge node n
$\mathcal{A}_{n,t}$	Set of arriving SR queries of edge n at time t
$d_{n,k,t}$	Display duration of query k of edge n at time t
$q_{n,t}$	Total duration awaiting processing of edge n at time t
τ	Cost for migrating a single inference query
$b_{n,t}$	Transference budget of edge n at time t
$f_t(\cdot)$	Abstract optimization objective
$\mathbf{g}_t(\cdot)$	Abstract long-term constraint
Decision	Description
$x_{n,m,t}$	Indicator of hosting model m at edge n at time t
$y_{n,m,t}$	# ^a of model m hosted at edge n at time t
$z_{k,n,n',t}$	Indicator of migrating query k from edge n to edge n'

#^a: The number of service instance.

video-duration-based backlog dynamics rather than treating queries as unit-cost tasks; and (iii) the two-step randomized rounding procedure (Algorithm 2) is specifically designed to handle the coupling between deployment intensity and the capacity/hosting constraints, ensuring capacity feasibility while preserving per-component expectations for the regret analysis.

We conducted extensive numerical evaluations using Shanghai's 3233 base stations [27] as the edge infrastructure. The dynamic mobile device count at each base station was mapped to the number of super-resolution (SR) queries originating there. We collected this data over a 15-day period, resulting in 360 one-hour time slots. Furthermore, we obtained twenty representative state-of-the-art video super-resolution models from a GitHub repository [28], including EGVSr [29], STDO [30], VIDIM [31], STAR [32], and StableVSR [33]. Experimental results confirm the superiority of our approach over state-of-the-art alternatives, demonstrating reductions of at least 23.7% in inference quality loss, 35.1% in dynamic regret, and 41.6% in constraint violations. Additionally, our proposed algorithm exhibits sublinear growth in constraint violations (consistent with the $\mathcal{O}(\sqrt{T})$ bound) and bounded per-slot dynamic regret under practical variation conditions, aligning with our theoretical analysis. It also performs robustly across different settings (e.g., varying numbers of queries and edge nodes) when its parameters are appropriately controlled.

2. System model and problem formulation

2.1. System model

We summarize the major notations used in Table 1.

Mobile Edge Computing Infrastructure: We investigate a distributed mobile edge computing infrastructure comprising a set of heterogeneous edge nodes (e.g., cellular base stations equipped with co-located microservers), indexed by the set $\mathcal{N} = \{1, 2, \dots, N\}$. Each node operates an access point to serve end-user devices within its coverage area, with the number of connected devices fluctuating over time due to user mobility. These edge nodes are interconnected via high-speed wireline backhaul networks, forming a cohesive system that bridges connectivity to a centralized cloud infrastructure. The cloud hosts multiple diverse super-resolution models and interfaces with the edge layer through an IP backbone network. To characterize computational capabilities, we define the resource capacity of aggregate CPU cycles of edge node $n \in \mathcal{N}$ as C_n .

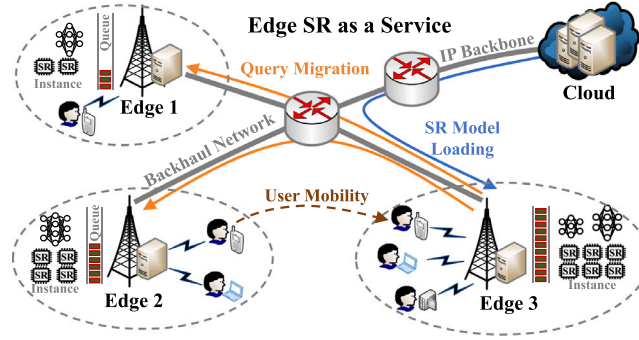


Fig. 1. System architecture of VSR as a service at edge.

Edge Super-Resolution Service: We model the system operation over a discrete-time horizon, represented by consecutive time slots denoted as $\mathcal{T} = \{1, 2, \dots, T\}$. The system comprises a set of super-resolution models, $\mathcal{M} = \{1, 2, \dots, M\}$, hosted in a cloud environment and periodically trained using offline data from edge nodes. For each model $m \in \mathcal{M}$, we define the inference quality loss $s_{m,t}$ as one minus the achieved structural similarity measure (SSIM), representing its deviation from optimal performance. Due to continuous offline training, $s_{m,t}$ varies over time. Each model m has specific resource requirements in computational demand c_m in CPU units per instance and traffic demand for model loading r_m in bytes. We also denote the processing capability p_m of model m as the maximum seconds of video that a single instance can process per time slot.

Processing Pattern of SR Query: For each edge node $n \in \mathcal{N}$ and time slot $t \in \mathcal{T}$, we define $\mathcal{A}_{n,t} = \{1, 2, \dots, A_{n,t}\}$ as the set of arriving user queries, where $A_{n,t}$ denotes the number of queries. Each query $k \in \mathcal{A}_{n,t}$ is associated with a video display duration in seconds, denoted by $d_{n,k,t}$. Each edge node maintains a local SQF (Shortest Query First) queue buffer for incoming inference requests. The queue state $q_{n,t} \in \mathbb{Z}^+$ represents the current backlog at node n in time slot t , measured as the total remaining video duration in seconds awaiting processing. The system supports dynamic query distribution, allowing inference tasks to be processed locally or migrated to neighboring edge nodes based on real-time network conditions and resource availability. Query migration incurs a transmission cost, modeled as τ per second of video, where τ captures the overhead (e.g., in network hops or bandwidth consumption) for both forwarding queries and retrieving results. To manage network traffic, each edge node n adheres to a time-varying migration budget $b_{n,t}$, which constrains the maximum allowable migration traffic while accounting for bandwidth capacity and quality-of-service (QoS) requirements. The budget parameter $b_{n,t}$ serves as a key control variable, balancing computational load distribution against communication overhead across the edge network.

Control Decision: We characterize the edge computing system state through three key decision variables: (1) the *model hosting indicator* $x_{n,m,t} \in \{0, 1\}$, which determines whether SR model m is hosted ($x_{n,m,t} = 1$) or not ($x_{n,m,t} = 0$) at edge node n during time slot t ; (2) the *deployment intensity* $y_{n,m,t} \in \mathbb{N}$, a non-negative integer variable quantifying the number of active instances of model m hosted at edge node n during time t , reflecting the allocated computational capacity; and (3) the *query migration indicator* $z_{k,n,n',t} \in \{0, 1\}$, which captures workload balancing by determining whether query $k \in \mathcal{A}_{n,t}$ is migrated from edge node n to n' ($z_{k,n,n',t} = 1$) or not ($z_{k,n,n',t} = 0$). These variables adhere to domain constraints ensuring physically meaningful solutions within the edge computing framework.

Query Conservation Constraint: In the considered mobile edge computing scenario, each edge node receives SR queries from both local end devices and other edge nodes; therefore, the total arrival queries $A_{n,t}$ of edge node n at time slot t can be calculated as

$$\forall t, n : \sum_{n'} \sum_k z_{k,n,n',t} = A_{n,t}, \quad (c1)$$

this conservation law ensures balanced query distribution.

Query Queue Constraint: As stated before, at the beginning of time slot $t+1$, each edge node n maintains a SQF queue $q_{n,t+1}$, whose length in backlog seconds dynamics reflects: (i) incremental growth through incoming query arrivals $a_{n,t}$ (i.e., $\sum_{n'} \sum_k z_{k,n,n',t} d_{n',k,t}$), and (ii) decremental reduction via processed queries $l_{n,t}$ (i.e., $\sum_m p_m x_{n,m,t} y_{n,m,t}$). In addition, every queue in each node should be finally cleared. Therefore, we get the queue constraint as

$$\begin{aligned} \forall t, n : q_{n,t+1} &= [q_{n,t} + a_{n,t} - l_{n,t}]^+, \\ q_{n,1} &\geq 0, q_{n,T+1} = 0, \end{aligned} \quad (c2)$$

this constraint governs the temporal evolution of queue states, where $[\cdot]^+ \triangleq \max\{\cdot, 0\}$ maintains non-negative queue lengths through a projection operation.

Traffic Budget Constraint: To mitigate potential network congestion, we impose a nodal traffic budget constraint for each edge node n . This restricts (1) cloud-to-edge model transfer costs, captured by the non-negative activation term $[x_{n,m,t} - x_{n,m,t-1}]^+$ for first-time model deployments, and (2) inter-node query migration overhead, quantified by the τ -weighted communication term $b_{n,t}^{\text{mig}}$. The total traffic consumption must satisfy:

$$\forall t, n : \sum_m r_m [x_{n,m,t} - x_{n,m,t-1}]^+ + b_{n,t}^{\text{mig}} \leq b_{n,t}, \quad (c3)$$

where $b_{n,t}^{\text{mig}} = (\tau + \epsilon) \sum_{n' \neq n} \sum_k z_{k,n,n',t} d_{n',k,t}$. Here, τ and ϵ respectively represent the migration cost from node n to n' and the traffic cost for returning processed videos, with ϵ acting as an upscaling factor for the latter. It should be noted that a model only requires a single download from the cloud to an edge node for generating instances, provided the node does not previously host that model.

Capacity Bound Constraint: In contrast to cloud servers with abundant computational resources, edge nodes possess limited capacity that only supports hosting a constrained number of super-resolution (SR) instances. This limitation is formally expressed by the constraint:

$$\forall t, n : \sum_m c_m y_{n,m,t} \leq x_{n,m,t} C_n, \quad (c4)$$

this constraint enforces the resource capacity limit C_n for each node n , and it also ensures that the SR instance can only be deployed on an edge node when the corresponding model is hosted there (i.e., $x_{n,m,t} = 1$).

Service Hosting Constraint: Edge nodes provide super-resolution services to end users, generating hosting overhead; accordingly, we denote the cost of deploying model m per time slot as $o_{m,t}$, which incorporates real-world factors such as computational resource consumption, model complexity, licensing fees, hardware depreciation, and power consumption, enabling unified quantification of heterogeneous resources. Concretely, $o_{m,t}$ aggregates: (a) computational resource consumption (GPU/CPU time proportional to c_m), (b) energy costs (power consumption during model execution), (c) hardware depreciation (amortized GPU/server wear), and (d) software licensing and maintenance overhead. For example, $o_{m,t}$ can be instantiated as $o_{m,t} =$

$\alpha \cdot c_m \cdot p_{\text{GPU}} + \beta \cdot p_{\text{power}} + \gamma \cdot p_{\text{HW}}$, where p_{GPU} , p_{power} , and p_{HW} are unit prices and α, β, γ are weighting coefficients. The budget O_n then reflects the edge operator's financial or contractual limit for node n . Therefore, to effectively manage deployment expenses while adhering to the practical limitations of edge computing, we set the service hosting constraint as

$$\forall t, n : \sum_m o_{m,t} y_{n,m,t} x_{n,m,t} \leq O_n, \quad (\text{c5})$$

where O_n is the hosting expense budget of edge node n .

2.2. Problem formulation

Original Control Problem $\mathcal{P}_1$: Based on the established system model, we formulate the following optimization problem to minimize the cumulative inference quality degradation across all models deployed on edge nodes:

$$\mathcal{P}_1 : \min \sum_t \left\{ \sum_n \sum_m \omega_{m,t} y_{n,m,t} \right\}$$

s.t. Constraint (c1)~(c5)

$$\text{var. } \forall t, n, n', m, k : y_{n,n',t} \in \mathbb{N}, x_{n,m,t}, z_{k,n,n',t} \in \{0, 1\},$$

where $\omega_{m,t}$ is the *loss-processing rate ratio (LPR)* of model m at time slot t , which is defined as model inference loss $s_{m,t}$ dividing model processing ability p_m , i.e., $\omega_{m,t} = s_{m,t} / p_m$.

Control Problem $\hat{\mathcal{P}}_1$ with Observable Inputs: In practical scenarios where ground truth data is unavailable during inference (due to the absence of oracle super-resolution models), we propose an operational version of $\mathcal{P}_1$ that utilizes historical inference outcomes as empirical approximations:

$$\hat{\mathcal{P}}_1 : \min \sum_t \left\{ \sum_n \sum_m \hat{\omega}_{m,t} y_{n,m,t} \right\}$$

s.t. Constraint (c1)~(c5)

$$\text{var. } \forall t, n, n', m, k : y_{n,n',t} \in \mathbb{N}, x_{n,m,t}, z_{k,n,n',t} \in \{0, 1\},$$

where $\hat{s}_{m,t}$ represents the average loss from observable inputs. In practice, since ground truth HR frames are unavailable during real-time inference, $\hat{s}_{m,t}$ is obtained through: (i) an offline calibration phase, where each model m is evaluated on a held-out test set to establish an initial quality estimate $\hat{s}_{m,0}$; (ii) online updates using no-reference quality metrics (e.g., BRISQUE, NIQE) as proxies for SSIM, or pre-computed SSIM values included as metadata when HR originals are available at the server side. The gap between $\hat{s}_{m,t}$ and the true loss $s_{m,t}$ is formally bounded by [Assumption 3](#) (θ_0), ensuring near-optimality of solutions to $\hat{\mathcal{P}}_1$ relative to $\mathcal{P}_1$.

A key challenge in edge-based super-resolution provisioning arises from the inherently stochastic nature of user queries. Without prior knowledge of inference workloads, achieving optimal real-time decision-making under strict per-slot constraints becomes computationally infeasible. This fundamental limitation motivates our approach: instead of enforcing instantaneous constraints, we relax them to long-term average guarantees while designing online algorithms that jointly optimize the objective function and provably bound cumulative constraint violations over time. Therefore, we transform $\mathcal{P}_1$ to the following constrained optimization problem $\mathcal{P}_2$:

$$\mathcal{P}_2 : \min \sum_t \left\{ \sum_n \sum_m \hat{\omega}_{m,t} y_{n,m,t} \right\}$$

$$\text{s.t. } \forall n : \sum_t g_t^{1,n}, \sum_t g_t^{2,n}, \sum_t g_t^{3,n}, \sum_t g_t^{4,n} \leq 0,$$

$$\forall t, n : h_t^{1,n} := \sum_m c_m y_{n,m,t} - x_{n,m,t} C_n \leq 0,$$

$$\forall t, n : h_t^{2,n} := \sum_m o_{m,t} y_{n,m,t} x_{n,m,t} - O_n \leq 0,$$

$$\text{var. } \forall t, n, n', m, k : y_{n,n',t} \in \mathbb{N}, x_{n,m,t}, z_{k,n,n',t} \in \{0, 1\},$$

where we have converted Constraints (c1)~(c3) to their long-term versions correspondingly. This relaxation is necessary because deployment and migration decisions must be made *before* the actual query arrivals $A_{n,t}$ are revealed, making strict per-slot equality enforcement computationally infeasible in the online setting. Physically, per-slot deviations correspond to temporary queueing effects absorbed by the SQF queue buffer at each node (constraint c2): unserved queries accumulate in $q_{n,t+1}$ and are processed in subsequent slots, while the queue clearance condition $q_{n,T+1} = 0$ ensures all queries are ultimately processed. This long-term relaxation approach is well-established in online optimization with stochastic inputs (e.g., Lyapunov optimization, virtual queue methods), which makes it possible to derive provable performance bounds that strict per-slot constraints with unknown future inputs would preclude. For (c1), we have

$$\sum_t g_t^{1,n} := \sum_t \left\{ \sum_{n'} \sum_k z_{k,n',n,t} - A_{n,t} \right\} \leq 0, \quad (1)$$

and similarly we have

$$\sum_t g_t^{2,n} := \sum_t \left\{ A_{n,t} - \sum_{n'} \sum_k z_{k,n',n,t} \right\} \leq 0. \quad (2)$$

For (c2), due to the property of $[\cdot]^+$, we have

$$\forall n : q_{n,T+1} \geq q_{n,T} + a_{n,T} - l_{n,T} \geq q_{n,T-1} + a_{n,T} - l_{n,T} + a_{n,T-1} - l_{n,T-1} \geq \dots \geq q_{n,1} + \sum_t (a_{n,t} - l_{n,t}).$$

Thus, based on $0 \geq q_{n,T+1} - q_{n,1}$, we have

$$\sum_t g_t^{3,n} := \sum_t (a_{n,t} - l_{n,t}) \leq 0. \quad (3)$$

For (c3), we introduce auxiliary binary variables to denote whether model m should be downloaded from the cloud to edge n at time t , i.e., binary variable $x_{n,m,t}^{\text{down}}$, and then have

$$\sum_t g_t^{4,n} := \sum_t \left\{ \sum_m r_m x_{n,m,t}^{\text{down}} + b_{n,t}^{\text{mig}} - b_{n,t} \right\} \leq 0, \quad (4)$$

where $x_{n,m,t}^{\text{down}} := [x_{n,m,t} - x_{n,m,t-1}]^+, \forall n, m, t$.

Remark (Causality and backlog representation). A natural concern is whether replacing the per-slot conservation equality (c1) by the long-term inequalities $g_t^{1,n}$ and $g_t^{2,n}$ permits non-causal or “phantom” query dispatching. It does not, and the physical validity of every per-slot decision is enforced *structurally* rather than only in aggregate. First, the migration variable $z_{k,n,n',t}$ is defined only for queries $k \in \mathcal{A}_{n,t}$ that are physically present at the originating node n during slot t ; a query can be dispatched only from the node and slot at which it has already arrived, so no not-yet-arrived query can ever be assigned, which rules out non-causal assignments by construction. Second, the SQF queue state $q_{n,t}$ in (c2) is the *explicit representation of backlogged* (admitted-but-not-yet-processed) work, measured in residual video-seconds: any admitted duration exceeding the per-slot processing capacity $l_{n,t} = \sum_m p_m x_{n,m,t} y_{n,m,t}$ is carried forward through the recursion $q_{n,t+1} = [q_{n,t} + a_{n,t} - l_{n,t}]^+ \geq 0$, i.e., it remains in the queue rather than being silently discarded or fictitiously served. Hence the long-term relaxation relaxes only the *timing* at which aggregate conservation is met, not whether queries are served. Concretely, the running backlog $q_{n,t}$ is, by construction, the accumulated (non-negative) gap between admitted and processed work, a physically valid quantity; moreover, because of the projection $[\cdot]^+$ the recursion obeys $q_{n,t+1} \geq q_{n,t} + a_{n,t} - l_{n,t}$, so telescoping yields the inequality $\sum_t (a_{n,t} - l_{n,t}) \leq q_{n,T+1} - q_{n,1}$ (the same one-sided bound used above to derive the long-term constraint $g_t^{3,n}$). Combined with the terminal condition $q_{n,T+1} = 0$ (constraint $g_t^{3,n}$) and $q_{n,1} \geq 0$, this gives $\sum_t l_{n,t} \geq \sum_t a_{n,t}$, i.e., the cumulative processed duration is at least the cumulative admitted duration, so every admitted query is eventually served. The two one-sided inequalities then play complementary physical roles: $g_t^{1,n}$ prevents dispatching, in aggregate, more queries than have arrived (no over-service/phantom assignment),

and $g_i^{2,n}$ ensures all arrivals are eventually admitted (no permanent under-service). Thus per-slot physical validity is provided by the causal index set and the backlog queue, and the long-term inequalities only govern the aggregate balance.

Concise Representation: For the ease of the presentation, we simplify the representation of our problem formulation:

$$\begin{aligned} \mathcal{P}_3 : \min \sum_t f_t(\mathbf{v}_t), \quad \text{s.t.} \quad & \sum_t \mathbf{g}_t(\mathbf{v}_t) \leq 0, \mathbf{h}(\mathbf{v}_t) \leq 0, \\ & \forall t : f_t(\mathbf{v}_t) := [\mathbf{0}_{NM \times 1}]^\top, \hat{\omega}_t^\top, [\mathbf{0}_{NM \times K}]^\top \cdot \mathbf{v}_t, \\ & \forall t : \mathbf{g}_t(\mathbf{v}_t) := [g_t^{1,1}, \dots, g_t^{4,1}, \dots, g_t^{1,N}, \dots, g_t^{4,N}]^\top, \\ & \forall t : \mathbf{h}(\mathbf{v}_t) := [h_t^{1,1}, \dots, h_t^{1,N}, h_t^{2,1}, \dots, h_t^{2,N}]^\top, \\ & \forall t : \mathbf{v}_t = [\mathbf{x}_t^{\text{down}}]^\top, \mathbf{y}_t^\top, \mathbf{z}_t^\top \in \mathbb{N}^D, \end{aligned}$$

where the vector $\mathbf{v}_t$ is the aggregation of $\mathbf{x}_t^{\text{down}}$, $\mathbf{y}_t$ and $\mathbf{z}_t$, $\mathbf{x}_t^{\text{down}} \in \{0, 1\}^{NM}$, $\mathbf{z}_t \in \{0, 1\}^{NMK}$, and $D = \dim(\mathbf{v}_t)$ is the dimension of $\mathbf{v}_t$.

Note that problem $\mathcal{P}_3$ is NP-hard even in the offline setting, where all inputs are known a priori and decisions can be made simultaneously for all time slots. This complexity arises from the discrete nature of decision variables and the problem's inherent connection to the minimum knapsack problem. A formal proof is omitted here due to space constraints.

3. Online algorithm design

Our intuition suggests that, in each time slot, constraints should be utilized to their maximum feasible extent without violation. To operationalize this, we design a novel polynomial-time Online Algorithm for Service Hosting (OASH). The algorithm combines two key components: an online learning mechanism that introduces a virtual queue to quantify long-term constraint violations and decomposes the problem into per-epoch subproblems, and a randomized rounding procedure to convert fractional solutions into integer decisions.

3.1. Online learning component

Problem Relaxation: To tackle the NP-hardness of problem $\mathcal{P}_3$, we first relax it to the real domain, i.e.,

$$\tilde{\mathcal{P}}_3 : \min \sum_t f_t(\tilde{\mathbf{v}}_t), \quad \text{s.t.} \quad \sum_t \mathbf{g}_t(\tilde{\mathbf{v}}_t) \leq 0, \mathbf{h}(\tilde{\mathbf{v}}_t) \leq 0,$$

where decision $\tilde{\mathbf{v}}_t$ and its corresponding domain $\tilde{\mathcal{X}}$ are defined in real domain, i.e.,

$$\tilde{\mathcal{X}} = \left\{ \left[\begin{array}{c} (\tilde{\mathbf{x}}_t^{\text{down}})^\top, \tilde{\mathbf{y}}_t^\top, \tilde{\mathbf{z}}_t^\top \end{array} \right]^\top \left| \begin{array}{l} \tilde{\mathbf{x}}_t^{\text{down}} \in \mathbb{N}^{NM}, \\ \tilde{\mathbf{y}}_t \in \mathbb{N}^{NN}, \\ \tilde{\mathbf{z}}_t \in \{0, 1\}^{NMK} \end{array} \right. \right\}. \quad (5)$$

In the edge super-resolution provisioning scenario under consideration, the convex feasible set $\tilde{\mathcal{X}}$ has a bounded radius. Specifically, $\|\mathbf{a} - \mathbf{b}\| \leq R, \forall \mathbf{a}, \mathbf{b} \in \tilde{\mathcal{X}}$, since both the edge resource capacities and the number of queries are finite.

Lemma 1. $\tilde{\mathcal{P}}_3$ is a convex optimization problem

Proof. The proof is presented in [Appendix A](#). $\square$

Queue-Based Methodology: To address this online long-term optimization problem, we propose a queue-based [34] decomposition approach that breaks it into sequential subproblems. This method enables us to iteratively minimize the original objective while maintaining strict adherence to the long-term constraints. Specifically, we introduce a virtual queue vector $\mathbf{Q}_t$ to track constraint violations, where each queue corresponds to a distinct long-term constraint. The aggregated queue vector $\mathbf{Q}_t$ evolves as follows:

$$\forall t : \mathbf{Q}_{t+1} = \max \{ -\mathbf{g}_t(\tilde{\mathbf{v}}_{t+1}), \mathbf{Q}_t + \mathbf{g}_t(\tilde{\mathbf{v}}_{t+1}) \}, \quad (6)$$

Algorithm 1 Online Algorithm for Service Hosting (OASH)

Input: Quality loss function, constraint function sequences, and regular term: $\{f_t, \mathbf{g}_t\}_{t=0}^\infty$ and $\mathbf{h}$;
1: Initialize control decision $\tilde{\mathbf{v}}_0$;
2: Initialize step size $\mu_t = 1/t^2$, $\mathbf{Q}_1 = 0$, $\mathbf{g}_0(\cdot) = 0$;
3: **for** time slot $t = 1$ to T **do**
4: Obtain $\mathbf{v}_t$ by using [Algorithm 2](#) with input $\tilde{\mathbf{v}}_t$;
5: Host edge super-resolution service based on $\mathbf{v}_t$;
6: Observe current estimated loss $\hat{f}_t(\mathbf{v}_t)$ and constraint $\mathbf{g}_t(\mathbf{v}_t)^*$;
7: Obtain $\tilde{\mathbf{v}}_{t+1}$ by solving problem $\tilde{\mathcal{P}}_3^{t+1}$;
8: Update the virtual queues in $\mathbf{Q}_{t+1}$ based on:
9: $\mathbf{Q}_{t+1} = \max \{ -\mathbf{g}_t(\tilde{\mathbf{v}}_{t+1}), \mathbf{Q}_t + \mathbf{g}_t(\tilde{\mathbf{v}}_{t+1}) \}$;
10: **end for**
Output: Control decision sequence: $\{\mathbf{v}_t\}_{t=0}^\infty$.

where $\mathbf{Q}_t = [Q_{1,1,t}, \dots, Q_{4,N,t}] \in \mathbb{R}^{4N}$ represents the virtual queue vector, where the $\max\{\cdot\}$ operator is applied element-wise across vector dimensions. At each time slot, $\mathbf{Q}_t$ is updated according to the constraint violation increments $\mathbf{g}_t(\tilde{\mathbf{v}}_{t+1})$, thereby tracking the cumulative violation of long-term constraints. The $4N$ components of $\mathbf{Q}_t$ specifically encode violations of query queue length constraints, query conservation requirements, and allocated traffic budgets.

Problem Decoupling: To jointly optimize inference quality loss, constraint violations, and the regularization term, we decompose $\tilde{\mathcal{P}}_3$ into multiple tractable subproblems, i.e.,

$$\begin{aligned} \tilde{\mathcal{P}}_3^{t+1} : \min_{\tilde{\mathbf{v}} \in \tilde{\mathcal{X}}} \{ & f_t(\tilde{\mathbf{v}}) + (\mathbf{Q}_t + \mathbf{g}_{t-1}(\tilde{\mathbf{v}}_t))^\top \mathbf{g}_t(\tilde{\mathbf{v}}) + \mu \|\tilde{\mathbf{v}} - \tilde{\mathbf{v}}_t\|_2^2 \} \\ \text{s.t.} \quad & \mathbf{h}(\tilde{\mathbf{v}}) \leq 0. \end{aligned}$$

At the end of slot t , we solve the convex optimization problem $\tilde{\mathcal{P}}_3^{t+1}$ to obtain the fractional decision $\tilde{\mathbf{v}}_{t+1}$, using information from the previous slot. The vector $\mathbf{Q}_t + \mathbf{g}_{t-1}(\tilde{\mathbf{v}}_t)$ quantifies the accumulated violation of the constraint $\mathbf{g}_t$, caused by prior decisions. As this violation grows, the optimization adapts by reducing data transmission or energy consumption in the current slot to compensate. The step size $\mu > 0$ regulates the deviation between consecutive decisions, ensuring stability. Importantly, based on [Lemma 1](#) and the non-negativity condition $\mathbf{Q}_t + \mathbf{g}_{t-1}(\tilde{\mathbf{v}}_t) \geq 0$, $\tilde{\mathcal{P}}_3^{t+1}$ remains a convex problem and can thus be solved efficiently.

*The “observed” loss $\hat{f}_t(\mathbf{v}_t)$ is computed from the estimated quality $\hat{s}_{m,t}$ (obtained via offline calibration and no-reference quality proxies), not the true SSIM-based loss $f_t(\mathbf{v}_t)$. The estimation error is bounded by [Assumption 3](#) (θ_0).

3.2. Randomized rounding component

The two-step design of Step ① ensures that the rounded integer solution satisfies the per-slot capacity constraint $\mathbf{h}(\cdot) \leq 0$ while preserving the expected value of each decision variable. If each fractional $\tilde{y}_{n,m,t}$ were rounded independently, individual expectations would be preserved, but the capacity constraint $\sum_m c_m y_{n,m,t} \leq C_n$ could be violated with high probability, because independent rounding does not control the sum. On the other hand, rounding the vector as a whole would control the sum but fail to preserve per-component expectations, breaking the guarantee $\mathbb{E}[\mathbf{v}_t] = \tilde{\mathbf{v}}_t$ needed for the regret analysis. The core idea of Step #1-1 is to uniformly scale the column vector $\mathbf{F}_t$ (containing the fractional deployment intensities of model m across all edge nodes) by a random factor, chosen so that the scaled column sum becomes an integer ($\lfloor \epsilon \rfloor$ or $\lceil \epsilon \rceil$) while preserving the expected value of each component.

Our randomized rounding algorithm, outlined in [Algorithm 2](#), proceeds in two stages. Step ① rounds $\tilde{\mathbf{y}}_t$ randomly while strictly adhering to $\mathbf{h}(\cdot) \leq 0$. This step iteratively selects and rounds pairs of fractional values simultaneously. Step ② independently rounds $\tilde{\mathbf{x}}_t^{\text{down}}$ and $\tilde{\mathbf{z}}_t$. This

Algorithm 2 Randomized Rounding**Input:** Fractional control decision $\tilde{\mathbf{v}}_t \in \tilde{\mathcal{X}}$.

```

1: % ① Rounds  $\tilde{\mathbf{y}}_t$ 
2: for each model  $m \in \mathcal{M}$  do
3:   Set  $\mathbf{F}_t = [\tilde{y}_{1,m,t}, \dots, \tilde{y}_{N,m,t}]^\top$ ;
4:   % #1-1 Ensures integer sum
5:   Set  $e = \mathbf{1}^\top \mathbf{F}_t, \gamma_1 = \frac{\lfloor e \rfloor}{e}, \gamma_2 = \frac{\lceil e \rceil}{e}$ 
6:    $\mathbf{U}_t^\top = \begin{cases} \text{prob.}(\lfloor e \rfloor - e) : [\gamma_1 F_{1,t}, \dots, \gamma_1 F_{N,t}] \\ \text{prob.}(e - \lfloor e \rfloor) : [\gamma_2 F_{1,t}, \dots, \gamma_2 F_{N,t}] \end{cases}$ 
7:   % #1-2 Ensures integer components
8:   Set  $\mathbf{I}_t = \mathbf{U}_t - \lfloor \mathbf{U}_t \rfloor$ 
9:   while  $I_{i,t} \in (0, 1) \wedge I_{j,t} \in (0, 1)$  do
10:     $\theta_1, \theta_2 = \min\{1 - I_{i,t}, I_{j,t}\}, \min\{I_{i,t}, 1 - I_{j,t}\}$ 
11:     $(I_{i,t}, I_{j,t}) = \begin{cases} \text{prob.}(\frac{\theta_2}{\theta_1 + \theta_2}) : (I_{i,t} + \theta_1, I_{j,t} - \theta_1) \\ \text{prob.}(\frac{\theta_1}{\theta_1 + \theta_2}) : (I_{i,t} - \theta_2, I_{j,t} + \theta_2) \end{cases}$ 
12:   end while
13:    $y_{n,m,t} = I_{n,t}, \forall n \in \mathcal{N}$ 
14: end for
15: % ② Rounds  $\tilde{\mathbf{x}}_t^{\text{down}}$  and  $\tilde{\mathbf{z}}_t$ 
16: Set  $\mathbf{E}_t = [\tilde{\mathbf{x}}_t^{\text{down}}]^\top, \tilde{\mathbf{z}}_t^\top$ ;
17: Set  $\mathbf{E}'_t = \mathbf{E}_t - \lfloor \mathbf{E}_t \rfloor$ ;
18: for each column  $c$  in  $\mathbf{E}'_t$  do
19:    $\mathbf{E}''_{c,t} = 1$  with probability  $\mathbf{E}'_{c,t}$ ; otherwise 0.
20: end for
21:  $[\tilde{\mathbf{x}}_t^{\text{down}}]^\top, \tilde{\mathbf{z}}_t^\top = \lfloor \mathbf{E}_t \rfloor^\top + \mathbf{E}''_{c,t}^\top$ ;

```

Output: Integer decision $\mathbf{v}_t = [(\mathbf{x}_t^{\text{down}})^\top, \mathbf{y}_t^\top, \mathbf{z}_t^\top]^\top$

separation is permissible because the constraint $\mathbf{h}(\tilde{\mathbf{v}}_t)$ in subproblem $\tilde{\mathcal{P}}_3^{t+1}$ pertains only to $\tilde{\mathbf{y}}_t$. We elaborate on these steps below.

Step ① rounds $\tilde{\mathbf{y}}_t$, where Step #1-1 ensures that the sum of $y_{n,m,t}$ over all $n \in \mathcal{N}$ is an integer while maintaining $\mathbb{E}[y_{n,m,t}]$ equal to its pre-rounding value (lines 3–6). This is achieved by either: (a) scaling down each column of $\mathbf{F}_t$ by a factor $\gamma_1 < 1$, yielding a column sum in $\mathbf{U}_t$ of $\lfloor \mathbf{1}^\top \mathbf{F}_t \rfloor$; or (b) scaling up each column by $\gamma_2 > 1$, yielding a sum of $\lceil \mathbf{1}^\top \mathbf{F}_t \rceil$. The respective probabilities for these actions are $\lfloor e \rfloor - e$ and $e - \lfloor e \rfloor$, guaranteeing $\mathbb{E}[\mathbf{U}_t] = \mathbf{F}_t$. Crucially, for any given m with $e \leq \Omega_m$ and $\Omega_m \in \mathbb{Z}$, the sum e increases at most to $\lceil e \rceil$, which satisfies $\lceil e \rceil \leq \Omega_m$, thereby preserving $\mathbf{h}(\cdot) \leq 0$.

Step #1-2 further rounds the remaining fractional values in $\mathbf{U}_t$ to integers (lines 7–14), maintaining the overall sum (achieved in Step #1-1) and ensuring each value's expectation equals its pre-rounding value. First, $\mathbf{U}_t$ is decomposed into its integer and fractional parts. Then, using the fractional parts as probabilities, columns are rounded in pairs such that their fractional components compensate for each other. Since the total column sum is already an integer from Step #1-1, this pairwise rounding guarantees that the resulting $\mathbf{V}_t$ is a binary vector (0/1). The inner while loop has a worst-case complexity of $O(N^2)$ [35].

Step ② processes each fractional value individually, rounding it to an integer such that its expectation equals the original fractional value (lines 15–21). Specifically, each fractional value is decomposed into its integer part $\lfloor \mathbf{E}_t \rfloor$ and fractional part $\mathbf{E}'_t$, where the fractional part dictates the rounding probability. Finally, the control decision vector $\mathbf{v}_t$ is formed by combining $\mathbf{x}_t^{\text{down}}$, $\mathbf{y}_t$, and $\mathbf{z}_t$. We emphasize that obtaining $\mathbb{E}[\mathbf{v}_t] = \tilde{\mathbf{v}}_t$ from our rounding algorithm is essential for the subsequent performance analysis.

4. Performance analysis

4.1. Definition of key performance metrics

Dynamic Regret: Dynamic regret measures the discrepancy between the long-term objective achieved by online decisions $\{\mathbf{v}_t\}$ and

an optimal dynamic benchmark $\{\mathbf{v}_t^*\}$. This benchmark comprises the solutions obtained by optimizing each per-slot problem a priori, given oracle knowledge of the corresponding inputs. We denote the optimal solutions in the integral (discrete) and real (continuous) domains as $\{\hat{\mathbf{v}}_t^*\}$ and $\{\tilde{\mathbf{v}}_t^*\}$, respectively. The dynamic regret for each domain is then defined as follows:

$$\text{Reg}(T) := \mathbb{E} \left[\sum_t f_t(\hat{\mathbf{v}}_t) \right] - \sum_t f_t(\hat{\mathbf{v}}_t^*), \quad (7)$$

$$\hat{\mathbf{v}}_t^* \in \arg \min_{\mathbf{v} \in \mathcal{X}} f_t(\mathbf{v}), \quad \text{s.t. } \mathbf{g}_t(\mathbf{v}) \leq \mathbf{0}, \mathbf{h}(\mathbf{v}) \leq \mathbf{0},$$

$$\widetilde{\text{Reg}}(T) := \sum_t f_t(\tilde{\mathbf{v}}_t) - \sum_t f_t(\tilde{\mathbf{v}}_t^*), \quad (8)$$

$$\tilde{\mathbf{v}}_t^* \in \arg \min_{\mathbf{v} \in \tilde{\mathcal{X}}} f_t(\mathbf{v}), \quad \text{s.t. } \mathbf{g}_t(\mathbf{v}) \leq \mathbf{0}, \mathbf{h}(\mathbf{v}) \leq \mathbf{0}.$$

The expectation $\mathbb{E}[\cdot]$ in Eq. (7) accounts for the randomness introduced by randomized rounding.

Dynamic Constraint Violation: To ensure long-term constraints are approximately satisfied over time, dynamic violation quantifies the cumulative violation incurred by the online decisions. For each constraint i (where $i = 1, \dots, N+1$), the dynamic violation in the integral and real domains is defined respectively as:

$$\text{Viol}_i(T) := \mathbb{E} \left[\sum_t g_{t,i}(\hat{\mathbf{v}}_t) \right], \quad (9)$$

$$\widetilde{\text{Viol}}_i(T) := \sum_t g_{t,i}(\tilde{\mathbf{v}}_t), \quad (10)$$

where $g_{t,i}$ denotes the i th element of the long-term constraint vector $\mathbf{g}_t$. The expectation in $\text{Viol}_i(T)$ in Eq. (9) is employed to mitigate the effects of decision randomness.

4.2. Theoretical analysis

We establish that the dynamic regret and dynamic constraint violation of our algorithm exhibit sublinear growth with respect to the time horizon T . The following standard and technical assumptions, which are readily satisfiable in typical scenarios, facilitate our analysis:

Assumption 1 (Boundedness). The decision domain, objective function, and constraint function are bounded. Specifically, $\exists R, F, G > 0$ such that $\forall t$ and $\forall \mathbf{v} \in \mathcal{X}$:

$$\|\mathbf{v}\|_2 \leq R, |f_t(\mathbf{v})| \leq F, \|\mathbf{g}_t(\mathbf{v})\|_2 \leq G. \quad (11)$$

This ensures transient suboptimal decisions induce bounded cumulative effects on overall performance metrics (e.g., inference quality loss, hosting cost, and migration overhead).

Assumption 2 (Lipschitz Continuity). The constraint function $\mathbf{g}_t(\cdot)$ is Lipschitz continuous with modulus β for all t , i.e.,

$$\forall \mathbf{v}_1, \mathbf{v}_2 \in \mathcal{X} : \|\mathbf{g}_t(\mathbf{v}_1) - \mathbf{g}_t(\mathbf{v}_2)\|_2 \leq \beta \|\mathbf{v}_1 - \mathbf{v}_2\|_2. \quad (12)$$

This is prevalent in primal–dual online algorithm analysis.

Assumption 3 (Prediction Error Bound). The discrepancy between the objective with observable inputs and the ground truth is bounded: $\sum_t [f_t(\tilde{\mathbf{v}}_t^*) - f_t^G(\tilde{\mathbf{v}}_t^*)] \leq \theta_0$. This guarantees near-optimality of solutions to $\tilde{\mathcal{P}}_1$ relative to $\mathcal{P}_1$. We emphasize that Assumption 3 bounds a cumulative (time-summed) discrepancy, not a per-slot one, and it suffices that this cumulative bound is sublinear, $\theta_0 = \theta_0(T) = o(T)$: in that case the estimation error contributes only a sublinear additive term to the regret (it is absorbed into the constant C_1 when $\theta_0 = \mathcal{O}(1)$, or into a sublinear term otherwise) and does not affect the asymptotic rates of Corollary 1.

Remark (Order preservation of model selection). The connection between the proxy-estimated loss and the true SSIM-based objective is weaker than uniform numerical agreement, because the online decision depends on $\hat{s}_{m,t}$ only through the LPR coefficients $\hat{\omega}_{m,t} = \hat{s}_{m,t}/p_m$ and, for the model-selection component, only through their relative ordering: for given dual prices and capacity/hosting costs, a node prefers to deploy the model with the smaller LPR $\hat{\omega}_{m,t}$ (best quality per unit processing rate). Hence absolute accuracy of $\hat{s}_{m,t}$ is not required; it suffices that the proxy preserves the LPR ranking. Let $\xi_t := \max_m |\hat{\omega}_{m,t} - \omega_{m,t}|$ be the per-slot proxy error and let $\Delta_{n,t}$ be the LPR gap between the best and second-best model at node n in slot t . If $\Delta_{n,t} > 2\xi_t$ (a margin condition), then $\arg \min_m \hat{\omega}_{m,t} = \arg \min_m \omega_{m,t}$, so the estimated-loss decision coincides with the true-loss decision and incurs no extra regret in that slot. Summing only the slots where the margin fails, the additional regret due to estimation is at most $\sum_t 2\xi_t \mathbf{1}\{\Delta_{n,t} \leq 2\xi_t\}$, which is exactly the cumulative quantity bounded by $\theta_0(T)$ in [Assumption 3](#). This gives the assumption a concrete, content-dependent interpretation: $\theta_0(T)$ is small (indeed sublinear) whenever the proxy is order-accurate in all but a sublinear number of slots, a far weaker requirement than $\hat{s}_{m,t} \approx s_{m,t}$ holding uniformly across all content. Robustness to changing video content therefore reduces to the proxy preserving model rankings rather than tracking absolute SSIM, which is realistic since no-reference proxies (BRISQUE, NIQE) and offline-calibrated estimates tend to be consistent in ordering even when their absolute calibration drifts.

Assumption 4 (Action Evolution). The algorithm step size μ satisfies $\mu \geq \beta^2$, where β is the Lipschitz modulus. This ensures μ is sufficiently large, resulting in gradual updates that prevent abrupt changes during the evolution of actions.

Notations of Key Variations: We quantify three types of temporal Variations inherent in the problem:

1. Variation of dynamic optimal decisions $\{\tilde{\mathbf{v}}_t^*\}$, i.e.,

$$H_v(T) := \sum_t \|\tilde{\mathbf{v}}_{t+1}^* - \tilde{\mathbf{v}}_t^*\|_2 \quad (13)$$

2. Variation of dual optimal solutions sequence $\{\lambda_t^*\}$, i.e.,

$$H_\lambda(T) := \sum_t \|\lambda_{t+1}^* - \lambda_t^*\|_2 \quad (14)$$

3. Variants of quality loss f_t , query queue and migration budget constraint $\mathbf{g}_t$, and squared variation of $\mathbf{g}_t$, i.e.,

$$H_f(T) := \sum_t \max_{\mathbf{v} \in \mathcal{X}} \{|f_{t+1}(\mathbf{v}) - f_t(\mathbf{v})|\}, \quad (15)$$

$$\tilde{H}_g(T) := \sum_t \max_{\mathbf{v} \in \mathcal{X}} \{|\mathbf{g}_t(\mathbf{v}) - \mathbf{g}_{t-1}(\mathbf{v})|\}, \quad (16)$$

$$H_g(T) := \sum_t \max_{\mathbf{v} \in \mathcal{X}} \{\|\mathbf{g}_t(\mathbf{v}) - \mathbf{g}_{t-1}(\mathbf{v})\|_2^2\}. \quad (17)$$

Lemma 2. In integral and real domains, based on the Eqs. (7)–(10), the dynamic regret and constraint violation hold

$$\forall i : \text{Reg}(T) \leq \widetilde{\text{Reg}}(T) + M\sigma_\beta, \text{Viol}_i(T) = \widetilde{\text{Viol}}_i(T), \quad (18)$$

where $M\sigma_\beta$ is a constant upper-bound in Jensen gap [36].

Proof. The proof is presented in [Appendix B](#). $\square$

Theorem 1 (Performance Bounds). In an integral domain, the dynamic regret and violation are upper-bounded as:

$$\text{Reg}(T) \leq H_f(T) + H_g(T) + 4\mu RH_v(T) + C_1 = R_T, \quad (19)$$

$$\begin{aligned} \text{Viol}_i(T) &\leq 4H_\lambda(T) + 2\sqrt{H_f(T)} + \sqrt{2H_g(T)} + \tilde{H}_g(T) \\ &\quad + \sqrt{8\mu RH_v(T)} + C_2 = H_T, \end{aligned} \quad (20)$$

where C_1, C_2 are constants.

Proof. The proof is presented in [Appendix C](#). $\square$

Corollary 1 (Sublinear Growth). With step size $\mu = \Theta(1)$, if the temporal variations satisfy $H_f(T) = o(T)$, $H_g(T) = o(T)$, $\tilde{H}_g(T) = o(T)$, $H_\lambda(T) = o(T)$, and $H_v(T) = o(T)$, then the dynamic regret and violation grow sublinearly:

$$\forall i : \text{Reg}(T) = o(T), \quad \text{Viol}_i(T) = o(T). \quad (21)$$

For a concrete regime, it is essential to distinguish between the variations that enter the violation bound H_T through a square root and those that enter it linearly. Suppose the former may be as large as linear, i.e., $H_f(T), H_g(T), H_v(T) = \mathcal{O}(T)$, while the dual-variable variation $H_\lambda(T)$ and the incremental (first-difference) constraint variation $\tilde{H}_g(T)$, which appear linearly, are sublinear at the square-root rate, i.e., $H_\lambda(T), \tilde{H}_g(T) = \mathcal{O}(\sqrt{T})$, with $\mu = \mathcal{O}(1)$. Then $2\sqrt{H_f(T)}$, $\sqrt{2H_g(T)}$ and $\sqrt{8\mu RH_v(T)}$ are each $\mathcal{O}(\sqrt{T})$, and $4H_\lambda(T)$, $\tilde{H}_g(T)$ are also $\mathcal{O}(\sqrt{T})$, so every term of H_T is $\mathcal{O}(\sqrt{T})$ and we obtain $\text{Viol}_i(T) \leq H_T = \mathcal{O}(\sqrt{T})$ (sublinear violation), while $\text{Reg}(T) \leq R_T = \mathcal{O}(T)$ (bounded average per-slot regret). The $\mathcal{O}(T)$ allowance on $H_g(T)$ is in fact conservative: being the squared variation of the same per-slot constraint increments as the first-difference path length $\tilde{H}_g(T)$, it satisfies $H_g(T) \leq D \tilde{H}_g(T) = \mathcal{O}(\sqrt{T})$ on the compact feasible set $\tilde{\mathcal{X}}$, where D bounds those increments, so its square-root term contributes only $\mathcal{O}(T^{1/4})$ and never controls the rate. We stress that controlling the linearly-entering variations $H_\lambda(T)$ and $\tilde{H}_g(T)$ to $\mathcal{O}(\sqrt{T})$ is necessary for the $\mathcal{O}(\sqrt{T})$ violation rate: were they allowed to grow as $\mathcal{O}(T)$, the bound H_T would itself be $\mathcal{O}(T)$ and sublinearity could not be claimed. This regime is mild, since $\tilde{H}_g(T)$ is a first-difference path length (not a squared variation) and $H_\lambda(T)$ is the drift of the dual optima, both of which stay sublinear whenever the constraint geometry (arrival statistics and budgets) drifts smoothly, even if the objective $\hat{\omega}_t$ and the primal optimum fluctuate substantially per slot.

Proof. The proof is presented in [Appendix D](#). $\square$

Remark: The achievable rates depend on the temporal variation of the problem inputs. When all variations are sublinear (e.g., the environment stabilizes over time, $H_f(T) = \mathcal{O}(T^{1/2})$), both regret and violation are strictly sublinear by the general case. The concrete regime above isolates the practically relevant situation in which the objective and primal optimum may vary quickly per slot (so $H_f(T), H_v(T) = \mathcal{O}(T)$ and the regret is only bounded per slot), yet the dual/constraint geometry drifts slowly ($H_\lambda(T), \tilde{H}_g(T) = \mathcal{O}(\sqrt{T})$), which is exactly the condition that keeps the violation at the sublinear $\mathcal{O}(\sqrt{T})$ rate. If, instead, the linearly-entering variations $H_\lambda(T)$ and $\tilde{H}_g(T)$ were themselves $\mathcal{O}(T)$, the violation bound would degrade to $\mathcal{O}(T)$; hence the $\mathcal{O}(\sqrt{T})$ claim is asserted only under the stated sublinear-drift condition, not for fully adversarial variation of every quantity.

Remark (Metric invariance and content sensitivity). All results above depend on the quality metric only through the boundedness of the loss $s_{m,t} \in [0, 1]$ ([Assumption 1](#)) and the variation quantities $H_f(T)$ and $H_g(T)$. They are therefore invariant to the specific perceptual metric used to define the loss: replacing SSIM by VMAF, LPIPS, or PSNR (suitably normalized to $[0, 1]$) only changes the numerical values of $s_{m,t}$, and hence of $\omega_{m,t} = s_{m,t}/p_m$, while leaving the convexity ([Lemma 1](#)), the rounding guarantees, and the regret/violation bounds ([Theorem 1](#)) structurally unchanged. Furthermore, content-dependent fluctuations of the metric—e.g., a model scoring differently on vlogs versus sports—are absorbed into the time-varying loss $s_{m,t}$ and quantified by $H_f(T) = \sum_t \max_{\mathbf{v}} |f_{t+1}(\mathbf{v}) - f_t(\mathbf{v})|$. Since $f_t(\mathbf{v}) = \sum_{n,m} \hat{\omega}_{m,t} y_{n,m,t}$, a content or metric shift that changes the per-slot loss of any model by at most η changes f_t by at most $\eta \max_m (1/p_m) \|\mathbf{y}\|_1 = \mathcal{O}(\eta R)$, contributing at most $\mathcal{O}(\eta R)$ to H_f per slot. Therefore, if such shifts are bounded per slot (so $H_f(T) = \mathcal{O}(T)$), the bounded-per-slot regret of [Corollary 1](#) continues to hold, and if they occur on only a sublinear number of slots (so $H_f(T) = o(T)$), the regret is strictly sublinear. In other words, metric instability degrades the guarantees only gracefully and quantifiably through $H_f(T)$, rather than catastrophically, and switching metrics changes only the constants, not the rates.

Table 2
Summary of used traces for evaluation.

Content	Description
Query and edges settings	Telecom Shanghai Dataset [27]
Edge-cloud bandwidth	WAN bandwidth data from Aliyun [37]
Inter-edge bandwidth	Four public throughput datasets in [38]
SR model selection	Repo. Awesome-Super-Resolution [28]

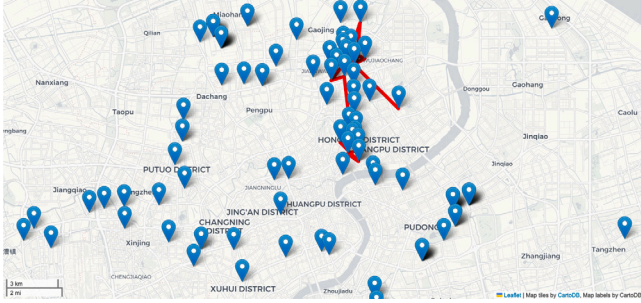


Fig. 2. Base station distribution and user mobility in dataset [27].

5. Implementation and evaluation

5.1. Data and settings

The employed real-world traces are presented in Table 2.

Edge Nodes, Cloud and Queries: As illustrated in Fig. 2, we leverage the Telecom Shanghai Dataset [27], containing over 7.2 million records documenting internet access activities of 9481 mobile devices across 3233 base stations over six months. Each record includes six parameters: month, date, start time, end time, base station location, and device ID. We preprocess this data by mapping the dynamic mobile device count at each base station to the number of SR queries originating there. Additionally, we calculate query duration as the interval between a mobile device's arrival and departure at a corresponding base station. These processed query metrics are aggregated into hourly measurements over 15 days. Consequently, our analysis considers a 360-time-slot half-month period. Dynamic WAN bandwidth data, required for loading the SR model, originates from Aliyun [37] and was measured by downloading files from the cloud. Furthermore, dynamic inter-edge bandwidth data for migrating queries utilizes four public real-world network throughput datasets from [38] to simulate diverse network conditions.

Super-Resolution Models: From github repository [28], we collect twenty representative state-of-the-art video super-resolution models, such as EGVSR [29], STDO [30], VIDIM [31], STAR [32], and StableVSR [33]. These models are executed on four dedicated servers (Dell PowerEdge R740 and Inspur SN5160M4), each equipped with NVIDIA GeForce RTX 2080 Ti GPUs. During this offline procedure, we measure their (1) processing capacity (defined as the maximum video duration in seconds that can be processed within a specific time slot), which ranged from 90 to 500 s/slot; (2) computational demand in GPU-core-hours, ranging from 150 to 1600; and (3) inference quality in PSNR, which scored between 28 dB and 39 dB, and in SSIM (unitless), which ranged from 0.82 to 0.97. The inference quality loss used in our optimization is $s_{m,t} = 1 - \text{SSIM}$, ranging from 0.03 to 0.18. We note that the content-dependent variability of these quality metrics across different video types (e.g., vlogs vs. sports) is captured by the time-varying nature of $s_{m,t}$ in our formulation. Alternative perceptual metrics such as VMAF or LPIPS can also be used in place of SSIM without changing the optimization structure. Additionally, we recorded the size of each model, varying from 100 MB to 1000 MB.

Baselines: Except for the online scheme we proposed, i.e., OASH, we also compare our scheme with other algorithms:

- **OAEI** [39] adopts a convex-concave equivalent formulation, and uses primal-dual based method to obtain the decision, but it does not consider the durations of queries and denotes variable y as the number of migrated queries. Unlike OASH, OAEI treats each query as a unit-cost task and cannot distinguish between short and long video queries, nor can it control per-query migration policies. This limits its ability to manage queue backlog dynamics and leads to suboptimal resource utilization when query durations are heterogeneous.
- **MaxPPR** hosts the model with the highest performance-price ratio¹ on each edge node, activates a carefully calculated number of instances based on historical query volumes, and migrates a predefined number of queries to other edge nodes. Unlike OASH, MaxPPR does not perform joint online optimization across model selection, instance deployment, and query migration. Its fixed migration policy and reliance on historical averages make it unable to adapt to sudden workload changes.
- **LoadAW** targets global workload balance through the dynamic migration of queries: increasing assignments to idle edge nodes and decreasing them to busy nodes. This allocation strategy relies on the queue state from the preceding time slot. It is important to note that LoadAW operates under a fixed model hosting policy. In contrast, OASH jointly optimizes model hosting and query migration, enabling it to adapt the deployed model portfolio in response to changing workload patterns and quality requirements.

In summary, OASH fundamentally differs from all baselines in that it (i) jointly optimizes all three decision variables ($x_{n,m,t}$, $y_{n,m,t}$, $z_{k,n,n',t}$) through online learning with provable guarantees; (ii) incorporates query durations into queue dynamics; and (iii) uses a virtual queue mechanism to track and compensate for long-term constraint violations, which the baselines do not provide.

5.2. Evaluation results

Fig. 3(a) compares the normalized total system inference loss per time slot across all algorithms. OASH (with step size $\mu = 0.5$) reduces average loss by at least 32.4% compared to other strategies. Its per-slot loss variations are also more stable. Although OAEI and MaxPPR occasionally achieve losses similar to OASH, their variations are significantly more severe, exhibiting larger peaks. We emphasize that OASH's computation overhead remains minimal, typically under 5 s for thousands of variables, making it highly suitable for 1-hour intervals handling hundreds of edges. This timing was measured on Dell PowerEdge R740 and Inspur SN5160M4 servers equipped with NVIDIA GeForce RTX 2080 Ti GPUs, Intel Xeon processors, and 64 GB RAM, using CVXPY with the MOSEK solver to solve the convex sub-problem $\tilde{\mathcal{P}}_3^{t+1}$. Its advantage over OAEI stems from incorporating query durations and controlling per-query migration policies. Fig. 3(b) shows the average inference loss over 360 time slots. OASH reduces at least 23.7%, 69.2%, and 42.8% inference quality loss compared to OAEI, LoadAW, and MaxPPR, respectively.

Figs. 3(c) and 3(d) depict dynamic violations and dynamic regrets for all algorithms. OASH outperforms other strategies in both metrics, achieving at least 35.1% and 41.6% reductions in their mean values, respectively. The figures visually confirm the sub-linear growth of OASH's constraint violation (consistent with the $\mathcal{O}(\sqrt{T})$ bound in Corollary 1) and bounded per-slot dynamic regret, aligning with our theoretical analysis. Unlike other approaches, OASH updates model instance deployments based solely on user queries from the preceding slot, maintaining an effective balance between the objective function and constraints. While other algorithms also exhibit sub-linear growth

¹ For a specific model, performance-price ratio (PPR) is defined as its processing ability divided by its resource consumption.

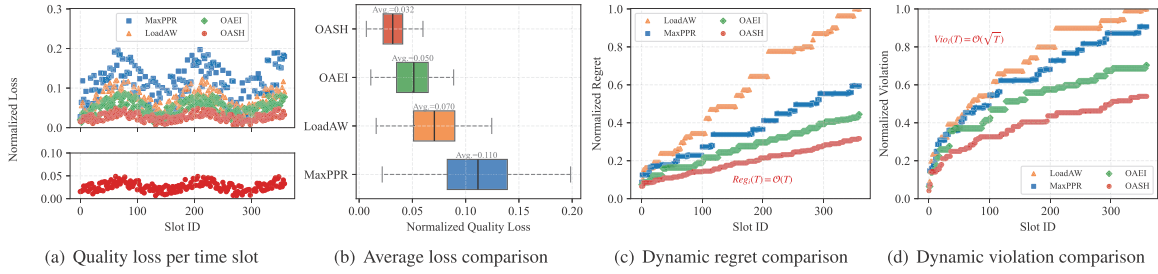


Fig. 3. Overall performance of OASH and other algorithms.

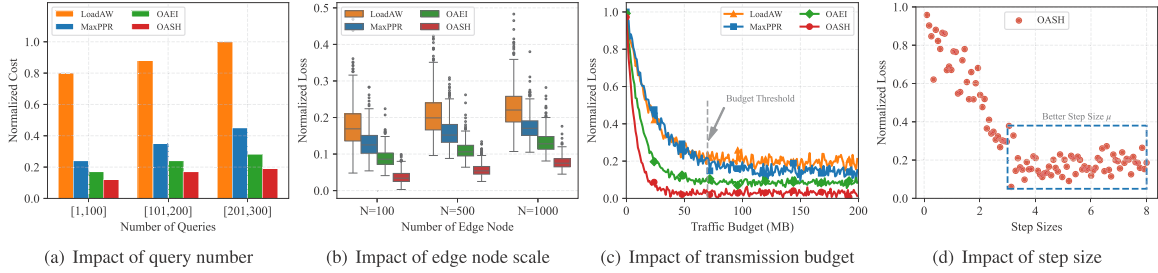
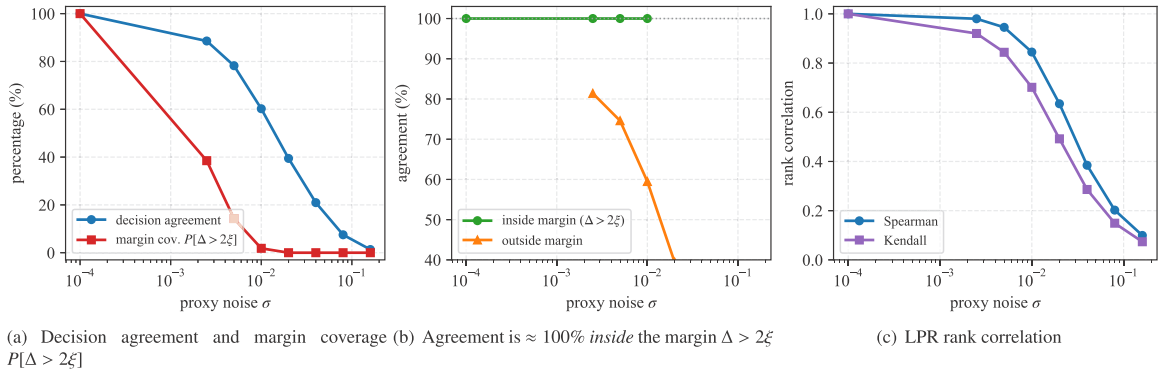


Fig. 4. Further performance under different settings.

Fig. 5. Order preservation of model selection under a no-reference proxy of varying noise σ . The selection agreement *inside* the margin set $\Delta_{n,t} > 2\xi_t$ is exactly 100% in every slot satisfying the margin condition, confirming the order-preservation remark, while overall agreement and rank correlation degrade gracefully.

in regret and violation, they demonstrate higher growth rates as OAEI and MaxPPR's query queue violations and LoadAW's traffic budget violations.

Fig. 4 presents performance under varying conditions. By controlling query arrivals per edge node via crowded base stations from dataset [27], Fig. 4(a) shows average inference quality loss. OASH consistently outperforms all strategies, achieving mean reductions of at least 25.2% (light queries), 27.5% (moderate queries), and 30.3% (heavy queries). Similarly, Fig. 4(b) demonstrates OASH's superiority when scaling edge nodes, with mean reductions of at least 28.1% (small-scale), 29.4% (medium-scale), and 32.5% (large-scale).

We also investigate the impact of traffic budget ($b_{n,t}$) on inference loss (Fig. 4(c)). Increasing budgets yield significant loss reduction across all algorithms, enabling more aggressive model downloads from the cloud and SR query migration. OASH maintains superiority regardless of budget level. Fig. 4(d) further shows OASH's mean cost across step sizes, confirming that our online schema selects effective values. Larger step sizes reduce constraint violations faster, while smaller step sizes enable more gradual adaptation.

5.3. Empirical validation of observability and metric robustness

To address the observability and metric-stability concerns directly, we add a focused empirical study. Unless noted, we use the trace-driven

setting of Section 5 and report averages over multiple seeds; each study sweeps its stressor across a range wide enough to *falsify* the claim, so the results are genuine sensitivity analyses rather than confirmatory demonstrations.

Observability (estimated loss vs. true SSIM). We model the estimated loss $\hat{s}_{m,t}$ as a monotone no-reference proxy of the true loss corrupted by content noise of level σ , and measure how often the proxy preserves the model-selection decision. Consistent with the order-preservation remark, the selection agreement *inside* the margin set $\Delta_{n,t} > 2\xi_t$ is exact: wherever the margin condition holds, the proxy selects the same model as the truth (100% agreement at every tested noise level for which the margin set is non-empty; Fig. 5b), while overall agreement and the LPR rank correlation degrade gracefully as σ grows (Fig. 5a,c). For the decision-relevant cumulative discrepancy $\theta_0(T)$ (the excess true cost of optimizing the proxy), a fixed proxy error yields a linear-in- T but small-per-slot contribution, whereas *improving* calibration over time ($\sigma_t \propto 1/\sqrt{t}$, realistic for accumulating offline data) makes $\theta_0(T)$ strictly sublinear (slope 0.13 vs. 1.12 for fixed σ ; Fig. 6), confirming that Assumption 3's $\theta_0(T) = o(T)$ is achievable. We further corroborate these findings on *real* SR data. We run five real upscalers (bicubic, FSRCNN, ESPCN, LapSRN, EDSR via OpenCV `dnn_superres`) on the Set14 benchmark and score each output with full-reference SSIM/PSNR/LPIPS and no-reference NIQE/BRISQUE

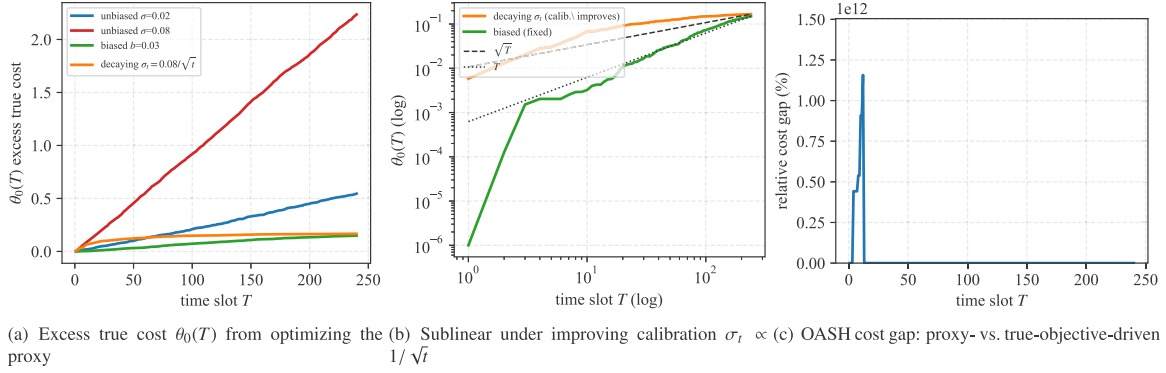


Fig. 6. Decision-relevant discrepancy $\theta_0(T)$ (excess true cost of optimizing the proxy). Under improving calibration $\theta_0(T)$ is strictly sublinear, whereas a fixed/biased proxy gives linear growth; the realized OASH cost gap stays small (near-optimality of $\hat{P}_1$ vs. P_1).

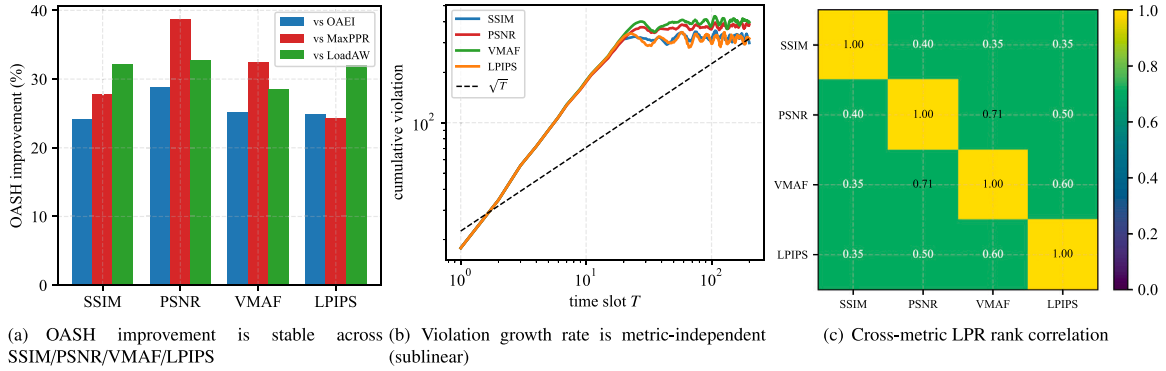


Fig. 7. Metric invariance. OASH's improvement over the baselines and the sublinear constraint-violation growth rate are both essentially unchanged when the quality metric is switched among SSIM, PSNR, VMAF and LPIPS, confirming that a metric swap changes only constants, not rates.

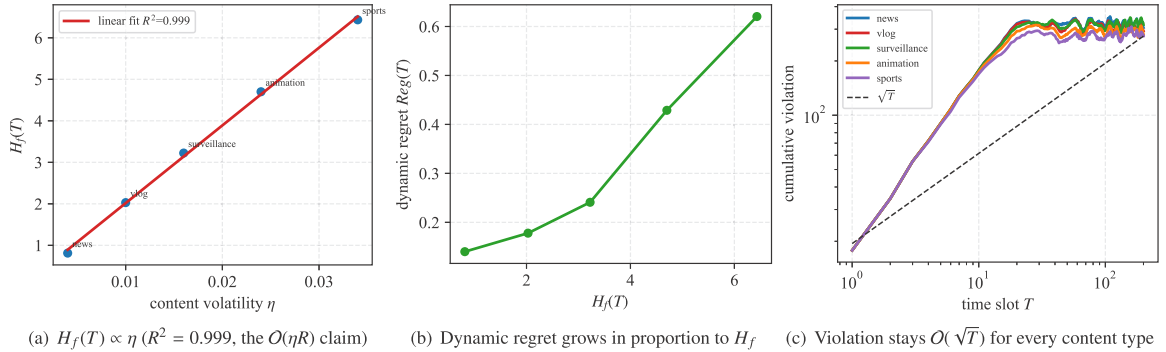


Fig. 8. Content sensitivity through H_f . Across content types from calm (news) to volatile (sports), $H_f(T)$ grows linearly in the per-slot volatility η and the regret grows in proportion, while the violation rate remains $\mathcal{O}(\sqrt{T})$ regardless of content—content enters only through H_f .

(pyiqa). Two observations follow. First, the full-reference metrics are *strongly* mutually consistent (pairwise Spearman 0.65–0.83 among SSIM, PSNR and LPIPS), and all three rank EDSR best: this directly supports the metric-invariance claim, since swapping among these quality metrics preserves the model ordering. Second, the no-reference proxies are usable but imperfect: BRISQUE correlates moderately with SSIM (Spearman 0.70) and selects the SSIM-best model in 66.7% of images, whereas NIQE is weaker (0.30). This imperfect agreement is precisely why we do *not* assume the proxy equals the true objective but instead bound their discrepancy (Assumption 3) and rely on the order-preservation margin: when a reference is available server-side, SSIM metadata can be used directly, and otherwise the bounded-discrepancy/order-preservation guarantees (Figs. 5–6) ensure graceful behavior. Fig. 9 reports these real-data correlations.

Metric robustness. We re-define the loss $s_{m,t}$ from each of SSIM, PSNR, VMAF and LPIPS and re-run the full pipeline. OASH's improvement over the strongest baseline is stable across metrics (24.1–28.9% over OAEI; spread only 4.8 points), and the constraint-violation growth rate stays sublinear and essentially metric-independent (slope 0.25–0.32; Fig. 7), empirically confirming that switching the perceptual metric changes only constants, not rates. Finally, content-dependent volatility enters solely through H_f : across content types from calm (news) to volatile (sports), $H_f(T)$ grows *linearly* in the per-slot volatility η ($R^2 = 0.999$), the dynamic regret grows in proportion to H_f , and the violation rate remains $\mathcal{O}(\sqrt{T})$ regardless of content (slope 0.22–0.25; Fig. 8). Thus metric/content instability degrades the guarantees only gracefully and quantifiably through H_f , as claimed.

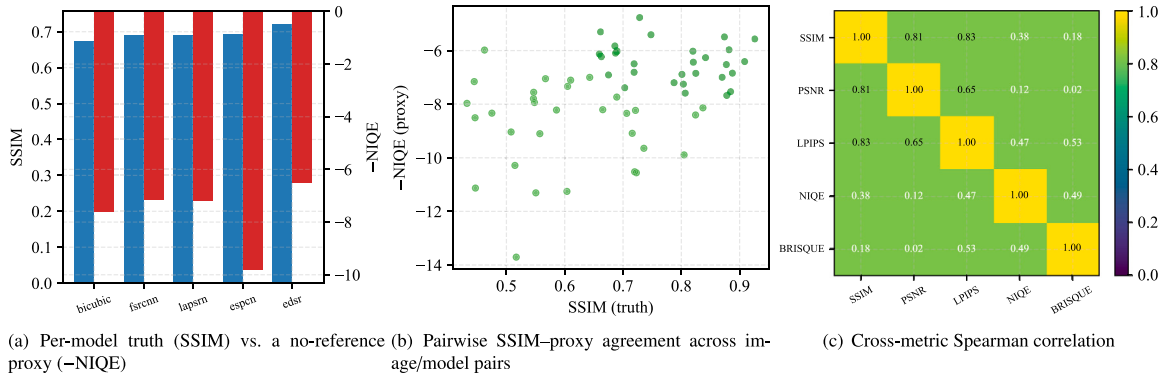


Fig. 9. Five real upscalers (bicubic/FSRCNN/ESPCN/LapSRN/EDSR) on Set14 scored with *pyiqa*. The full-reference metrics SSIM/PSNR/LPIPS are strongly mutually consistent (all rank EDSR best), while the no-reference proxies are usable but imperfect—motivating the bounded-discrepancy treatment rather than assuming a perfect proxy.

6. Related work

We categorize prior research into two groups, summarizing each and highlighting their limitations relative to our approach.

Video Super-Resolution at Edge: Rosevin [8] handled bitrate and resource allocation decisions for distributed end-users. To accelerate inference, approaches like EVASR [9], ViChaser [40] and TileSR [10] selectively performed super-resolution (SR) on high-saliency or high-variation image patches at the edge; Liu et al. [11] used enhanced embedded FPGAs and DSPs; ECBSR [12] introduced a novel SR network architecture featuring an edge-oriented convolution block; ESHP [13] and MobiSR [14] leveraged heterogeneous computational resources (e.g., CPU, GPU, NPU) within a mobile device. Several other studies like SRSC [15], ECCSR [16], SRA360 [17], and TBSR [18] also focused on optimizing mobile device or single-edge-server resource utilization, particularly for generating volumetric or 360-degree videos. Finally, EADV [19] and Scrava [20] utilized mobile device resources to perform SR for specialized applications like surveillance and video analytics.

These researches often run video super-resolution directly on individual devices or isolated edge servers, rarely optimizing inference as a service within distributed edge infrastructures for large-scale users. In contrast, we address edge SR service provisioning online, designing algorithms with rigorously provable performance.

Online Service Provisioning at Edge: To optimize service placement, MOSP [21] addressed spatial demand coupling, service heterogeneity, and decentralized coordination in MEC systems, Liu et al. [22] proposed a parameterized deep Q-network approach, and Gao et al. [23] designed a two-phase algorithm based on matching and game theory. In addition, Chen et al. [24] introduces a two-timescale approach to dynamically optimize satellite access selection, user request dispatching, and service replica placement, Zhang et al. [25] develops an online algorithm for dynamic DT replica placement, incorporating a prediction mechanism that reserves future DT replicas using timestamps, and Hu et al. [26] utilizes alternating optimization and quadratic transform methods for efficient resource allocation and task scheduling.

These researches primarily address service provisioning, not video super-resolution. Furthermore, their algorithmic approaches remain inadequate for the specific challenges tackled in our study, as few incorporate holistic online decision-making. By contrast, our work successfully addresses time-coupled ramp constraints and queueing state-transition constraints within an online learning framework, while rigorously bounding long-term constraint violations. While the high-level optimization template (virtual queue + primal–dual decomposition) shares similarities with prior online service provisioning works, our formulation differs in three key aspects: (i) the LPR objective $\omega_{m,t} = s_{m,t}/p_m$ captures a quality–throughput trade-off inherent to SR model heterogeneity, where higher-quality models have lower processing rates; (ii) duration-aware queueing dynamics model video-specific

backlog in seconds rather than treating requests as unit-cost tasks; and (iii) the two-step randomized rounding procedure handles the coupling between deployment intensity and capacity constraints, which is absent in standard independent rounding approaches used in prior works.

7. Conclusion

Enabling edge VSR as a service over mobile edge computing infrastructures to large-scale, geographically distributed users represents a critical step toward universal artificial intelligence. To maximize the overall inference quality of these edge services, we formulate the problem as an online non-linear integer program, subject to challenging constraints including time-coupling restrictions, uncertainty in inputs, and discrete (integral) decision variables. We design an online algorithm combining an online learning component with a randomized rounding component to address these challenges. We rigorously prove that our approach achieves sub-linear dynamic regret and maintains dynamic violation. Finally, we validate the practical superiority of our solution through trace-driven evaluations and comparisons against baseline algorithms.

Beyond VSR, the core algorithmic framework of OASH is built on three abstract components: a time-varying loss function $f_t(v_t)$, long-term constraint functions $g_t(v_t)$, and per-slot hard constraints $h(v_t)$. This formulation can be applied to other edge-based stateful inference services, such as edge-based large language model (LLM) serving, edge-based image generation with diffusion models, and federated inference with model heterogeneity. Adapting the framework to a new application mainly involves redefining the loss function f_t and constraint parameters according to domain-specific quality metrics, resource requirements, and query characteristics; the online learning and rounding components remain applicable as-is.

CRedit authorship contribution statement

Ning Chen: Writing – original draft, Visualization, Methodology. **He Huang:** Validation, Supervision, Data curation. **Sheng Zhang:** Supervision, Investigation, Conceptualization. **Jie Wu:** Supervision, Investigation, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Proof of Lemma 1

Proof. To establish that $\tilde{\mathcal{P}}_3$ is a convex optimization problem, we verify the convexity of its objective f_t and constraints $\mathbf{g}_t$ and $\mathbf{v}$. The objective $f_t(\mathbf{v}_t) := \left[\mathbf{0}_{NM \times 1}^\top, \hat{\mathbf{S}}_t^\top, \mathbf{0}_{NM \times K}^\top \right] \mathbf{v}_t$ simplifies to $\hat{\mathbf{S}}_{m,t} y_{n,m,t}$, which is linear in $y_{n,m,t}$ and thus concave in $\mathbf{v}_t$. For constraints, $g_t^{1,n}$ and $g_t^{2,n}$ are linear in $z_{k,n',t}$, $g_t^{3,n}$ combines linear components for incoming queries $a_{n,t} = \sum_{n'} \sum_k z_{k,n',t} d_{n',k,t}$ and processing reductions $l_{n,t} = \sum_m p_m y_{n,m,t}$, while $g_t^{4,n}$ sums linear functions in $x_{n,m,t}$ and $z_{k,n,n',t}$, making all $\mathbf{g}_t$ components linear (hence convex). Since $\mathbf{h}$ consists of affine functions and $\tilde{\mathbf{v}}_t$'s domain is convex, $\tilde{\mathcal{P}}_3$ satisfies all requirements: concave objective, convex constraints, affine equalities, and convex domain, confirming it is a convex optimization problem. $\square$

Appendix B. Proof of Lemma 2

Proof. First, observe that $\tilde{\mathbf{v}}_t$ is a random vector with $\mathbb{E}[\tilde{\mathbf{v}}_t] = \hat{\mathbf{v}}_t$. Applying definition (7), we derive:

$$\begin{aligned} \text{Reg}(T) &\stackrel{(22a)}{\leq} \mathbb{E} \left[\sum_t f_t(\hat{\mathbf{v}}_t) \right] - \sum_t f_t^G(\tilde{\mathbf{v}}_t^*) \\ &\stackrel{(22b)}{\leq} \sum_t f_t(\tilde{\mathbf{v}}_t) + M\sigma_\beta - \sum_t f_t^G(\tilde{\mathbf{v}}_t^*) \\ &\stackrel{(22c)}{=} \widetilde{\text{Reg}}(T) + M\sigma_\beta, \end{aligned}$$

where (22a) follows because the real-valued optimum dominates the integral optimum, (22b) applies the Jensen gap from [36], with $M\sigma_\beta$ bounding variations in f_t , and (22c) uses definition (8). Similarly, via definition (9), $\forall i$:

$$\begin{aligned} \text{Vio}_i(T) &= \mathbb{E} \left[\sum_t g_{t,i}(\hat{\mathbf{v}}_t) \right] = \sum_t \mathbb{E} [g_{t,i}(\hat{\mathbf{v}}_t)] \\ &\stackrel{(23a)}{=} \sum_{t=1}^T g_{t,i}(\mathbb{E}[\hat{\mathbf{v}}_t]) = \sum_t g_{t,i}(\tilde{\mathbf{v}}_t) = \widetilde{\text{Vio}}_i(T), \end{aligned}$$

with equality (23a) justified by the linearity of $g_{t,i}$ and the expectation operator. $\square$

Appendix C. Proof of Theorem 1

Proof. Bound $\text{Reg}(T)$: Fix any $i = 1, \dots, 4N$, then $Q_{0,i} = 0$, which is nonnegative. Proceeding by induction, assume $Q_{t,i} \geq 0$ for some $t \geq 0$. We establish nonnegativity for $t+1$ in two cases: If $g_{t,i}(\mathbf{v}_{t+1}) \geq 0$, then (6) implies

$$Q_{t+1,i} \geq Q_{t,i} + g_{t,i}(\mathbf{v}_{t+1}) \geq 0; \quad (\text{C.1})$$

If $g_{t,i}(\mathbf{v}_{t+1}) < 0$, then $Q_{t+1,i} \geq -g_{t,i}(\mathbf{v}_{t+1}) > 0$. Thus, $Q_{t+1,i} \geq 0$ holds. Furthermore, for all $t \geq 0$, we have

$$Q_{t+1,i} + g_{t,i}(\mathbf{v}_{t+1}) \geq 0, \text{ i.e., } Q_{t+1,i} \geq -g_{t,i}(\mathbf{v}_{t+1}) \quad (\text{C.2})$$

with the $t = -1$ case being trivial. Finally, for $t \geq 0$: When $g_{t,i}(\mathbf{v}_{t+1}) \geq 0$, we have

$$Q_{t+1,i} \geq Q_{t,i} + g_{t,i}(\mathbf{v}_{t+1}) \stackrel{(a)}{=} g_{t,i}(\mathbf{v}_{t+1}) = |g_{t,i}(\mathbf{v}_{t+1})| \quad (\text{C.3})$$

Similarly, when $g_{t,i}(\mathbf{v}_{t+1}) < 0$,

$$Q_{t+1,i} \geq -g_{t,i}(\mathbf{v}_{t+1}) = |g_{t,i}(\mathbf{v}_{t+1})|. \quad (\text{C.4})$$

Consequently, $Q_{t+1,i} \geq |g_{t,i}(\mathbf{v}_{t+1})|$ universally, implying $Q_{t+1,i}^2 \geq g_{t,i}^2(\mathbf{v}_{t+1})$. Summing over k gives

$$\|\mathbf{Q}_{t+1}\|_2 \geq \|\mathbf{g}_t(\mathbf{v}_{t+1})\|_2. \quad (\text{C.5})$$

Based on Lemma 2–4 in [34], we derive

$$\begin{aligned} \sum_t f_t(\tilde{\mathbf{v}}_t) - \sum_t f_t(\tilde{\mathbf{v}}_t^*) &\leq \frac{1}{2} \|\mathbf{g}_T(\tilde{\mathbf{v}}_{T+1})\|_2^2 - \frac{1}{2} \|\mathbf{Q}_{T+1}\|_2^2 \\ + H_f(T) + H_g(T) + 4\mu RH_x(T) + \|\mathbf{g}_1(\tilde{\mathbf{v}}_1)\|_2^2 + F + f_1(\tilde{\mathbf{v}}_1) \\ + \|\tilde{\mathbf{v}}_1 - \tilde{\mathbf{v}}_1^*\|_2^2 &\stackrel{(27a)}{\leq} H_f(T) + H_g(T) + 4\mu RH_x(T) + C_0, \end{aligned} \quad (\text{C.6})$$

where Inequality (27a) meets by using (26) and denoting the sum of constants as C_0 . Similarly, we can derive that

$$\begin{aligned} \widetilde{\text{Reg}}(T) &= \sum_t [f_t(\tilde{\mathbf{v}}_t) - f_t^G(\tilde{\mathbf{v}}_t^*)] \\ &= \sum_t [f_t(\tilde{\mathbf{v}}_t) - f_t(\tilde{\mathbf{v}}_t^*)] + [f_t(\tilde{\mathbf{v}}_t^*) - f_t^G(\tilde{\mathbf{v}}_t^*)] \\ &\stackrel{(28a)}{\leq} H_f(T) + H_g(T) + 4\mu RH_x(T) + C_0 + \theta_0, \end{aligned}$$

where Inequality (28a) holds based on Inequality (27) and previous Assumption 3. According to Lemma 2, we get

$$\text{Reg}(T) \leq \widetilde{\text{Reg}}(T) + M\sigma_\beta \leq H_f(T) + H_g(T) + 4\mu RH_x(T) + C_1,$$

where the constant C_1 is the sum of C_0 , θ_0 , and $M\sigma_\beta$.

Bound Vio_i^d : Based on Lemma 6 in [34], we get

$$\begin{aligned} \|\mathbf{Q}_T\|_2 &\stackrel{(29a)}{\leq} 4H_\lambda(T) + 2\sqrt{H_f(T)} + \sqrt{2H_g(T)} + \sqrt{8\mu RH_v(T)} \\ &\quad + \underbrace{\sqrt{G^2 + 4F + 2\mu\|\tilde{\mathbf{v}}_1 - \tilde{\mathbf{v}}_1^*\|_2^2 + 2\|\mathbf{g}_1(\tilde{\mathbf{v}}_1)\|_2^2 + 2\|\mathbf{A}_1^*\|_2^2}}_{\text{Constant } C'_0}. \end{aligned}$$

Then based on Lemma 2, we can derive

$$\begin{aligned} \text{Vio}_i(T) &= \widetilde{\text{Vio}}_i(T) = \sum_t g_{t,i}(\tilde{\mathbf{v}}_t) \leq \sum_t g_{t,i}(\tilde{\mathbf{v}}_{t+1}) \\ &\quad + g_{1,i}(\tilde{\mathbf{v}}_1) + \sum_t |g_{t+1,i}(\tilde{\mathbf{v}}_{t+1}) - g_{t,i}(\tilde{\mathbf{v}}_{t+1})| \\ &\stackrel{(30a)}{\leq} g_{1,i}(\tilde{\mathbf{v}}_1) + Q_{T,i} + \tilde{H}_g(T) \leq g_{1,i}(\tilde{\mathbf{v}}_1) + \|\mathbf{Q}_T\|_2 + \tilde{H}_g(T) \\ &\stackrel{(30b)}{\leq} 4H_\lambda(T) + 2\sqrt{H_f(T)} + \sqrt{2H_g(T)} + \sqrt{8\mu RH_v(T)} + C_2, \end{aligned}$$

where Inequality (30a) meets as the virtue queue updating rule; Inequality (30b) holds by using Inequality (29a) and denoting the sum of C'_0 and $g_{1,k}(\tilde{\mathbf{v}}_1)$ as a constant C_2 . $\square$

Appendix D. Proof of Corollary 1

Proof. General Case: With step size $\mu = \Theta(1)$, substituting into Theorem 1, each term in R_T and H_T is bounded by the corresponding variation. If all variations grow sublinearly, i.e., $H_f(T) = o(T)$, $H_g(T) = o(T)$, $\tilde{H}_g(T) = o(T)$, $H_\lambda(T) = o(T)$, and $H_v(T) = o(T)$, then every term in both R_T and H_T is $o(T)$, yielding $\text{Reg}(T) = o(T)$ and $\text{Vio}_i(T) = o(T)$.

Concrete case (corrected regime): A careful inspection of the violation bound H_T in Theorem 1 shows that its terms enter through two distinct functional forms: the variations $H_f(T)$, $H_g(T)$ and $H_v(T)$ enter through a square root ($2\sqrt{H_f(T)}$, $\sqrt{2H_g(T)}$, $\sqrt{8\mu RH_v(T)}$), whereas the dual variation $H_\lambda(T)$ and the incremental constraint variation $\tilde{H}_g(T)$ enter linearly ($4H_\lambda(T)$, $\tilde{H}_g(T)$). Consequently, to certify a sub-linear $\mathcal{O}(\sqrt{T})$ violation, it is *not* sufficient (and, indeed, would be self-contradictory) to assume that every variation grows as $\mathcal{O}(T)$: if $H_\lambda(T) = \Theta(T)$ or $\tilde{H}_g(T) = \Theta(T)$, then $4H_\lambda(T) + \tilde{H}_g(T) = \Theta(T)$ and hence $H_T = \Theta(T)$, giving only $\text{Vio}_i(T) = \Theta(T)$. We therefore state the concrete regime precisely. Assume

$$H_f(T), H_g(T), H_v(T) = \mathcal{O}(T), \quad H_\lambda(T), \tilde{H}_g(T) = \mathcal{O}(\sqrt{T}),$$

with $\mu = \mathcal{O}(1)$. Then, term by term in Theorem 1:

- **Regret.** $R_T = H_f(T) + H_g(T) + 4\mu RH_v(T) + C_1 = \mathcal{O}(T)$, i.e., the cumulative regret is at most linear, equivalently the average per-slot regret R_T/T is bounded.

- **Violation.** The square-root terms satisfy $2\sqrt{H_f(T)} = \mathcal{O}(\sqrt{T})$, $\sqrt{2H_g(T)} = \mathcal{O}(\sqrt{T})$ and $\sqrt{8\mu RH_v(T)} = \mathcal{O}(\sqrt{T})$; the linearly-entering terms satisfy $4H_\lambda(T) = \mathcal{O}(\sqrt{T})$ and $\tilde{H}_g(T) = \mathcal{O}(\sqrt{T})$ by hypothesis; and $C_2 = \mathcal{O}(1)$. Adding these, $H_T = \mathcal{O}(\sqrt{T})$, and therefore $V_{io_i}(T) \leq H_T = \mathcal{O}(\sqrt{T})$.

The sublinear-drift hypothesis on $H_\lambda(T)$ and $\tilde{H}_g(T)$ is mild and self-consistent: $\tilde{H}_g(T) = \sum_i \max_v |g_i(v) - g_{i-1}(v)|$ is a *first-difference* path length (in contrast to the *squared* variation $H_g(T)$, which is built from the *same* per-slot increments; since these increments are bounded by some D on the compact feasible set $\tilde{\mathcal{X}}$, one has $H_g(T) \leq D\tilde{H}_g(T)$, so the assumed allowance $H_g(T) = \mathcal{O}(T)$ is conservative and indeed $H_g(T) = \mathcal{O}(\sqrt{T})$ under the present hypothesis), and by the Cauchy–Schwarz inequality $\tilde{H}_g(T) \leq \sqrt{T \sum_i \max_v \|g_i(v) - g_{i-1}(v)\|^2}$, so $\tilde{H}_g(T) = \mathcal{O}(\sqrt{T})$ holds whenever the constraints change appreciably in only $\mathcal{O}(\sqrt{T})$ slots and are near-stationary otherwise (e.g., piecewise-stable arrival statistics and budgets); under the same smooth-drift condition the dual optima λ_i^* move slowly, giving $H_\lambda(T) = \mathcal{O}(\sqrt{T})$. Crucially, this regime does *not* require the objective or the primal optimum to be slowly varying: $H_f(T)$ and $H_v(T)$ may be $\mathcal{O}(T)$, which is precisely why the regret is only bounded per slot rather than strictly sublinear. When, in addition, the environment stabilizes so that all variations are sublinear (e.g., $H_f(T) = \mathcal{O}(T^{1/2})$), both regret and violation become strictly sublinear, recovering the general case. $\square$

Data availability

The authors do not have permission to share data.

References

- [1] Youtube official website <https://www.youtube.com/>.
- [2] Netflix official website <https://www.netflix.com/>.
- [3] iQIYI official website <https://www.iq.com/>.
- [4] Z. Ning, H. Hu, X. Wang, L. Guo, S. Guo, G. Wang, X. Gao, Mobile edge computing and machine learning in the internet of unmanned aerial vehicles: a survey, *ACM Comput. Surv.* 56 (1) (2023) 1–31.
- [5] M. Reiss-Mirzaei, M. Ghoabaei-Arani, L. Esmaili, A review on the edge caching mechanisms in the mobile edge computing: A social-aware perspective, *Internet Things* 22 (2023) 100690.
- [6] L. Kong, J. Tan, J. Huang, G. Chen, S. Wang, X. Jin, P. Zeng, M. Khan, S.K. Das, Edge-computing-driven internet of things: A survey, *ACM Comput. Surv.* 55 (8) (2022).
- [7] Q. Luo, S. Hu, C. Li, G. Li, W. Shi, Resource scheduling in edge computing: A survey, *IEEE Commun. Surv. & Tutorials* 23 (4) (2021) 2131–2165.
- [8] X. Zhang, H. Xu, L. Zou, J. Duan, C. Wu, Y. Xue, Z. Chen, X. Chen, Rosevin: Employing resource- and rate-adaptive edge super-resolution for video streaming, in: *Proceedings of IEEE Conference on Computer Communications, INFOCOM, 2024*, pp. 491–500.
- [9] N. Li, Y. Liu, EVASR: Edge-based video delivery with salience-aware super-resolution, in: *Proceedings of the 14th ACM Multimedia Systems Conference (MMsys)*, 2023, pp. 142–152.
- [10] N. Chen, S. Zhang, Y. Liang, J. Wu, Y. Chen, Y. Yan, Z. Qian, S. Lu, TileSR: Accelerate on-device super-resolution with parallel offloading in tile granularity, in: *Proceedings of IEEE Conference on Computer Communications, INFOCOM, 2024*, pp. 2538–2547.
- [11] H. Liu, Y. Qian, Y. Liang, B. Zhang, Z. Liu, T. He, W. Zhao, J. Lu, B. Yu, A high-performance accelerator for real-time super-resolution on edge FPGAs, *ACM Trans. Des. Autom. Electron. Syst.* 29 (3) (2024) 1–25.
- [12] X. Zhang, H. Zeng, L. Zhang, Edge-oriented convolution block for real-time super resolution on mobile devices, in: *Proceedings of the 29th ACM International Conference on Multimedia, MM, 2021*, pp. 4034–4043.
- [13] Q. Wang, W. Fang, L. Qian, Y. Chen, N.N. Xiong, An intelligent co-scheduling framework for efficient super-resolution on edge platforms with heterogeneous processors, *IEEE Internet Things J.* 11 (10) (2024) 17651–17662.
- [14] R. Lee, S.I. Venieris, L. Dudziak, S. Bhattacharya, N.D. Lane, MobiSR: Efficient on-device super-resolution through heterogeneous mobile processors, in: *Proceedings of Annual International Conference on Mobile Computing and Networking, MobiCom, 2019*, pp. 1–16.
- [15] J. Li, D. Xu, Z. Fan, J. Wang, X. Wang, Edge-assisted super resolution for volumetric video enhancement, in: *Proceedings of IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, 2024*, pp. 1–6.
- [16] J. Zeng, X. Zhou, K. Li, Toward high-quality low-latency 360° video streaming with edge-client collaborative caching and super-resolution, *IEEE Internet Things J.* 11 (17) (2024) 29020–29034.
- [17] Z. Wu, L. Lu, X. Chen, Y. Ma, S. Yang, M. Wang, L. Zhong, D.O. Wu, C. Xu, SRA360: Super-resolution enhanced adaptive 360-degree video streaming for heterogeneous viewers, *IEEE Trans. Netw. Sci. Eng.* (2025) 1–14.
- [18] L. Zhang, H. Zhou, H. Wang, L. Cui, TBSR: Tile-based 360° video streaming with super-resolution on commodity mobile devices, in: *Proceedings of IEEE Conference on Computer Communications, INFOCOM, 2024*, pp. 501–510.
- [19] L. Ge, W. Bao, D. Yuan, B.B. Zhou, Edge-assisted deep video denoising and super-resolution for real-time surveillance at night, in: *Proceedings of the 28th Annual International Conference on Mobile Computing and Networking, MobiCom, 2022*, pp. 783–785.
- [20] Y. Liang, S. Zhang, J. Wu, Scrava: Super resolution-based bandwidth-efficient cross-camera video analytics, *IEEE Trans. Mob. Comput.* 24 (1) (2025) 293–305.
- [21] L. Wang, X. Deng, J. Gui, X. Chen, S. Wan, Microservice-oriented service placement for mobile edge computing in sustainable internet of vehicles, *IEEE Trans. Intell. Transp. Syst.* 24 (9) (2023) 10012–10026.
- [22] T. Liu, S. Ni, X. Li, Y. Zhu, L. Kong, Y. Yang, Deep reinforcement learning based approach for online service placement and computation resource allocation in edge computing, *IEEE Trans. Mob. Comput.* 22 (7) (2023) 3870–3881.
- [23] B. Gao, Z. Zhou, F. Liu, F. Xu, B. Li, An online framework for joint network selection and service placement in mobile edge computing, *IEEE Trans. Mob. Comput.* 21 (11) (2022) 3836–3851.
- [24] S. Chen, L. Jiao, K. Zhu, L. Zhang, Online satellite selection, request dispatching, and service provisioning in LEO edge constellations, *IEEE Trans. Netw. Sci. Eng.* (2025).
- [25] Y. Zhang, W. Liang, Z. Xu, X. Jia, Mobility-aware service provisioning in edge computing via digital twin replica placements, *IEEE Trans. Mob. Comput.* 23 (12) (2024) 11295–11311.
- [26] H. Hu, K. Song, C. Zhan, R. Fan, J. Yang, Joint service caching and resource allocation over different timescales in satellite edge computing networks, *IEEE Trans. Mob. Comput.* 24 (7) (2025) 5649–5664.
- [27] Telecom shanghai dataset <https://www.kaggle.com/datasets/mexwell/telecom-shanghai-dataset>.
- [28] Awesome-super-resolution <https://github.com/ChaofWang/Awesome-Super-Resolution>.
- [29] Y. Lu, Z. Wang, M. Liu, H. Wang, L. Wang, Learning spatial-temporal implicit neural representations for event-guided video super-resolution, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2023*, pp. 1557–1567.
- [30] G. Li, J. Ji, M. Qin, W. Niu, B. Ren, F. Afghah, L. Guo, X. Ma, Towards high-quality and efficient video super-resolution via spatial-temporal data overfitting, in: *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2023*, pp. 10259–10269.
- [31] S. Jain, D. Watson, E. Tabellion, B. Poole, J. Kontkanen, et al., Video interpolation with diffusion models, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2024*, pp. 7341–7351.
- [32] R. Xie, Y. Liu, P. Zhou, C. Zhao, J. Zhou, K. Zhang, Z. Zhang, J. Yang, Z. Yang, Y. Tai, Star: Spatial-temporal augmentation with text-to-video models for real-world video super-resolution, 2025, arXiv preprint arXiv:2501.02976.
- [33] C. Rota, M. Buzzelli, J. van de Weijer, Enhancing perceptual quality in video super-resolution through temporally-consistent detail synthesis using diffusion models, in: *Proceedings of European Conference on Computer Vision, ECCV, Springer, 2024*, pp. 36–53.
- [34] X. Cao, J. Zhang, H.V. Poor, A virtual-queue-based algorithm for constrained online convex optimization with applications to data center resource allocation, *IEEE J. Sel. Top. Signal Process.* 12 (4) (2018) 703–716.
- [35] R. Gandhi, S. Khuller, S. Parthasarathy, A. Srinivasan, Dependent rounding and its applications to approximation algorithms, *J. ACM* 53 (3) (2006) 324–360.
- [36] S. Khan, M. Adil Khan, S.I. Butt, Y.-M. Chu, A new bound for the jensen gap pertaining twice differentiable functions with applications, *Adv. Difference Equ.* 2020 (1) (2020) 333.
- [37] Bandwidth from edges to aliyun www.aliyun.com/.
- [38] N. Kan, Y. Jiang, C. Li, W. Dai, J. Zou, H. Xiong, Improving generalization for neural adaptive video streaming via meta reinforcement learning, in: *Proceedings of the 30th ACM International Conference on Multimedia, MM, 2022*, pp. 3006–3016.
- [39] Y. Jin, L. Jiao, Z. Qian, S. Zhang, N. Chen, S. Lu, X. Wang, Provisioning edge inference as a service via online learning, in: *Proceedings of Annual IEEE International Conference on Sensing, Communication, and Networking, SECON, 2020*, pp. 1–9.
- [40] N. Chen, S. Zhang, Z. Ma, Y. Chen, Y. Jin, J. Wu, Z. Qian, Y. Liang, S. Lu, Vichaser: chase your viewpoint for live video streaming with block-oriented super-resolution, *IEEE/ACM Transactions on Networking* 32 (1) (2024) 445–459.



Ning Chen received the Ph.D. degree from Nanjing University, and is currently a Lecturer in the School of Computer Science and Technology, Soochow University. His research interests include edge computing, deep reinforcement learning, and video streaming. To date, he has published over 20 papers, including those appeared in INFOCOM, ICDE, TON, TPDS, SECON, Computer Networks, ICPADS, et al.



He Huang (Senior Member, IEEE) received the Ph.D. degree from the School of Computer Science and Technology, University of Science and Technology of China (USTC), China, in 2011. He is currently a professor with the School of Computer Science and Technology, Soochow University, China. From 2019 to 2020, he was a visiting research scholar with Florida University, Gainesville. He has authored more than 100 papers in related international conference proceedings and journals. His current research interests include traffic measurement, computer networks, and algorithmic game theory. He is a member of the Association for Computing Machinery (ACM). He received the best paper awards from Bigcom 2016, IEEE MSN 2018, and Bigcom 2018. He has served as the Technical Program Committee member of several conferences, including IEEE INFOCOM, IEEE MASS, IEEE ICC, and IEEE Globecom.



Sheng Zhang is an associate professor in the Department of Computer Science and Technology, Nanjing University. He is also a member of the State Key Lab. for Novel Software Technology. He received the BS and Ph.D. degrees from Nanjing University in 2008 and 2014, respectively. His research interests include cloud computing and edge computing. To date, he has published more than 80 papers, including those appeared in *JSAC*, *TMC*, *TON*, *TPDS*, *TC*, *MobiHoc*, *ICDCS*, and *INFOCOM*. He received the Best Paper Award of IEEE ICCN 2020 and the Best Paper Runner-Up



Award of IEEE MASS 2012. He is the recipient of the 2020 ACM Nanjing Rising Star Award and the 2015 ACM China Doctoral Dissertation Nomination Award. He is a senior member of IEEE and CCF, and a member of ACM.

Jie Wu is the Director of the Center for Networked Computing and Laura H. Carnell professor at Temple University. He also serves as the Director of International Affairs at College of Science and Technology. He served as Chair of Department of Computer and Information Sciences from the summer of 2009 to the summer of 2016 and Associate Vice Provost for International Affairs from the fall of 2015 to the summer of 2017. Prior to joining Temple University, he was a program director at the National Science Foundation and was a distinguished professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. Dr. Wu regularly publishes in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE Transactions on Mobile Computing, IEEE Transactions on Service Computing, Journal of Parallel and Distributed Computing, and Journal of Computer Science and Technology. Dr. Wu was general co-chair for IEEE MASS 2006, IEEE IPDPS 2008, IEEE ICDCS 2013, ACM MobiHoc 2014, ICPP 2016, and IEEE CNS 2016, as well as program co-chair for IEEE INFOCOM 2011 and CCF CNCC 2013. He was an IEEE Computer Society Distinguished Visitor, ACM Distinguished Speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a CCF Distinguished Speaker and a Fellow of the IEEE. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award.