Optimal Admission Control Mechanism Design for Time-Sensitive Services in Edge Computing

Shutong Chen[1] Lin Wang[2,3] Fangming Liu*[1]

[1]National Engineering Research Center for Big Data Technology and System,
Key Laboratory of Services Computing Technology and System, Ministry of Education,
School of Computer Science and Technology, Huazhong University of Science and Technology, China
[2]VU Amsterdam, The Netherlands [3]TU Darmstadt, Germany

***Abstract*—Edge computing is a promising solution for reducing service latency by provisioning time-sensitive services directly from the network edge. However, upon workload peaks at the resource-limited edge, an edge service has to queue service requests, incurring high waiting time. Such quality of service (QoS) degradation ruins the reputation and reduces the long-term revenue of the service provider.**

To address this issue, we propose an admission control mechanism for time-sensitive edge services. Specifically, we allow the service provider to offer admission advice to arriving requests regarding whether to join for service or balk to seek alternatives. Our goal is twofold: maximizing revenue of the service provider and ensuring QoS if the provided admission advice is followed. To this end, we propose a threshold structure that estimates the highest length of the request queue. Leveraging such a threshold structure, we propose O2A, a mechanism to balance the trade-off between increasing revenue from accepting more requests and guaranteeing QoS by advising requests to balk. Rigorous analysis shows that O2A achieves the goal and that the provided admission advice is optimal for end-users to follow. We further validate O2A through trace-driven simulations with both synthetic and real-world service request traces.

***Index Terms*—edge computing, admission control, mechanism design, queueing theory**

I. Introduction

With the rapid advancement of mobile and Internet of Things (IoT) technologies, the past decade has witnessed the proliferation of modern applications such as augmented reality and intelligent personal assistants [1], [2]. These applications are typically based on computation-intensive services such as object recognition and voice recognition with deep learning, and impose strict latency requirements for handling service requests [2]. Edge computing aims to bring computing resources to the edge of the network to support these time-sensitive services [3], [4], [5], [6]. Compared with centralized cloud computing, edge computing reduces the network latency and traffic through improved data locality, and has become an essential platform for hosting emerging time-sensitive services in the 5G era [7].

However, edge computing infrastructure is typically equipped with limited resources at each location due to their dispersed nature [3], which brings challenges for admission control for edge-based services [8]. During workload peaks where edge resources are highly utilized, an arriving service request will probably need to wait in a request queue until enough resources become available to serve it. Considering the latency requirement of edge services, the utility of serving a service request is thus conditioned by the state of the edge service, i.e., the waiting time in the queue. Without any admission control, the queue can keep growing, which results in high waiting time and consequently severe quality of service (QoS) degradation. Accordingly, this harms the reputation of the edge service provider and increases the chance of end-users switching to alternative service providers permanently, leading to revenue loss for the edge service provider in the long term [9]. On the other hand, a strict admission control mechanism which rejects service requests blindly without revealing any information to the end-user can also erode the trust of end-users in the edge service provider. Ideally, the edge service provider should provide useful admission advice, with which the end-user can make decisions on whether to stay or to balk, based on some shared system information.

Existing works have explored how to share system information to help end-users choose service providers with acceptable waiting time [10], [11], [12]. However, most of these solutions fall short with respect to one or both of the following aspects: (1) The information being shared in these mechanisms is too sensitive and is usually crucial to the system's security. For example, exposing the queue length of the edge computing system directly to the end-user may increase the vulnerability to potential denial-of-service attacks. (2) The service price is assumed to be state-dependent according to the dynamic waiting time. However, most existing commercial services such as Amazon Rekognition Image for image analysis [13] and Amazon Textract for text extraction [14] adopt fixed-price policies regardless of the system state.

In this work, we focus on admission control mechanisms that produce admission advice for time-sensitive edge services with fixed service prices on unobservable request queues. Generating optimal admission advice is challenging for the service provider. In particular, the service provider needs to balance carefully the trade-off between increasing revenue

*The corresponding author is Fangming Liu (fangminghk@gmail.com).

This work was funded in part by National Key Research & Development (R&D) Plan under grant 2017YFB1001703, by NSFC under grant 61722206 and 61761136014, and by National Program for Support of Top-notch Young Professionals in National Program for Special Support of Eminent Professionals. Lin Wang is supported partially by the German Research Foundation (DFG) Collaborative Research Center (CRC) 1053 – MAKI subproject B2.

from serving more requests and providing QoS guarantee by advising requests to balk. Admitting more service requests may improve the obtained revenue, but also lengthen the waiting queue and hence increase the probability of QoS degradation. Moreover, as the end-user is strategic, designing the admission control mechanism should consider the effect of admission advice on the end-user's strategy for deciding to join or balk. Besides, the admission advice should be straightforward—complex admission advice will limit the practicability of the admission control mechanism.

To overcome these challenges, we leverage the advantage of the strategic queueing approach [15] and propose an optimal admission control mechanism, named O2A, to provide advice to arriving requests regarding whether to join for service or balk to seek alternatives. The objective of the proposed mechanism is to maximize the revenue for the service provider while guaranteeing the expected QoS for service requests. In the proposed mechanism, we design a threshold structure that estimates the highest length of request queue considering the requests' arrival rate, QoS requirement, and strategy, as well as the service rate. To ensure security and practicality, the proposed mechanism provides a simple yet effective join-balk admission advice rather than show an anticipated delay [16] or expose the queue length. Note that rational service providers are committed to obeying the admission control mechanism. That is, the service provider will not provide deceptive or untrue admission advice for maximizing the likelihood that arriving requests join for obtaining edge service.

Further, we rigorously prove that for any fixed service price, the proposed admission control mechanism balances the tradeoff between increasing revenue and providing QoS guarantee optimally. And the admission advice is optimal for the end-user to follow when the system state is unobservable. Finally, we conduct extensive experiments to evaluate the performance of O2A. Specifically, we simulate the edge service system using both the synthetic trace and the real-world service request trace from Azure [17]. The results show that the proposed mechanism achieves maximal revenue regardless of the scarcity of edge resources. Moreover, the proposed mechanism provides QoS guarantee in expectation for accepted requests and efficiently ensures the QoS.

II. Problem Formulation

In this section, we provide models to characterize the system and present the problem formulation.

A. Model and Problem Formulation

System model. We consider an edge computing system that provides time-sensitive edge services such as object recognition with a fixed service price p. The homogeneous service request arrives following a Poisson process with expected rate λ. The required service time for each request is independent and identically distributed, and exponentially distributed with mean service rate μ. We assume a request queue to holding service requests before service, and requests in the queue are served in a first-come-first-served manner, following the general request management manner [9], [18]. The revenue of a request joining the queue is characterized by $R - c(n+1)$, where R is the value of obtaining the edge service, c is the per-unit waiting penalty, and $n \in \mathbb{Z}_0^+$ is the number of requests that already exist in the request queue (i.e., the system state) upon its arrival. If the arriving request joins the queue, its expected utility is given by $u(n,p) = R - c(n+1) - p$; otherwise, its utility is set to zero. The formulation of $u(n,p)$ is consistent with the discussion in Section I that the utility of obtaining the service is negatively correlated with the waiting time in the queue. And it is natural to assume that $u(0,p) \geqslant 0$. If the arriving request joins, the revenue of the service provider equals the received service price p.

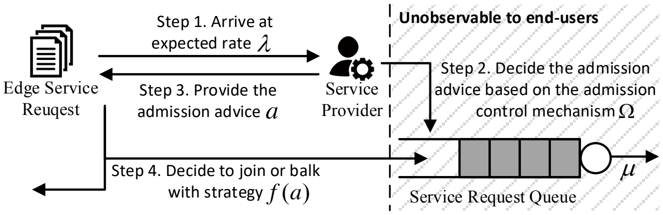

Step 1. Arrive at expected rate λ
Unobservable to end-users
Edge Service Reuqest
Step 3. Provide the admission advice a
Service Provider
Step 2. Decide the admission advice based on the admission control mechanism Ω
Step 4. Decide to join or balk with strategy $f(a)$
μ
Service Request Queue

Fig. 1. An overview of our admission control mechanism.

In this work, we assume the service price and the utility of requests are homogeneous. In practical, the value of obtaining the edge service can be further affected by the priority, the cost of data offloading, etc., and the service price is varied according to the allocated resources. We leave designing an admission control mechanism considering the heterogeneous services to future work.

Admission advice design. As in Fig. 1, when an edge service request is submitted by an end-user, the service provider will provide the admission advice a to the arriving request based on the current system state n. We assume end-users are strategic and Bayesian rational [15] where end-users can be motivated (or persuaded) to take the action following the admission advice [19]. To ensure the practicability of the admission control mechanism, we design a straightforward admission advice space $\mathcal{A} = \{0,1\}$. When the system state is n, the service provider suggests the arriving request joins (i.e., $a=1$) or balks (i.e., $a=0$) the queue with probability $\omega(n,a) \in [0,1]$ which satisfies $\sum_{a\in\mathcal{A}} \omega(n,a) = 1$. We define the admission control mechanism as $\Omega = (\mathcal{A}, \omega)$. The rational service provider is committed to obeying the admission control mechanism Ω, and Ω is announced to all end-users.

End-user equilibrium. After receiving the admission advice a from the service provider, the end-user will choose to join with probability $f(a) \in [0,1]$. A request cannot quit the request queue without processing if it joins the queue. We consider a symmetric equilibrium where all end-users follow the same strategy, which will be discussed in the Appendix of our technical report [20]. As end-users are Bayesian rational, they have a prior belief about the system state upon arrival of their requests [15]. Since the arrival pattern of requests follows a Poisson distribution, based on the PASTA property, the expected system state of end-users upon arrival is the steady state of the queue n_∞ which follows the steady state distribution of the queue π_∞ [21].

Leveraging the equilibrium characterization [15] discussed in the Appendix of our technical report [20], the end-user's optimal strategy is to follow the received advice strictly. That is, $f(a) = a$ for $a \in \mathcal{A} = \{0, 1\}$. Using this binary admission advice space and corresponding join-balk strategy, the problem of designing admission control mechanism Ω can be reduced to optimizing the probability of admission control advice ω. In the following, we utilize ω to represent the admission control mechanism. We consider that all end-users follow the optimal strategy $f(a) = a$, $\forall a \in \mathcal{A} = \{0, 1\}$, and ω and Ω are equivalent under this strategy.

Problem formulation. When the system reaches a steady state under the admission control mechanism ω and end-user equilibrium $f(a)$, the request joins and leaves the request queue at rate $\lambda \sum_{a\in\mathcal{A}} \omega(n_\infty, a) f(a)$ and μ, respectively. Here, $\sum_{a\in\mathcal{A}} \omega(n_\infty, a) f(a)$ can be regarded as the expected probability of joining the request queue in the steady state. As a result, the throughput of the edge service system in the steady state can be obtained as follows:

$$\mathbf{E}\Big[\lambda \sum_{a\in\mathcal{A}} \omega(n_\infty, a) f(a)\Big] = \mathbf{E}[\lambda\omega(n_\infty, 1)]$$
$$= \lambda \sum_{n=0}^{\infty} \pi_\infty(n)\omega(n, 1). \tag{1}$$

Under the admission control mechanism ω, when receiving admission advice a, the expected utility of the request can be obtained by $\mathbf{E}[u(n_\infty, p)]$. Since we have $f(a) = a$, $\forall a \in \mathcal{A} = \{0, 1\}$, the service provider should ensure the non-negative expected utility if it suggests the arriving request join the queue. Further, as the service price p is fixed, the revenue maximization problem is thus equivalent to the service throughput maximization problem, formulated as follows:

$$(P_1) \quad \max \quad \mathbf{E}[\lambda\omega(n_\infty, 1)] \tag{2}$$
$$\text{s.t.} \quad \mathbf{E}[u(n_\infty, p)|a = 1] \geqslant 0, \tag{3}$$
$$\mathbf{E}[u(n_\infty, p)|a = 0] \leqslant 0, \tag{4}$$
$$\omega(n, a) \in [0, 1], \;\; n \in \mathbb{Z}_0^+, \forall a \in \{0, 1\}. \tag{5}$$

For clarity, the important notations are listed in Table I. Here, the constraint (3) ensures that the service provider can provide the QoS guarantee if they advise the arriving request to join. The constraint (4) ensures that if the service provider cannot guarantee the QoS, they should recommend the arriving request to balk to seek alternatives. The constraints (3) and (4) together indicate that end-user's optimal strategy is to fully follow the received advice.

B. Problem Reformulation

We find that it is challenging to solve the problem (P_1) since the number of variables $\omega(n, a)$ is infinite and the constrains (3) and (4) are non-linear. To address these difficulties, in this section, we reformulate the problem (P_1) using the concept of steady system state [15], [22].

TABLE I
LIST OF NOTATIONS

Notations	Description
λ	the arrival rate of requests
μ	the mean service rate
R	the value of obtaining edge service
p	the service price
c	the per-unit waiting penalty
n	the number of waiting requests in the request queue
a	the admission advice
π_∞	the steady state distribution of the request queue
n_∞	the length of request queue independently follows π_∞
Ω	the admission control mechanism
$u(n, p)$	the expected utility of a request joining the queue with n waiting requests and service price p
$\omega(n, a)$	the probability of providing admission control advice a when the queue length is n
$f(a)$	the probability of joining the request queue if the arriving request receives advice a

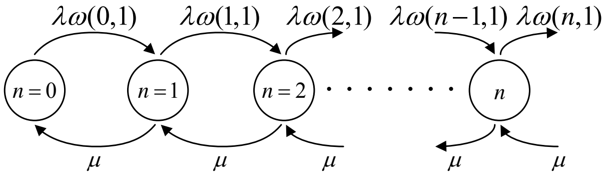


Fig. 2. An illustration of the system state transition.

As defined in Section II-A, the request arrives following a Poisson process with expected rate λ, and the service time is exponentially distributed with mean service rate μ. Thus, from the system state transition as depicted in Fig. 2, under the admission control mechanism ω, the steady state distribution π_∞^ω of the request queue satisfies

$$\pi_\infty^\omega(n + 1) = \frac{\lambda}{\mu}\omega(n, 1)\pi_\infty^\omega(n). \tag{6}$$

Since $\sum_{n=0}^\infty \pi_\infty^\omega(n) = 1$, we can find that

$$\pi_\infty^\omega(n) = \frac{\lambda^n}{\mu^n}\prod_{i=0}^{n-1}\omega(i, 1)\pi_\infty^\omega(0) \tag{7}$$

$$\Rightarrow \; \pi_\infty^\omega(0) = \Big[\sum_{n=0}^{\infty}\frac{\lambda^n}{\mu^n}\big(\prod_{i=0}^{n-1}\omega(i, 1)\big)\Big]^{-1}. \tag{8}$$

Let $\pi_n = \pi_\infty^\omega(n) \geqslant 0$, $\forall n \in \mathbb{Z}_0^+$. We have $\sum_{n=0}^\infty \pi_n = 1$, and

$$\frac{\lambda}{\mu}\pi_n - \pi_{n+1} = \frac{\lambda}{\mu}\pi_\infty^\omega(n) - \pi_\infty^\omega(n + 1)$$
$$\geqslant \frac{\lambda}{\mu}\pi_\infty^\omega(n)\omega(n, 1) - \pi_\infty^\omega(n + 1) = 0, \; \forall n \in \mathbb{Z}_0^+. \tag{9}$$

Based on the definition of request's expected utility and throughput, we can reformulate, respectively, the objective function (2), and the constraints (3) and (4) by the following inductions:

$$\begin{aligned}\mathbf{E}[\lambda\omega(n_\infty,1)] &= \sum_{n=0}^{\infty}\lambda\pi_\infty^\omega(n)\omega(n,1)\\ &= \mu\sum_{n=0}^{\infty}\pi_\infty^\omega(n+1) = \mu\sum_{n=1}^{\infty}\pi_n, \qquad (10)\end{aligned}$$

Since $\mathbf{E}[u(n_\infty,p)|a=1] = \sum_{n=0}^{\infty}\frac{\pi_\infty^\omega(n,a=1)}{P(a=1)}u(n,p)$ and $\mathbf{E}[u(n_\infty,p)|a=0] = \sum_{n=0}^{\infty}\frac{\pi_\infty^\omega(n,a=0)}{P(a=0)}u(n,p)$, where $P(a=1)$ and $P(a=0)$ represent the probability of admission advice $a=1$ and $a=0$, respectively, we have

$$\begin{aligned}&\sum_{n=0}^{\infty}\pi_\infty^\omega(n,a=1)u(n,p) = \sum_{n=0}^{\infty}\pi_\infty^\omega(n)\omega(n,1)u(n,p)\\ &= \frac{\mu}{\lambda}\sum_{n=0}^{\infty}\pi_\infty^\omega(n+1)u(n,p) = \frac{\mu}{\lambda}\sum_{n=1}^{\infty}\pi_n u(n-1,p) \geqslant 0, \qquad (11)\end{aligned}$$

$$\begin{aligned}&\sum_{n=0}^{\infty}\pi_\infty^\omega(n,a=0)u(n,p) = \sum_{n=0}^{\infty}\pi_\infty^\omega(n)(1-\omega(n,1))u(n,p)\\ &= \frac{\mu}{\lambda}\sum_{n=0}^{\infty}\big(\frac{\lambda}{\mu}\pi_n - \pi_{n+1}\big)u(n,p) \leqslant 0. \qquad (12)\end{aligned}$$

Combining all the above, the problem (P_2) can be reformulated as follows:

$$\begin{aligned}(P_2)\quad \max_{\pi}\quad & \mu\sum_{n=1}^{\infty}\pi_n && (13)\\ \text{s.t.}\quad & \sum_{n=1}^{\infty}\pi_n u(n-1,p) \geqslant 0, && (14)\\ & \sum_{n=0}^{\infty}(\frac{\lambda}{\mu}\pi_n - \pi_{n+1})u(n,p) \leqslant 0, && (15)\\ & \frac{\lambda}{\mu}\pi_n - \pi_{n+1} \geqslant 0,\ \forall n \in \mathbb{Z}_0^+, && (16)\\ & \sum_{n=0}^{\infty}\pi_n = 1, && (17)\\ & \pi_n \geqslant 0,\ \forall n \in \mathbb{Z}_0^+. && (18)\end{aligned}$$

where the objective function (13) and the constraints (14), (15), and (16) are derived from equations (10), (11), (12), and (9), respectively. We can find that the problem (P_2) largely reduces the model complexity by reformulating non-linear constraints (3) and (4) to linear constraints (14) and (15). Next we will discuss the relationship between the solutions of the problems (P_1) and (P_2) in Theorem 1, and present the admission control mechanism O2A by solving the problem (P_2) in the following Sec. III.

Theorem 1. *The solution of the problem (P_1) is feasible to the problem (P_2) and vice versa under certain definitions: For any feasible solution ω to the problem (P_1), the state distribution $\boldsymbol{\pi}$ defined as*

$$\boldsymbol{\pi} = \{\pi_n\},\ \pi_n = \pi_\infty^\omega(n),\ n \in \mathbb{Z}_0^+ \qquad (19)$$

is feasible to the problem (P_2) with the same objective value. On the contrary, for any feasible solution $\boldsymbol{\pi} = \{\pi_n\}$, $n \in \mathbb{Z}_0^+$ to the problem (P_2), the mechanism ω defined as

$$\omega(n,1) = \begin{cases}\frac{\mu\pi_{n+1}}{\lambda\pi_n}, & \textit{if } \pi_n > 0\\ 0, & \textit{if } \pi_n = 0\end{cases}, n \in \mathbb{Z}_0^+, \qquad (20)$$

is feasible to the problem (P_1) with the same objective value.

Proof. Following the above discussion, we can find that for any feasible solution ω to the problem (P_1), $\boldsymbol{\pi}$ defined by equation (19) is feasible to the problem (P_2) with the same objective value.

We next consider a feasible solution $\boldsymbol{\pi} = \{\pi_n\}$, $n \in \mathbb{Z}_0^+$ to the problem (P_2) and an admission control mechanism ω defined by equation (20). According to equations (20), (7), and (8), we have that for any $n \in \mathbb{Z}_0^+$,

$$\pi_\infty^\omega(n) = \frac{\lambda^n}{\mu^n}\prod_{i=0}^{n-1}\omega(i,1)\pi_\infty^\omega(0) = \pi_n.$$

As $\boldsymbol{\pi}$ is feasible to the problem (P_2), from equations (11) and (12), we can find that the constraints (3) and (4) hold. Also, from the definition of $\omega(n,1)$ in equation (20), it is easy to observe that the constraint (5) is satisfied. Lastly, from the equation (10), we can see that for any $\boldsymbol{\pi}$ feasible to the problem (P_2), ω defined by equation (20) is feasible to the problem (P_1) with the same objective value. □

III. Admission Control Mechanism Design

In this section, we leverage the advantage of the strategic queueing approach [15] and present the design of an admission control mechanism O2A based on a carefully designed threshold structure, with the goal of achieving the optimal revenue for the service provider and providing QoS guarantee for accepted requests. We first prove the existence of the optimal admission control mechanism for any fixed service price p and provide the definition of the threshold structure ω^x. Using the structure, we design the optimal admission control mechanism for the time-sensitive edge services with maximal revenue as defined in Section II.

Theorem 2. *For any fixed price p, there exists an optimal admission control mechanism with threshold structure ω^x, where $x \in \mathbb{R}_+ \cup \{\infty\}$, $x \geqslant N_p$, and N_p is the smallest integer satisfying $u(n,p) < 0$. The threshold structure ω^x is defined as: For $x \in \mathbb{R}_+$, we have*

$$\omega^x(n,1) = \begin{cases}1, & \textit{if } n < \lfloor x \rfloor\\ x - \lfloor x \rfloor, & \textit{if } n = \lfloor x \rfloor\\ 0, & \textit{otherwise}\end{cases}. \qquad (21)$$

For $x = \infty$, we have $\omega^\infty(n,1) = 1$, $\forall n \geqslant 0$.

We prove the theorem by analyzing the property of the optimal solution to the problem (P_2), the existence of the threshold structure ω^x, and finally the existence of the optimal solution with the structure ω^x. More details for the proof are provided in the Appendix of our technical report [20].

Algorithm 1 Optimal Admission Control Mechanism O2A

```
Initialize: the arriving rate $\lambda$ of requests, the service rate $\mu$, the expected utility of requests $u(n,p)=R-c(n+1)-p$, and the threshold structure $\omega^{N+q}$ with parameters $N$ and $q$ which are respectively given by
    $$N=\begin{cases}\infty, & \text{if } \frac{\lambda}{\mu}\leqslant 1-\frac{c}{R-p}\\ \lfloor\frac{(W_0(-ye^{-y})+y)}{\log(\frac{\lambda}{\mu})}\rfloor, & \text{if } \frac{\lambda}{\mu}\in(1-\frac{c}{R-p},1)\\ \lfloor 2(R-p)/c-1\rfloor, & \text{if } \frac{\lambda}{\mu}=1\\ \lfloor\frac{(W_{-1}(-ye^{-y})+y)}{\log(\frac{\lambda}{\mu})}\rfloor, & \text{if } \frac{\lambda}{\mu}>1\end{cases}$$
    and $q=0$ if $\frac{(W_0(-ye^{-y})+y)}{\log(\frac{\lambda}{\mu})}\in\mathbb{Z}^+$ for $\frac{\lambda}{\mu}\in(1-\frac{c}{R-p},1)$ or $\frac{(W_{-1}(-ye^{-y})+y)}{\log(\frac{\lambda}{\mu})}\in\mathbb{Z}^+$ for $\frac{\lambda}{\mu}>1$, otherwise
    $$q=\begin{cases}\frac{\sum_{n=0}^{N-1}(\frac{\lambda}{\mu})^n(R-c(n+1)-p)}{(\frac{\lambda}{\mu})^N(c(N+1)+p-R)}, & \\ \qquad \text{if } \frac{\lambda}{\mu}\in(1-\frac{c}{R-p},1)\cup(1,+\infty) & ,\\ \frac{cN(N-1)/2-N(R-c-p)}{R-c(N+1)-p}, \text{ if } \frac{\lambda}{\mu}=1 & \end{cases}$$
    where $W_i,\ i\in\{0,-1\}$ is the Lambert-W function, $y=(1-\rho)\log(1/\phi)\leqslant 1$, $\rho=\frac{R-p-c}{c}\cdot\frac{1-\frac{\lambda}{\mu}}{\frac{\lambda}{\mu}}$, and $\phi=(\frac{\lambda}{\mu})^{\frac{\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}}}$.
while the service is running do
    if the service provider receives edge service request from end-users then
        if $n<N$ then
            $a=1$;
        else if $n=N$ then
            $a=1$ with probability $q$, otherwise, $a=0$;
        else
            $a=0$;
        end if
        The service provider provides the admission advice $a$ to the arriving request;
    end if
end while
```

Recall that the edge service provider processes requests at an average rate of μ with a fixed service price p, and the request arrives at an expected rate of λ with utility function $u(n,p)=R-c(n+1)-p$.

Based on Theorem 2 and letting $x=N+q\geqslant N_p$, where $N\in\mathbb{Z}^+$ and $q\in[0,1]$, we have

$$\omega^x(n,1)=\begin{cases}1, & \text{if } n<N\\ q, & \text{if } n=N\\ 0, & \text{if } n>N\end{cases}. \tag{22}$$

From equation (6), the conditional steady state distribution of the system under admission advice $a=1$ can be given by

$$\pi^x_\infty(n|a=1)=\frac{(\frac{\lambda}{\mu})^n\mathbf{I}\{n<N\}+q(\frac{\lambda}{\mu})^N\mathbf{I}\{n=N\}}{\sum_{i=0}^{N-1}(\frac{\lambda}{\mu})^i+q(\frac{\lambda}{\mu})^N}, \tag{23}$$

where $\mathbf{I}$ is an indicator function where, for example, $\mathbf{I}\{n=N\}=1$ if $n=N$, and $\mathbf{I}\{n=N\}=0$ if $n\neq N$. Consequently, we can obtain the expected utility of the request given $a=1$ under threshold structure ω^x by

$$\mathbf{E}^{\omega^x}[u(n_\infty,p)|a=1]=\frac{\sum_{n=0}^{N-1}(\frac{\lambda}{\mu})^n u(n,p)+q(\frac{\lambda}{\mu})^N u(N,p)}{\sum_{n=0}^{N-1}(\frac{\lambda}{\mu})^n+q(\frac{\lambda}{\mu})^N}.$$

To ensure that it is optimal for the arriving request to join the queue when the admission advice $a=1$, the conditional expected utility $\mathbf{E}^{\omega^x}[u(n_\infty,p)|a=1]$ should be no less than zero. That is, we need to ensure

$$\sum_{n=0}^{N-1}(\frac{\lambda}{\mu})^n u(n,p)+q(\frac{\lambda}{\mu})^N u(N,p)\geqslant 0. \tag{24}$$

Note that from equations (8) and (10), we can obtain the throughput by

$$\begin{aligned}\mathbf{E}^{\omega^x}[\lambda\omega(n_\infty,1)]&=\mu\sum_{n=1}^{\infty}\pi^\omega_\infty(n)=\mu(1-\pi^\omega_\infty(0))\\ &=\mu\big[1-\big(\sum_{n=0}^{N-1}1/(\frac{\lambda}{\mu})^n+q(\frac{\lambda}{\mu})^N\big)\big].\end{aligned}$$

It is easy to verify that $\mathbf{E}^{\omega^x}[\lambda\omega(n_\infty,1)]$ is increasing in $x=N+q$, and the optimal revenue can be obtained if maximizing $x=N+q$. Thus, in the following, we will calculate the maximal N and q for ensuring equation (24) in the case of $\frac{\lambda}{\mu}=1$ and in the case of $\frac{\lambda}{\mu}\neq 1$, respectively.

Case 1: If $\frac{\lambda}{\mu}=1$, we have

$$\sum_{n=0}^{N-1}(R-c(n+1)-p)+q(R-c(N+1)-p)\geqslant 0 \tag{25}$$

$$\Rightarrow N\leqslant\frac{2}{c}\big(R-p+q(R-c(N+1)-p)/N\big)-1.$$

Hence, if we consider the case of $q=0$, it is easy to obtain the greatest $N^*=\lfloor 2(R-p)/c-1\rfloor$ that satisfies equation (24), where $\lfloor x\rfloor$ indicates the greatest integer no higher than x. If we further let equation (25) be an equality, we can obtain the maximum value of q by

$$q^*=\frac{cN^*(N^*-1)/2-N^*(R-c-p)}{R-c(N^*+1)-p}.$$

Case 2: If $\frac{\lambda}{\mu}\neq 1$, applying the same idea as above, we first let $q=0$ and calculate the largest N^* that satisfies equation (24) as follows:

$$\begin{aligned}&c\cdot\Big(\frac{1-(\frac{\lambda}{\mu})^N}{1-\frac{\lambda}{\mu}}+\frac{\frac{\lambda}{\mu}-N(\frac{\lambda}{\mu})^N+(N-1)(\frac{\lambda}{\mu})^{N+1}}{(1-\frac{\lambda}{\mu})^2}\Big)\\ &\quad-(R-p)\frac{1-(\frac{\lambda}{\mu})^N}{1-(\frac{\lambda}{\mu})}\leqslant 0\\ &\Rightarrow\ \frac{R-c-p}{c}\cdot\frac{(1-\frac{\lambda}{\mu})(1-(\frac{\lambda}{\mu})^N)}{\frac{\lambda}{\mu}}\end{aligned}$$

$$\geqslant 1 - N(\frac{\lambda}{\mu})^{N-1} + (N-1)(\frac{\lambda}{\mu})^{N}. \tag{26}$$

Let $\rho = \frac{R-p-c}{c} \cdot \frac{1-\frac{\lambda}{\mu}}{\frac{\lambda}{\mu}}$ and $\xi = \frac{1-\frac{\lambda}{\mu}}{\frac{\lambda}{\mu}}$, we can reformulate equation (26) as follows:

$$\rho(1-(\frac{\lambda}{\mu})^N) \geqslant 1-(\frac{\lambda}{\mu})^N - N\xi(\frac{\lambda}{\mu})^N$$
$$\Rightarrow \ (1-\rho+N\xi)(\frac{\lambda}{\mu})^N \geqslant 1-\rho. \tag{27}$$

If $\rho \geqslant 1$, i.e., $\frac{\lambda}{\mu} \leqslant 1 - c/(R-p)$, we have $1-\rho \leqslant 0$, and $1-\rho+N\xi = 1-\frac{1}{c}\cdot\frac{1-\frac{\lambda}{\mu}}{\frac{\lambda}{\mu}}u(N,p) \geqslant 1$ for $\forall N \geqslant N_p$. That is, equation (27) is satisfied for $\forall N \geqslant N_p$, and hence $N^* = \infty$.

If $\rho < 1$, multiplying both sides of equation (27) by $\big((\frac{\lambda}{\mu})^{1/\xi}\big)^{1-\rho}$, and letting $\psi = 1-\rho+N\xi$ and $\phi = (\frac{\lambda}{\mu})^{1/\xi}$, we have $\psi\phi^{\psi} \geqslant (1-\rho)\phi^{1-\rho}$. And then multiplying both sides by $\log(1/\phi)$, we have

$$\psi\log(1/\phi)\exp\big(-\psi\log(1/\phi)\big)$$
$$\geqslant (1-\rho)\log(1/\phi)\exp\big(-(1-\rho)\log(1/\phi)\big). \tag{28}$$

Then we further define the function $H(y)$ for $y > 0$ satisfying $H(y)\exp\big(-H(y)\big) = y\exp(-y)$ with $H(y) \neq y$ for $y \neq 1$. And we have $H(y) < 1$ for $y > 1$, $H(y) > 1$ for $0 < y < 1$, and $H(1) = 1$.

For $\frac{\lambda}{\mu} > 1$, it is easy to find that $1-\rho \geqslant 1$, $\xi < 0$, and $\log(1/\phi) = \frac{\frac{\lambda}{\mu}}{\frac{\lambda}{\mu}-1}\log(\frac{\lambda}{\mu}) \geqslant \frac{\frac{\lambda}{\mu}}{\frac{\lambda}{\mu}-1}\cdot\frac{\frac{\lambda}{\mu}-1}{\frac{\lambda}{\mu}} = 1$. Letting $y = (1-\rho)\log(1/\phi)$, we have $y \geqslant 1$. Then, we can re-write equation (28) as $H(y) \leqslant \psi\log(1/\phi) \leqslant y$.

For $\frac{\lambda}{\mu} < 1$, we have $0 \leqslant 1-\rho \leqslant 1$, $\xi > 0$, $\log(1/\phi) \leqslant 1$, and $y = (1-\rho)\log(1/\phi) \leqslant 1$. Then, we can re-write equation (28) as $y \leqslant \psi\log(1/\phi) \leqslant H(y)$.

Since $\psi = 1-\rho+N\xi$, we have

$$N \leqslant \frac{H(y)-y}{\xi\log(1/\phi)} \ \Rightarrow \ N^* = \left\lfloor\frac{H(y)-y}{\xi\log(1/\phi)}\right\rfloor.$$

Using Lambert-W function [23], [15], we have $H(y) = -W_i(-ye^{-y})$, where $i = 0$ if $y > 1$, and $i = -1$ if $y < 1$. Combining all the above, we have

$$N^* = \big\lfloor (W_i(-ye^{-y})+y)/\log(\frac{\lambda}{\mu})\big\rfloor,$$

where $i = 0$ if $\frac{\lambda}{\mu} \in \big(1-c/(R-p), 1\big)$, and $i = -1$ if $\frac{\lambda}{\mu} > 1$. If $(W_i(-ye^{-y})+y)/\log(\frac{\lambda}{\mu}) \notin \mathbb{Z}^+$, the maximum q^* is given by

$$q^* = \frac{\sum_{n=0}^{N^*-1}(\frac{\lambda}{\mu})^n(R-c(n+1)-p)}{(\frac{\lambda}{\mu})^{N^*}(c(N^*+1)+p-R)},$$

otherwise, $q^* = 0$.

Finally, following the above analysis, we design an optimal admission control mechanism O2A as shown in Algorithm 1.

IV. Evaluation

A. Evaluation Setup

Based on the assumption in Section II, we built a trace-driven simulator to simulate an edge service system that provides time-sensitive services to end-users. We assume the value of obtaining edge service $R = 10$, the per-unit waiting penalty $c = 2$, and the service price $p = 3$, which represents a modest sensitivity of the service to the waiting time. To investigate how does the ratio of arrival rate λ to service rate μ affect the performance of O2A, we vary $\frac{\lambda}{\mu}$ to obtain a spectrum of results, instead of focusing on the concrete values for λ and μ. The number of total requests is more than 5,000, which is sufficient to evaluate the average performance of the proposed mechanism.

We compare our proposed mechanism O2A with the following approaches: (1) Fully revealing, for which the service provider fully reveals the length of request queue, n, to the end-user, and the end-user chooses to join if and only if $u(n,p) = R-c(n+1)-p \geqslant 0$. (2) No revealing, for which the service provider does not provide any information about the system state except the arrival rate λ and service rate μ, and the arriving request joins the queue with probability q. In this case, the steady system state $n = \frac{q\lambda}{\mu-q\lambda}$ [22], and q can be calculated by $R-c(\frac{q\lambda}{\mu-q\lambda}+1)-p \geqslant 0 \Rightarrow q = \min\{1, \frac{\lambda}{\mu}(1-c/(R-p))\}$. (3) Mixed strategy, for which the request joins the request queue with probability 0.9 if the admission advice $a = 1$; otherwise, it joins with probability 0.1. (4) RL approach, for which the service provider makes admission control advice using policy-based method. The policy is trained by A3C [24], a state-of-the-art reinforcement learning (RL) algorithm. In this approach, the state input is the length of request queue n and the action is the admission advice a. The reward of accepting one request is the weighted sum of the revenue of the service provider and the utility of accepted requests[1], i.e., $R + wu(n,p)$, where the weight $w = 0.2$ in the evaluation. And there is one policy assigned to each possible $\frac{\lambda}{\mu}$.

B. Evaluation Results

Fig. 3 compares the normalized long-term revenue of the service provider obtained by O2A and four baseline approaches, as $\frac{\lambda}{\mu}$ increases. The normalized long-term revenue is the ratio of the total revenue obtained by the mechanism to the value by accepting all requests. The results show that O2A achieves up to 47% long-term revenue increase compared with all baseline approaches. Compared with the fully-revealing approach, we find that O2A always achieves the best performance especially when the edge resource is sufficient. With the fully-revealing mechanism, the end-user decides to join the queue if and only if the system state totally satisfies $u(n,p) = R-c(n+1)-p \geqslant 0$. This conservative strategy reduces the resource utilization when

[1] In general, it is difficult for RL approach to meet the hard constraint of non-negative utility, and we use the weighted-sum reward for simplicity. We refer the interested reader to existing approaches using dedicated design for handling the hard constraint [18], [25].

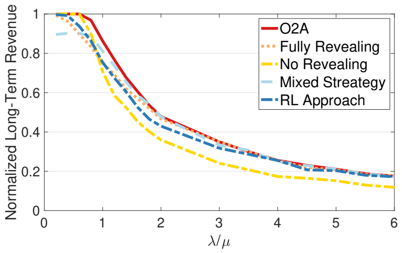


Fig. 3. Comparison of the normalized long-term revenue of the service provider with different approaches as $\frac{\lambda}{\mu}$ increases.

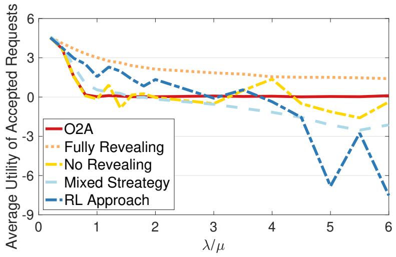


Fig. 4. Comparison of the average utility of accepted requests with different approaches as $\frac{\lambda}{\mu}$ increases.

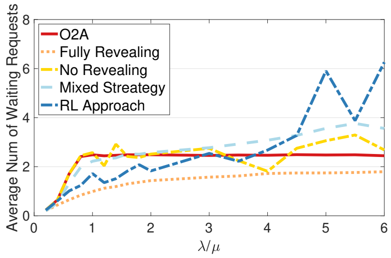


Fig. 5. Comparison of the average number of waiting requests with different approaches as $\frac{\lambda}{\mu}$ increases.

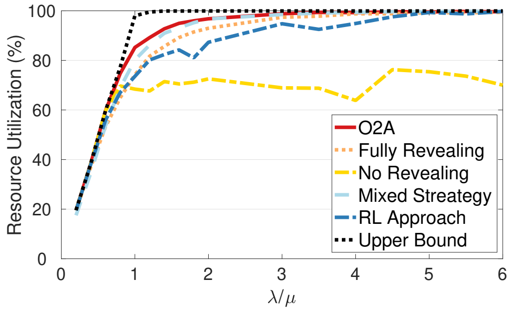


Fig. 6. Comparison of the resource utilization of the edge server with different approaches as $\frac{\lambda}{\mu}$ increases.

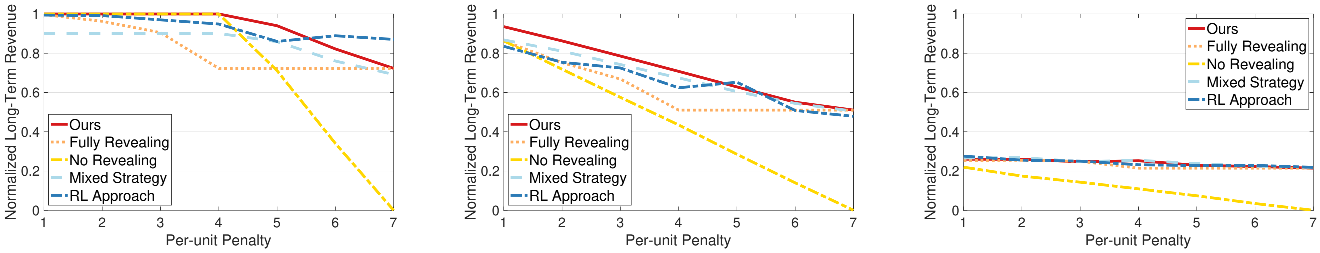
the edge capacity is sufficient and hence hurts the long-term revenue of the service provider when $\frac{\lambda}{\mu} < 2$. With the no-revealing mechanism, according to the discussion in Section I, the lack of information about the system state increases the chance of end-users switching to alternatives. Hence, it decreases the revenue of the service provider, especially when the arrival rate exceeds the service rate. Although the mixed strategy achieves the same or even a slightly higher level of the revenue compared with O2A when $\frac{\lambda}{\mu}$ is high, in the following results, we will show that this strategy is not beneficial for the end-user. Finally, we find that RL approach also obtains poor long-term revenue when $\frac{\lambda}{\mu} \leqslant 3$. The reason is that it is hard for the RL approach to ensure the QoS guarantee for accepted requests, and it has to be conservative when providing the admission advice.

Fig. 4 and Fig. 5 compare the average utility of accepted requests and the average number of waiting requests in the request queue with O2A and four baseline approaches, respectively. We can find that O2A ensures the non-negative utility for requests even when the edge resource is highly utilized, which is in line with the discussion in Section II. When the request arrival rate exceeds the service rate, i.e., $\frac{\lambda}{\mu} > 1$, the service provider should decide the threshold of the admission control mechanism more carefully. The results show that the proposed mechanism keeps the balance of the system state efficiently and ensures the maximum revenue with performance guarantee.

Since the fully-revealing mechanism is conservative, end-users will choose to join only if the length of request queue is small, which increases their average utility and lowers the number of waiting requests compared with O2A. However, when the service provider does not provide any admission advice, it is hard for end-users to obtain edge services with performance guarantee during workload peak periods. When end-users do not follow the admission advice entirely, they are more vulnerable to the utility degradation during workload peak periods. The results also support the argument in Section II that the admission advice provided by the mechanism is the best choice for end-users to follow when the system state is unobservable. As discussed above, RL approach cannot make sure of the QoS guarantee for accepted requests. Hence, RL approach fails to restrict the queue length when the resource capacity is strictly limited, and the utility of the accepted requests drops greatly. It is noting that the utility can be improved by increasing the weight w, which however, would hurt the long-term revenue of the service provider in turn.

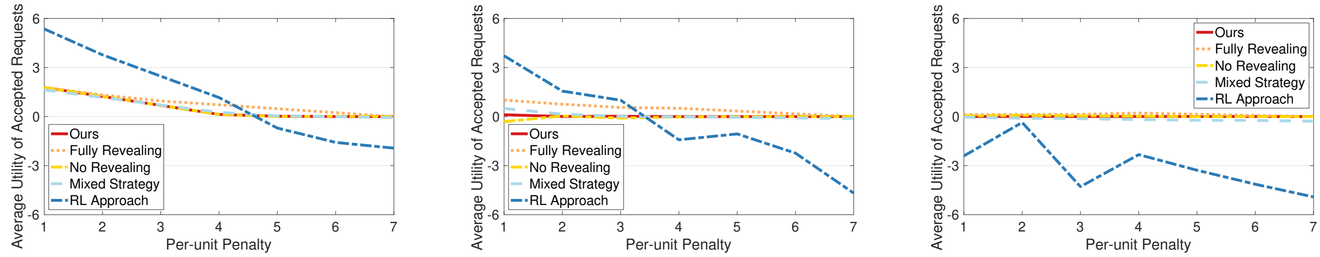
Fig. 6 shows the resource utilization of edge server achieved by different approaches. The black dotted line represents the maximum resource utilization under different $\frac{\lambda}{\mu}$. The results show that O2A can make use of edge resources adequately especially when the resource capacity is limited. Since the fully-revealing approach and RL approach are both conservative, sometimes resources are wasted when the edge capacity is sufficient. For the no-revealing approach, since the arriving request makes decision only according to the probability q provided in Sec. IV-A, the resource utilization stays in a low and stable level even the the resource is extremely limited.

Fig. 7 shows the normalized long-term revenue as the per-unit penalty c increases when $\frac{\lambda}{\mu} = 0.4$, 1, and 5, respectively. And Fig. 8 shows the corresponding average utility. We can find that compared with other approaches, O2A always achieves the best revenue with performance guarantee when the per-unit penalty varies. Specifically, when the service rate exceeds the arrival rate significantly, i.e., $\frac{\lambda}{\mu} = 0.4$, the probability that service requests wait in the queue is low. So when the request is less sensitive to the waiting delay, the service provider using O2A can serve all arriving requests and achieve the maximum revenue. When the resource capacity decreases, O2A has to balance the trade-off between increasing revenue from serving more requests and providing QoS guarantee, which limits the revenue compared with the case of $\frac{\lambda}{\mu} = 0.4$. It



(a) $\frac{\lambda}{\mu} = 0.4$ (b) $\frac{\lambda}{\mu} = 1$ (c) $\frac{\lambda}{\mu} = 5$

Fig. 7. Comparison of the normalized long-term revenue of the service provider as the per-unit penalty c increases with different $\frac{\lambda}{\mu}$.



(a) $\frac{\lambda}{\mu} = 0.4$ (b) $\frac{\lambda}{\mu} = 1$ (c) $\frac{\lambda}{\mu} = 5$

Fig. 8. Comparison of the average utility of accepted requests as the per-unit penalty c increases with different $\frac{\lambda}{\mu}$.

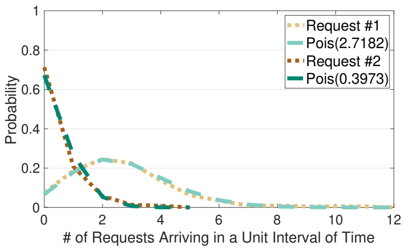


Fig. 9. The distributions of two selected service requests and the estimated Poisson distributions.

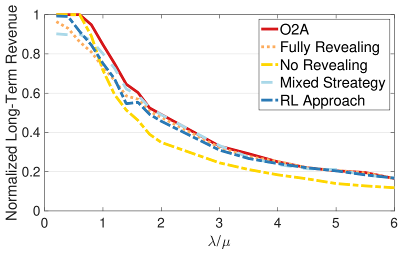


Fig. 10. Comparison of the normalized long-term revenue of the service provider with different approaches as $\frac{\lambda}{\mu}$ increases under Request #1.

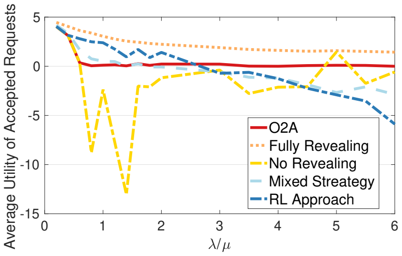


Fig. 11. Comparison of the average utility of accepted requests with different approaches as $\frac{\lambda}{\mu}$ increases under Request #1.

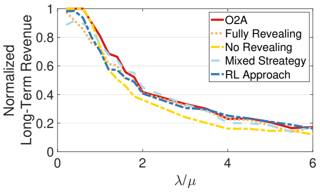


Fig. 12. Comparison of the normalized long-term revenue of the service provider with different approaches as $\frac{\lambda}{\mu}$ increases under Request #2.

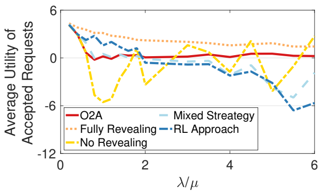


Fig. 13. Comparison of the average utility of accepted requests with different approaches as $\frac{\lambda}{\mu}$ increases under Request #2.

is also hard to increase the revenue gain by carefully providing admission control advice. However, even when the resource capacity is significantly restricted, O2A still achieves the best revenue while providing non-negative request utility. Note that although RL approach sometimes achieves better long-term revenue than O2A, as discussed before, it cannot ensure the non-negative utility for end-users. That is, RL approach fails to balance the trade-off between achieving higher revenue and providing QoS guarantee.

C. Evaluation Using Azure Request Trace

We further investigate the performance of O2A on the real-world request trace from Azure [17]. The trace records more than 5M virtual machine requests with the request type, start time, etc. The start time in this trace resembles the arrival time of edge service requests. We select two types of requests from the dataset and estimate the value of parameters using the maximum likelihood estimator [22]. Specifically, Request #1 and Request #2 approximately follow Poisson distribution with arrival rate $\lambda = 2.7182$ and $\lambda = 0.3973$, respectively. Fig. 9 provides the probability distribution of the requests and the corresponding estimated Poisson distribution. To eliminate the effect of the perturbation on the mechanism performance, we filter out outliers from the trace, e.g., the case when the number of arriving requests exceeds 12 per unit interval of time for Request #1, which only accounts for around 1% of total cases.

Fig. 10 and Fig. 11 show the long-term revenue and average utility of accepted requests, respectively, under Request #1. We find that O2A always achieves the best long-term revenue with performance guarantee, no matter what $\frac{\lambda}{\mu}$ is. Specifically, the maximal improvement of revenue is up to 49% compared with the baselines, which is consistent with the results in Section IV-B. O2A also provides the QoS guarantee for accepted requests although the parameter estimation is not entirely perfect.

Fig. 12 and Fig. 13 compare the performance of O2A and the baselines under Request #2. Because of the low arrival rate, the number of request samples is limited, which restricts the performance of its parameter estimation and affects the efficiency of the proposed mechanism. However, O2A still achieves the best long-term revenue and ensures the QoS for requests generally. It is noting that the long-term revenue of the RL approach is slightly higher than O2A, which is at the expense of end-users' utility. This result is also consistent with the observations in Section IV-B. We also find that under the real-world request trace, the no-revealing mechanism obtains the worse and more fluctuating average utility compared with the results in Fig. 4. This result further indicates the necessity of providing admission advice.

V. Related Work

Admission control has been widely used in the networking [26], [27] and the cloud and edge computing service management [28], [29], [30], [31], [32]. Ojijo et al. [33] survey the admission control for network slicing in the emerging 5G. Wu et al. [34] study the joint admission control for the communication traffic and geographical computation load balancing problem. The authors propose a Lyapunov-based algorithm ensuring the upper bound of the total system cost. Ferrer et al. [35] investigate the effect of unpredictability of edge resources on the admission control mechanism design. The authors design a two-step admission control mechanism which filters available hosts, and then ranks and selects the candidates according to the expected potential QoS achieved by the host and the battery level of the host. Guo et al. [36] focus on the admission control of multi-class end-users for the multi-level edge system, and make the admission decision for each class of end-users by an approximate algorithm without performance guarantee. Baranwal et al. [37] focus on the case that the strategic end-user may dynamically change their strategy. They design a game-based admission control to maximize the revenue of the service provider and the resource utilization. Raeis et al. [18] consider the multi-server queueing system for time-sensitive requests. They design a RL-based admission control algorithm to ensure the performance guarantee for accepted requests. However, as discussed in Sec. IV, the RL-based mechanism should be trained for each possible arrival rate, service rate, and the topology of queues, which would make the overhead bloated in practice. Dimitrakopoulos et al. [38] study the effect of service rate flexibility on the queueing system. In this dynamic scenario, the service rate can be adjusted according to the queue length, while higher service rate brings more operational costs. To balance the received service price and the operational cost, the authors propose a threshold structure for service and admission control. Jain et al. [39] survey the state dependent queueing under the admission control with the F-policy-based threshold structure. All these studies do not reveal any information about the system state to the end-user. Thus, the end-user in these mechanisms cannot make informed decisions regarding whether to join or balk.

Besides governing the admission directly, there are also some studies focusing on pricing mechanisms or signaling mechanisms for performing admission control indirectly [40], [41], [42]. Here we summarize queueing theory-related existing work only. Chen et al. [10] propose a state-dependent pricing mechanism for the observable queue to optimize the revenue and also the social welfare. Using the dynamic pricing, the service provider can incentivize the arriving end-user with low service valuation to balk when edge resources are limited. Borgs et al. [12] propose a closed-form threshold structure using the Lambert-W function. Yildirim et al. [11] focus on the state-dependent pricing mechanism for the end-user batches. Liu et al. [43] investigate the relationship between the admission fees for heterogeneous edge services and the revenue and benefit maximization. In these mechanisms, the system state is directly exposed to the arriving end-user, which can result in security issues and is impractical to the edge computing scenario.

Dughmi [44] studies the literature on the signaling mechanism where the service provider strategically shares information to persuade end-users to take desired actions. Lingenbrink et al. [15] propose a signaling and pricing mechanism for homogeneous end-users in the unobservable queueing system. Their goal is to maximize the profile of the service provider and ensure the performance guarantee for end-users. We extend the mechanism to a more general case where we further consider the effect of service rate on the system state, the threshold structure, and the achieved throughput, which makes the admission control mechanism more functional in practice. Altman et al. [45] propose a threshold-based mechanism that the service operator shows the arriving requests whether the system state exceeds a threshold value or not. The objective of this mechanism is to minimize the expected queue length and maximize the service throughput, which is different with O2A's goal. And the mechanism works only when $\frac{\lambda}{\mu} < 1$.

VI. Conclusion

We study the admission control problem for time-sensitive edge services when the system state is unobservable for end-users. We design an optimal admission mechanism O2A to recommend arriving service requests join or balk the request queue according to an efficient threshold structure. O2A is proved to achieve the optimal revenue for the service provider while providing QoS guarantee for the served requests. Our trace-driven evaluations further confirm superior performance of O2A compared with existing approaches.

References

[1] L. Liu, H. Li, and M. Gruteser, "Edge Assisted Real-time Object Detection for Mobile Augmented Reality," in *ACM MobiCom*, 2019, pp. 1–16.

[2] D. Crankshaw, G.-E. Sela, X. Mo, C. Zumar, I. Stoica, J. Gonzalez, and A. Tumanov, "InferLine: Latency-aware provisioning and scaling for prediction serving pipelines," in *ACM SoCC*, 2020, pp. 477–491.

[3] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.

[4] S. Chen, L. Jiao, F. Liu, and L. Wang, "Edgedr: An online mechanism design for demand response in edge clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 2, pp. 343–358, 2022.

[5] Q. Chen, Z. Zheng, C. Hu, D. Wang, and F. Liu, "On-edge multi-task transfer learning: Model and practice with data-driven task allocation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 6, pp. 1357–1371, 2020.

[6] M. Li, Q. Zhang, and F. Liu, "Finedge: A dynamic cost-efficient edge resource management platform for NFV network," in *IEEE/ACM IWQoS*. IEEE, 2020, pp. 1–10.

[7] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—a key technology towards 5G," *ETSI White Paper*, vol. 11, no. 11, pp. 1–16, 2015.

[8] S. Li, N. Zhang, S. Lin, L. Kong, A. K. Katangur, M. K. Khan, M. Ni, and G. Zhu, "Joint admission control and resource allocation in edge computing for internet of things," *IEEE Transactions on Networking*, vol. 32, no. 1, pp. 72–79, 2018.

[9] B. Han, V. Sciancalepore, D. Feng, X. Costa-Pérez, and H. D. Schotten, "A utility-driven multi-queue admission control solution for network slicing," in *IEEE INFOCOM*, 2019, pp. 55–63.

[10] H. Chen and M. Z. Frank, "State dependent pricing with a queue," *Iie Transactions*, vol. 33, no. 10, pp. 847–860, 2001.

[11] U. Yildirim and J. J. Hasenbein, "Admission control and pricing in a queue with batch arrivals," *Operations Research Letters*, vol. 38, no. 5, pp. 427–431, 2010.

[12] C. Borgs, J. T. Chayes, S. Doroudi, M. Harchol-Balter, and K. Xu, "The optimal admission threshold in observable queues with state dependent pricing," *Probability in the Engineering and Informational Sciences*, vol. 28, no. 1, pp. 101–119, 2014.

[13] Amazon Rekognition pricing. [Online]. Available: https://aws.amazon.com/rekognition/pricing/?nc=sn&loc=4#Amazon_Rekognition_Image_pricing

[14] Amazon Textract pricing. [Online]. Available: https://aws.amazon.com/textract/pricing

[15] D. Lingenbrink and K. Iyer, "Optimal signaling mechanisms in unobservable queues," *INFORMS Operations Research*, vol. 67, no. 5, pp. 1397–1416, 2019.

[16] M. Armony and C. Maglaras, "Contact centers with a call-back option and real-time delay information," *INFORMS Operations Research*, vol. 52, no. 4, pp. 527–545, 2004.

[17] O. Hadary, L. Marshall, I. Menache, A. Pan, E. E. Greeff, D. Dion, S. Dorminey, S. Joshi, Y. Chen, M. Russinovich, and T. Moscibroda, "Protean: VM allocation service at scale," in *USENIX OSDI*, 2020, pp. 845–861.

[18] M. Raeis, A. Tizghadam, and A. Leon-Garcia, "Reinforcement learning-based admission control in delay-sensitive service systems," in *IEEE GLOBECOM*, 2020, pp. 1–6.

[19] E. Kamenica and M. Gentzkow, "Bayesian persuasion," *American Economic Review*, vol. 101, no. 6, pp. 2590–2615, 2011.

[20] S. Chen, L. Wang, and F. Liu, "Optimal admission control mechanism design for time-sensitive services in edge computing," *Technical report*. [Online]. Available: https://fangmingliu.github.io/files/INFOCOM22-O2A-TechnicalReport.pdf

[21] R. W. Wolff, "Poisson arrivals see time averages," *INFORMS Operations Research*, vol. 30, no. 2, pp. 223–231, 1982.

[22] J. F. Shortle, J. M. Thompson, D. Gross, and C. M. Harris, *Fundamentals of queueing theory*. John Wiley & Sons, 2018, vol. 399.

[23] R. M. Corless, G. H. Gonnet, D. E. Hare, D. J. Jeffrey, and D. E. Knuth, "On the lambertw function," *Advances in Computational mathematics*, vol. 5, no. 1, pp. 329–359, 1996.

[24] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *ICML*. PMLR, 2016, pp. 1928–1937.

[25] Y. Ran, H. Hu, X. Zhou, and Y. Wen, "Deepee: Joint optimization of job scheduling and cooling control for data center energy efficiency using deep reinforcement learning," in *IEEE ICDCS*, 2019, pp. 645–655.

[26] J. Babiarz and M. Menth, "A survey of pcn-based admission control and flow termination," *IEEE COMMUNICATIONS SURVEYS & TUTORIALS*, vol. 12, no. 3, 2010.

[27] L. Khoukhi, H. Badis, L. Merghem-Boulahia, and M. Esseghir, "Admission control in wireless ad hoc networks: a survey," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, pp. 1–13, 2013.

[28] D. T. Hoang, D. Niyato, and P. Wang, "Optimal admission control policy for mobile cloud computing hotspot with cloudlet," in *IEEE WCNC*, 2012, pp. 3145–3149.

[29] F. Liu, Z. Zhou, H. Jin, B. Li, B. Li, and H. Jiang, "On arbitrating the power-performance tradeoff in saas clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 10, pp. 2648–2658, 2014.

[30] B. Gao, Z. Zhou, F. Liu, F. Xu, and B. Li, "An online framework for joint network selection and service placement in mobile edge computing," *IEEE Transactions on Mobile Computing*, 2021.

[31] Y. Xiao, Q. Zhang, F. Liu, J. Wang, M. Zhao, Z. Zhang, and J. Zhang, "Nfvdeep: Adaptive online service function chain deployment with deep reinforcement learning," in *IEEE/ACM IWQoS*, 2019, pp. 1–10.

[32] P. Jin, X. Fei, Q. Zhang, F. Liu, and B. Li, "Latency-aware vnf chain deployment with efficient resource reuse at network edge," in *IEEE INFOCOM*, 2020, pp. 267–276.

[33] M. O. Ojijo and O. E. Falowo, "A survey on slice admission control strategies and optimization schemes in 5g network," *IEEE Access*, vol. 8, pp. 14 977–14 990, 2020.

[34] H. Wu, L. Chen, C. Shen, W. Wen, and J. Xu, "Online geographical load balancing for energy-harvesting mobile edge computing," in *IEEE ICC*, 2018, pp. 1–6.

[35] A. J. Ferrer, J. Panadero, J. M. Marquès, and J. Jorba, "Admission control for ad-hoc edge cloud," *Future Generation Computer Systems*, vol. 114, pp. 548–562, 2021.

[36] S. Guo, D. Wu, H. Zhang, and D. Yuan, "Resource modeling and scheduling for mobile edge computing: A service provider's perspective," *IEEE Access*, vol. 6, pp. 35 611–35 623, 2018.

[37] G. Baranwal and D. Vidyarthi, "Admission control policies in fog computing using extensive form game," *IEEE Transactions on Cloud Computing*, 2020.

[38] Y. Dimitrakopoulos and A. Burnetas, "The value of service rate flexibility in an m/m/1 queue with admission control," *IISE Transactions*, vol. 49, no. 6, pp. 603–621, 2017.

[39] M. Jain and S. S. Sanga, "State dependent queueing models under admission control f-policy: A survey," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 9, pp. 3873–3891, 2020.

[40] M. Siew, K. Guo, D. Cai, L. Li, and T. Q. Quek, "Let's share VMs: Optimal placement and pricing across base stations in MEC systems," in *IEEE INFOCOM*, 2021, pp. 1–10.

[41] X. Wang and L. Duan, "Dynamic pricing and capacity allocation of uav-provided mobile services," in *IEEE INFOCOM*, 2019, pp. 1855–1863.

[42] Z. Zheng, R. Srikant, and G. Chen, "Pricing for revenue maximization in inter-datacenter networks," in *IEEE INFOCOM*, 2018, pp. 28–36.

[43] C. Liu and J. J. Hasenbein, "Naor's model with heterogeneous customers and arrival rate uncertainty," *Operations Research Letters*, vol. 47, no. 6, pp. 594–600, 2019.

[44] S. Dughmi, "Algorithmic information structure design: a survey," *ACM SIGecom Exchanges*, vol. 15, no. 2, pp. 2–24, 2017.

[45] E. Altman and T. Jimenez, "Admission control to an m/m/1 queue with partial information," in *International Conference on Analytical and Stochastic Modeling Techniques and Applications*. Springer, 2013, pp. 12–21.