IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 July 2023, accepted 27 July 2023, date of publication 1 August 2023, date of current version 7 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3300735

RESEARCH ARTICLE

Optimal Allocation Strategy of Cloud Resources With Uncertain Supply and Demand for SaaS Providers

LONGCHANG ZHANG[1,2], JING BAI[3], AND JIANJUN XU[3], (Member, IEEE)

[1]School of Information Engineering, Suqian University, Suqian, Jiangsu 223805, China
[2]Shenzhen Research Institute, Beijing University of Posts and Telecommunications, Shenzhen, Guangdong 518100, China
[3]School of Management Science and Engineering, Dongbei University of Finance and Economics, Dalian, Liaoning 116025, China

Corresponding author: Longchang Zhang (zlc_041018@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 62032003, in part by the National Social Science Fund of China under Grant 19BTQ028, in part by the Natural Science Foundation of Liaoning Province under Grant 2019-ZD-0496, in part by the Scientific Research Project of Liaoning Provincial Department of Education under Grant LJKZ1022, and in part by the Suqian University Research Initiation Foundation under Grant 2022XRC029.

ABSTRACT How to make full use of IaaS resources while guaranteeing the quality of service is one of the main issues facing IaaS resource allocation. For the existing cloud resource configuration schemes at the IaaS and PaaS levels, the randomness of user access (resource demand) and resource supply of applications, as well as the revenue of application providers, are not considered enough, the optimal allocation strategy of cloud resources with uncertain supply and demand for SaaS providers is proposed. This strategy is based on the SaaS level, which ensures that the SaaS provider's revenue is maximized without violating QoS constraints and also facilitates the IaaS provider to allocate and fully utilize IaaS resources accurately. The strategy proposes not only optimal allocation strategies for IaaS resources under three scenarios: uncertain demand and certain supply, certain demand and uncertain supply, uncertain demand and uncertain supply, but also establishes quantitative models of resources and demand. The effectiveness and efficiency of the three algorithms are verified through experiments, and the results shows that the revenue of the SaaS provider and IaaS resource utilization are effectively improved without violating the QoS constraint under the condition of uncertain supply and demand at the same time.

INDEX TERMS SaaS provider, IaaS resources, optimal allocation strategy, maximize revenue, uncertain supply and demand.

I. INTRODUCTION

Cloud computing, which offers IaaS, PaaS, and SaaS services to various users via the Internet, has been widely used in recent years. In particular, SaaS providers that offer services to users use leased infrastructure to deploy application services/programs to cloud platforms to reduce construction and maintenance costs [1]. Gartner has continuously released the IaaS market tracking data of cloud computing showing that the global cloud computing market maintains high growth in the number of SaaS products and users, Amazon Web Services (AWS), Oracle Certified Professional (OCP), Google Cloud Platform (GCP), Microsoft Azure, IBM Cloud Services, Sale force. Furthermore, some PaaS platforms and multi-tenant public clouds, Amazon EC2, Microsoft Azure-IaaS, Alibaba Cloud, IBM Cloud, and other IaaS resources, have a high market share. In multi-tenant public clouds, many SaaS share the same IaaS infrastructure, and competition for resources is frequent. Moreover, violations of SaaS QoS constraints and even downtime are inevitable when the IaaS load is overloaded and resource utilization is pursued excessively; this has a destructive impact on the user experience and consequently causes severe losses to the SaaS provider. Allocating more resources to ensure the QoS constraints

The associate editor coordinating the review of this manuscript and approving it for publication was Nitin Gupta.

of SaaS, but the demand for SaaS is indeed uncertain, the amount of resources allocated is challenging to determine. Over-allocation of resources reduces resource utilization and IaaS revenue. Therefore, the allocation of IaaS resources to effectively improve system resource utilization while satisfying the QoS constraint of SaaS has become a pressing challenge [2].

In response to the above issues, resource allocation research can be carried out from the IaaS, PaaS, and SaaS layers to achieve a guaranteed QoS constraint for SaaS and improve cloud resource usage. The goal of resource allocation on IaaS layer or PaaS layer is to ensure high resource utilization under QoS constraints, thereby achieving maximum revenue for IaaS providers. The intention of service selection and combination based on QoS [3] on SaaS layers is to obtain the QoS optimal service.

However, an efficient SaaS resource allocation strategy aims to improve cloud resource utilization while guaranteeing QoS constraints and maximizing the benefits of both SaaS providers and IaaS providers. The price of cloud resources [4], [5] is an essential factor in determining the revenue of SaaS providers and should be fully considered when allocating resources. There is uncertainty in user access and IaaS resource load, resulting in existing schemes that cannot simultaneously guarantee QoS constraints and fully utilize cloud resources. This paper proposes a two-stage optimal allocation strategy for cloud resources with uncertain supply and demand to address the existing problems. Its basic idea is to allocate cloud resources based on a SaaS revenue perspective, where SaaS providers actively request the amount of cloud resources under the condition of maximum expected revenue based on user volume, revenue and cloud resource prices, and the solution from this perspective does not violate QoS constraints. This way helps IaaS providers allocate cloud resources accurately and thus effectively improve resource utilization. Compared to the existing research results, this article provides useful supplements from the following aspects: (1) maximizing the benefits of SaaS providers; (2) SaaS providers propose resource requirements, and there is no violation of QoS constraints at the IaaS level. IaaS providers accurately grasp resource requirements, laying the foundation for further improving resource utilization; (3) quantifying the resource demand for cloud resources, virtual resources, and applications, laying the foundation for effective integration of demand and supply; (4) solving the problem of insufficient resource allocation in a random supply and demand cloud environment.

Section II summarizes the existing research; Section III describes the problem and introduces the notation and assumptions involved in the paper; Section IV models IaaS resources and virtual resources; Section V details three strategies for optimal allocation of resources under uncertain demand and uncertain supply; Section VI evaluates the effectiveness of the strategies proposed in the paper through experiments; Section VII concludes the paper and introduces the following research directions.

II. RELATED RESEARCH

In recent years, a great deal of research has been carried out on cloud resource allocation. The problem of cloud resource allocation that satisfies QoS constraints and can effectively improve resource utilization is one of the problematic issues, and this section will briefly describe the progress of research work related to this problem.

A. CLOUD RESOURCE ALLOCATION WITH APPLICATION QoS CONSTRAINTS BASED ON THE IaaS PROVIDER PERSPECTIVE

Applications deployed to virtual machines need to consider the problem of placing virtual machines to physical nodes that satisfy different application QoS constraints while being able to achieve optimal resource usage. There are three main approaches to achieve such problems: modeling resource allocation as a multi-objective optimization problem, which is then solved using heuristic algorithms; considering the topological characteristics of physical nodes and application deployment, mapping the resource allocation problem to a graph matching. The problem is solved by machine learning methods to predict the resource consumption, load variation and performance metrics of the application and adjust the resource allocation.

Heuristic algorithm. By modeling the application QoS demand, load, and cloud resource demand distribution, resource allocation is defined as a multi-objective optimization problem with multiple constraints and a genetic algorithm-based multi-objective optimization algorithm is proposed [6]. There are also some studies on resource allocation schemes based on immune cloning algorithms [7], ant colony algorithms [8], symbiotic biological search [9], and other heuristic algorithms. The main target of this class of methods is to pursue resource search accuracy and speed that meet application QoS requirements. Although dozens of such algorithms have been developed to improve application QoS, a heuristic multi-objective optimization algorithm that can effectively adapt to such dynamically changing resource allocation is still lacking so far due to the high uncertainty of application QoS requirements and resource load.

Graph matching algorithm. In the analysis of application deployment, it is found that the topology of cloud resource nodes is represented as a complex heterogeneous network graph, and the application deployment requirements proposed by different tenants can also be represented as a heterogeneous network graph with multi-dimensional performance requirement attributes. Therefore, the problem of cloud virtual machine placement for large-scale applications is mapped to a sub-graph query matching problem of nodes in the topology graph of the cloud resource, and the heterogeneous graph query matching method based on the partial order relationship can obtain a set of cloud resource nodes that meet user requirements [10]. In addition, the literature [11] has also done some exploration of cloud resource allocation based on graph theory. Although this method can achieve agile delivery and deployment of applications, there is still much room for improvement in the problems of solution accuracy, dynamic changes in QoS of cloud resources and applications, and cloud resource utilization.

Machine learning algorithm. Based on the characteristics of cloud resource sharing, researchers obtain job run-time monitoring data and cloud resource allocation information, establish heuristic rules for job classification and optimal cloud resource allocation, and apply the rules to a Bayesian optimization algorithm for resource allocation [12]. In order to adapt well to the dynamic changes in the cloud environment toward service-based systems, a model-free online learning algorithm is applied to solve the complex problem of guaranteeing system performance due to changes in user concurrency, which is achieved by repeating the ''execution-accumulation-learning-decision'' process. This method can continuously accumulate the empirical data and optimize decision results by repeating the process [13]. There are also studies on resource allocation based on algorithms such as neural networks [14], Markov prediction [15], [16], and supervised learning [17]. The fundamental problems with this class of methods are that the accuracy of the prediction of user QoS demand and resource operation is difficult to guarantee, and the violation of QoS constraints cannot be avoided as there is always some deviation between the predicted and actual results due to complex factors. In addition, the algorithms need to be trained and tuned, and thus the overhead system problem arises.

In addition, there are resource allocation methods based on cybernetics [18], game theory [19], [20], etc. The generation of cloud service user concurrency is highly uncertain. A single resource allocation cannot keep the cloud service running to meet the QoS constraints, so it needs to dynamically adjust the resource allocation while the cloud service is running, and adaptive resource adjustment can more effectively cope with real-time changes in the cloud environment [21], as in the literature [22], [23]. This approach is useful for cloud environments and applications that change more frequently. QoS requirements are difficult to cope with and incur a large additional system overhead. The use of resource reservation to guarantee QoS constraints is an effective approach. The literature [24] addresses the problem that if only one QoS metric demand (even a non-functional QoS parameter) cannot be satisfied in the entire reservation request. Using resource reservation is an effective method to ensure QoS constraints. They improved the calculation method of QoS deviation distance and reduced the rejection rate of reservation requests in the negotiation phase. However, reserving too many resources can lead to severe resource waste, even though it can effectively reduce the violation of QoS constraints.

B. CLOUD RESOURCE ALLOCATION BASED ON THE PaaS OR SaaS PROVIDER'S PERSPECTIVE OF APPLICATION QoS CONSTRAINTS

The main task of cloud resource providers is to ensure the adequate provision of resource utilization under user QoS constraints. Current resource allocation mechanisms mainly focus on the IaaS layer, with insufficient consideration of the application characteristics of the PaaS layer. Applications deployed on the PaaS platform vary significantly in their usage of resources, and accesses show different characteristics over time. Different types of applications are deployed together by predicting the changes in application request rates and the overhead of each resource. Applications with larger request volumes are divided into multiple units with relatively fixed resource overheads for processing to use server resources efficiently. This solution is helpful for applications with request volumes. However, the accuracy of the characterization of resource overhead and application request rate variation directly affects the effectiveness of the resource allocation scheme, and its accuracy is difficult to guarantee in the current approach [25].

There is no research related to cloud resource allocation based strictly on the SaaS perspective. Most of the research is based on QoS service selection and service combination [26], [27], this type of problem does not involve cloud resource allocation, and there are many research results. As described in reference [27], a cloud model is used to describe the QoS of services, and a TOPSIS method based on the cloud model is designed to select QoS stable services. Various uncertainties lead to alternative service QoS is uncertain or even service is failed, violation of QoS constraints often occurs, and shortcomings in terms of SaaS provider revenue considerations.

C. SUMMARY AND COMMENTS

On the basis of achieving dynamic cloud resource supply and elasticity, IaaS and PaaS providers attempt to adopt various possible methods to achieve efficient resource utilization under QoS constraints, in order to maximize the benefits of IaaS providers. Unfortunately, random resource demands often make it difficult for IaaS providers to make accurate resource allocation decisions. For example, Alibaba Cloud provides a cloud resource service that can be self-obtained at any time, autoscaling, and cost guaranteed. This flexibility also poses a huge challenge to the supply chain. To meet customer service levels while maximizing cloud resource utilization and reducing supply chain costs, Alibaba is soliciting solutions from around the world [28].

The relevant researches and our work are summarized and compared in Table 1. In summary, the current research work based on IaaS providers and PaaS providers still cannot effectively address the under-utilization of cloud resources under the application of QoS constraints, especially under the conditions of substantial uncertainty in the concurrency of service users and highly randomized cloud resource loads. More importantly, previous studies have focused on maximizing cloud resources while meeting QoS constraints, without considering the benefits of SaaS providers, resulting in increased costs. In our understanding, solving the problem of precise resource allocation decision-making should start from two aspects: the randomness of demand and the randomness

TABLE 1. Summary of the previous literature related to cloud resource allocation.

Literature	Objective	Perspective	Solution Method	Existing problems
Q. Li et al. [6]	Minimize the violation rate of service level objectives for multiple applications, reduce the use of physical nodes.	IaaS provider	Genetic algorithm	Violation of QoS constraints and insufficient utilization of resources, SaaS provider revenue cannot be guaranteed.
D. Sun al. [7]	Improve cloud system availability, load balancing deviation and valid time.		Immune clonal algorithms	
S. K. Addya et al. [8]	Maximize the overall revenue of SPs.		Ant colony algorithms	
A. Belgacem et al. [9]	Minimize the execution time of resources allocation, improve the QoS given to cloud users.		Symbiotic biological search	
W. Guo et al. [10]	Improve the efficiency of multi-dimensional heterogeneous cloud resource placement strategy.		Graph matching	
G. J. Kuang et al. [11]	Maximize satisfaction between cloud tasks and cloud resources.		Graph theory	
Y. W. Wu et al. [12]	Improve QoS and reduce costs.		Bayesian optimization	
Y. M. Yan et al. [13]	Ensure application performance by dynamically adjusting resource allocation.		reinforcement learning	
J. J. Sun et al. [14]	Maximize market surplus and overall prestige information of participants.		back propagation neural network	
P. Zhou et al. [15] C. Liu et al. [16]	Achieve rapid cloud service recovery and to improve the reliability of cloud services. Minimize the average completion time of tasks under migration energy budget.		Markov process	
M. Chen et al. [17]	Guarantee a reliable Quality of Service (QoS)		supervised learning	
L. Yu et al. [18]	Improve the utilization ration of virtual resources.		control theory	
Y. Ying et al. [19], Y. Wang et al. [20]	Guarantee the profit of service providers and increase infrastructure provider's revenue; Balance the utilities of users and service providers in service transactions.		game theory	
P. Haratian et al. [22], A. Alsarhan et al. [23]	Reduce the number of SLA violations.		adaptive resource allocation	
Z. A. Wu et al. [24]	Guarantee Quality of Service		advance resource reservation	
H. Wei et al. [25]	Save various resources and keep service quality	PaaS provider	elastic resource management mechanism	
L. Qi et al. [26], H. Ma et al. [27]	Improve Quality of Service	SaaS provider	Context-Aware, cloud model theory	
Our study	Not violate QoS constraints at IaaS and PaaS layers, improve IaaS resource utilization, and maximize application provider revenue.	SaaS provider	Inventory theory	There may be some users whose QoS cannot be guaranteed.

of load. IaaS and PaaS ultimately aim to provide services for applications, and the benefits of SaaS providers should be fully considered to ensure the sustainable development of cloud platforms. This paper proposes a cloud resource allocation strategy that maximizes the expected revenue of the SaaS provider based on the SaaS provider's perspective, considering the resource allocation under uncertain demand and uncertain supply conditions, and maximizing the SaaS provider's revenue while ensuring that there is no violation of QoS constraints.

III. PROBLEM, ASSUMPTIONS AND NOTATIONS

A. PROBLEM DESCRIPTION

In the cloud computing ecosystem, the software and services provided by the SaaS provider to the users run on the infrastructure leased from the IaaS provider, which is only responsible for operating and maintaining the SaaS, thus effectively reducing the SaaS provider's costs in infrastructure construction, operation, and maintenance. At the beginning of a service period, the SaaS provider leases IaaS to run its SaaS to provide services at a certain rental rate to meet the QoS requirements of users and is responsible for SaaS operation and maintenance. As the number of SaaS user visits is not only influenced by subjective factors such as users' needs, preferences, loyalty, service reputation, and QoS expectations but also largely influenced by many objective factors such as service quality, natural environment, network environment, and computing environment, resulting in a high degree of uncertainty in the actual number of user visits. As IaaS providers are not only affected by factors such as hardware, network, and resource allocation algorithms in the process of providing infrastructure services to multiple tenants at the same time, but also primarily influenced by uncontrollable factors such as uncertainty in user demand and the dynamics of resource consumption of SaaS, which results in uncertainty amount of SaaS user access they can support (i.e., the number of resources under the condition that QoS constraints are met supply uncertainty). Aiming at the problem of uncertain access and resource allocation of SaaS users under the IaaS lease model, a minimum optimal IaaS resource allocation strategy is proposed to maximize the expected revenue of SaaS providers without violating service QoS constraints.

B. NOTATION

TABLE 2. Relevant symbols are defined.

Symbols	Description
	(1) Parameters
D	the total number of users forecast by the SaaS provider
R	the amount of virtual resources consumed to provide the service to a user
Pu	the price of providing the service to one user
C	the unit rental cost of the IaaS resource
L	the unit cost of virtual resources out of stock (the cost of missing a unit of virtual resources)
H	the unit idle cost of a virtual resource (the loss of a unit of idle virtual resource)
	(2) Random Variables
u	the amount of virtual resource demand (i.e. $D\times R$), is a random variable whose probability density function, cumulative distribution function and expectation are f(u), F(u), U, respectively, $0 \le u \le \mathrm{D0} < +\infty$, D0 is a constant.
y	the amount of IaaS virtual resources that can be Amount the amount of virtual resources supported per unit IaaS computing instance, is a random variable whose probability density function, cumulative distribution function and expectation aref(y),F(y),Y, respectively ,$0 \le y \le k < +\infty$, where output$Q = \mathrm{X}\times\mathrm{E}(y)$
	(3) Decision variables
Q	The demanded amount of virtual resources
X	The allocated amount of IaaS resources (i.e., computing instances)

C. ASSUMPTION

This study builds a mathematical model based on the following assumptions: (1) There is only one IaaS provider in the system, who offers the infrastructure services to multiple tenants, and there is competition for resources among the tenants. (2) The SaaS in the system is a single product, the service period is fixed, and the distribution of the number of users accessing the SaaS during the service period can be described by a function. (3) The distribution of the amount of IaaS virtual resources during the service period can be described by a function and is independent of the distribution of the number of users. (4) The SaaS provider is rational and risk-averse neutral. (5) The amount of IaaS resources required by each user instance can be quantified and is the same or similar in quantity. (6) The amount of IaaS resources allocated is insufficient, and SaaS denies service to an excessive number of users, and this denial process does not affect the user's willingness to continue using it. (7) IaaS resource over-provisioning results in non-efficient use of resources (i.e., idle), where redundant resources are not returnable but can enhance the QoS of the service, improve the user experience, and increase the hidden revenue of the SaaS provider, and are measurable. Thus, the cost of idleness can be lower than the cost of leasing.

IV. THE RESOURCE MODEL

The resources required for SaaS operation are mainly computing resources such as CPU, memory, and external memory. A vector $\mathbf{Rq} = (r1, r2 \cdots, rn)$ describes the resources required to provide services to each user under QoS constraints, and the total resource requirements are linearly related to the number of user accesses [25], where the resource requirements for providing services to a user need to be determined. In order to satisfy QoS constraints, there are certain dependencies between them $r1, r2 \cdots rn$, and thus only one class of resources needs to be quantified to determine the other resource requirements. SaaS is packaged in virtual machines that run on specific IaaS compute instances in cloud computing, thus the demand, virtual resources, and compute resources need to be quantified to form an articulation relationship for subsequent resource allocation. The following are definitions of the relevant concepts and relationships.

Definition 1: IaaS computing resources: IaaS resources mainly include hardware resources, software resources, and network resources. SaaS providers mainly lease hardware and network resources, such as CPU/GPU, memory, external memory, I/O devices, switches, and bandwidth. IaaS computing resources refer to the abstraction of these hardware and network resources, such as CPU, storage, network.

Definition 2: IaaS computing instance: A hardware platform consists of computing resources that can run software systems independently. Different levels of instances are configured according to the performance requirements of the SaaS provider, and an instance with a basic configuration is set as a standard computing instance (i.e., it is set as a measurement unit of the instance's computing power, called Calculation Instance Power Unit - CIPU). For example, if a standard computing instance consists of 1G memory, 200G external memory, 100M network bandwidth, and 0.5G CPU, then a computing instance with 2 CIPU units consists of 2G memory, 400G external memory, 200M network bandwidth and 1G CPU.

Definition 3: Virtual computing instance: A system that contains some of the computing resources of an IaaS computing instance and is capable of running SaaS independently, simulated by software with the full functionality of an IaaS computing instance. A virtual compute instance runs on one or more specific IaaS compute instances and is measured in CIPUs.

Definition 4: IaaS virtual resource volume: A measure of the number of virtual compute instances required for a SaaS to provide services to a user under conditions that guarantee the user's QoS (e.g., response time, throughput, etc.) requirement is called the IaaS virtual resource volume. Given that SaaS provides services to a user, the amount of IaaS virtual resources that a user instance needs to configure in a service period under the condition of QoS constraints is R CIPU, i.e., the IaaS virtual resource amount R.

In this study, the amount of user access is converted into virtual resource requirements, and then the virtual resource requirements are converted into virtual compute instance requirements, which are then packaged to run on specific compute instances.

V. IAAS OPTIMAL RESOURCE ALLOCATION STRATEGY

The process of leasing IaaS resources to run SaaS from SaaS providers to provide services to users involves four parts of sales revenue and cost expenditure: service sales revenue, IaaS resource leasing cost, out-of-stock cost, and IaaS resource idle cost. The minimum optimal allocation strategy of IaaS resources is established under the uncertainty of both user access and IaaS virtual resource supply. The service sales revenue is earned by the SaaS provider for providing services to users; the IaaS resource leasing cost is the SaaS provider's expenditure for leasing computing resources to run SaaS, which must guarantee the users' QoS requirements for SaaS; the out-of-stock cost is the loss caused by the SaaS provider having to refuse to provide services to subsequent users because the access volume has reached its limit; the loss due to the configured IaaS resources are able to support more virtual resources than the amount of user demand and the cost of resource idleness, which arises from two factors: (1) virtual resource overload enhances QoS and user experience and somehow invisibly increases the SaaS provider's revenue, and (2) IaaS providers recycle the use of tenants' idle reserved resources in the form of financial incentives [29]. Other revenues and costs involved in the actual resource allocation process can continue to be added without affecting the core idea of this study's minimum optimal allocation approach for IaaS resources.

During the service period, the number of users accessing SaaS may be randomly variable, i.e., demand is uncertain; there also exists a fixed group of users of SaaS (especially enterprise users, where the volume of users remains constant over time), i.e., demand is deterministic. As a result, fewer IaaS infrastructure service tenants have sufficient computing instances to obtain a deterministic amount of IaaS virtual resources to meet QoS requirements, i.e., the supply of IaaS virtual resources is certain. There are also more IaaS infrastructure service tenants forming resource competition among tenants. Under the condition of satisfying QoS demand, the IaaS resource provider will dynamically allocate resources for each tenant according to the virtual resource demand and the total resource quantity, which leads to uncertainty in the supply of IaaS virtual resources for the tenant. Three IaaS resource optimal allocation strategies are designed to address the existing problems in the following. The relationship between the three optimal resource allocation strategies is shown in Figure 1.

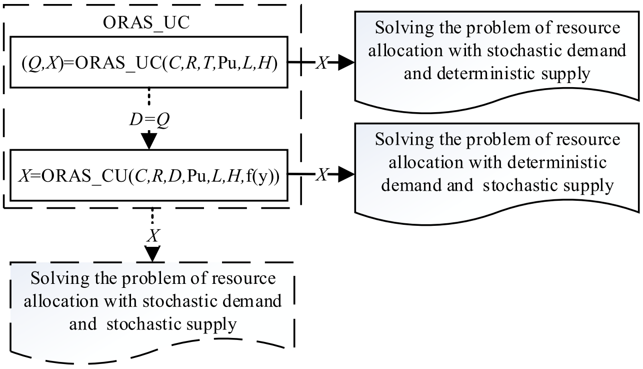

ORAS_UC
(Q,X)=ORAS_UC(C,R,T,Pu,L,H)
X
Solving the problem of resource allocation with stochastic demand and deterministic supply
D=Q
X=ORAS_CU(C,R,D,Pu,L,H,f(y))
X
Solving the problem of resource allocation with deterministic demand and stochastic supply
X
Solving the problem of resource allocation with stochastic demand and stochastic supply

FIGURE 1. The relationship between the three optimal resource allocation strategies.

A. OPTIMAL RESOURCE ALLOCATION STRATEGY WITH UNCERTAIN DEMAND AND CERTAIN SUPPLY (ORAS_UC)

The following calculates the demand for virtual resources under demand uncertainty. In this case, the SaaS provider tries to meet the market demand while maximizing its expected revenue while satisfying the user QoS constraints, i.e., by maximizing the following function:

$$max \left\{P \times E\left[min\left(Q,u\right)\right] - C \times \left(Q/Y\right) - L \times E\left[\left(u-Q\right)\right]^{+} -H \times E\left[\left(Q-u\right)\right]^{+}\right\} \quad (1)$$

Because the price of providing services to users is Pu, the price of selling virtual resources is $P = Pu\big/R$. Uncertainty in the number of users leads to uncertainty in the demand for virtual resources. The demand for virtual resources u is a random variable obeying the probability density function and cumulative distribution function $\mathrm{f}\left(u\right)$, $\mathrm{F}\left(u\right)$, respectively. In dealing with uncertain variables u in the objective function, the objective function is transformed into the expected objective function, as shown in Eq. (2):

$$\mathrm{G}\left(\mathrm{Q}\right) = \int_{0}^{Q} \left[\left(P \times u\right) - H \times \left(Q-u\right)\right] \mathrm{f}\left(\mathrm{u}\right) du + \int_{Q}^{+\infty} \left[P{\times}Q{-}L \times \left(u{-}Q\right)\right] \mathrm{f}\left(\mathrm{u}\right) du - C \times Q\big/Y \quad (2)$$

Proposition 1: The optimal demand for IaaS virtual resources Q over a service period satisfies Eq. (3) under uncertain demand and deterministic supply conditions.

$$\int_0^Q \mathrm{f}(\mathrm{u})du = \frac{P+L-C/Y}{P+L+H} \tag{3}$$

Proof: Expand G (Q)

$$\begin{aligned}\mathrm{G}(\mathrm{Q}) = & \int_0^Q (P\times u)\,\mathrm{f}(\mathrm{u})\,du \\ & -\int_0^Q (H\times Q)\,\mathrm{f}(\mathrm{u})\,du \\ & +\int_0^Q (H\times u)\,\mathrm{f}(\mathrm{u})\,du + \int_Q^{+\infty} (P\times Q)\,\mathrm{f}(\mathrm{u})\,du \\ & -\int_Q^{+\infty} (L\times u)\,\mathrm{f}(\mathrm{u})\,du \\ & +\int_Q^{+\infty} (L\times Q)\,\mathrm{f}(\mathrm{u})\,du - C\times Q/Y\end{aligned}$$

Finding the first-order derivative of the function G (Q) with respect to Q:

$$\begin{aligned}\frac{\partial \mathrm{G}(\mathrm{Q})}{\partial Q} = & (P\times Q)\,\mathrm{f}(\mathrm{Q}) - \int_0^Q H\times \mathrm{f}(\mathrm{u})du - Q\times H\times \mathrm{f}(\mathrm{Q}) \\ & + H\times Q\times \mathrm{f}(\mathrm{Q}) - \int_\infty^Q P\times \mathrm{f}(\mathrm{u})\,du \\ & - P\times Q\times \mathrm{f}(\mathrm{Q}) + L\times Q\times \mathrm{f}(\mathrm{Q}) \\ & - \int_\infty^Q L\times \mathrm{f}(\mathrm{u})\,du - L\times Q\times \mathrm{f}(\mathrm{Q}) - C/Y \\ \frac{\partial \mathrm{G}(\mathrm{Q})}{\partial Q} = & -\int_0^Q H\times \mathrm{f}(\mathrm{u})\,du \\ & -\int_\infty^Q P\times \mathrm{f}(\mathrm{u})\,du - \int_\infty^Q L\times \mathrm{f}(\mathrm{u})\,du - C/Y \\ = & (P+L)\times \int_Q^\infty \mathrm{f}(\mathrm{u})\,du - H \\ & \times \int_0^Q \mathrm{f}(\mathrm{u})du - C/Y\end{aligned}$$

$\int_0^Q \mathrm{f}(\mathrm{u})du = 1 - \int_Q^{+\infty} \mathrm{f}(\mathrm{u})du$. So, we have the following equation:

$$\begin{aligned}\frac{\partial \mathrm{G}(\mathrm{Q})}{\partial Q} = & (P+L)\times \left[1-\int_0^Q \mathrm{f}(\mathrm{u})\,du\right] - H \\ & \times \int_0^Q \mathrm{f}(\mathrm{u})\,du - C/Y = P+L-C/Y \\ & -(P+L+H)\int_0^Q \mathrm{f}(\mathrm{u})\,du\end{aligned}$$

Let the first order derivative function be 0, and the optimal solution Q satisfies the following equation.

$$\int_0^Q \mathrm{f}(\mathrm{u})du = \frac{P+L-C/Y}{P+L+H}$$

If the optimal solution Q is proved to satisfy the above equation, it is sufficient to prove the convex function by finding the second-order derivative of the function G (Q) with respect to Q:

$$\frac{\partial^2 \mathrm{G}(\mathrm{Q})}{\partial Q^2} = -(P+L+H)\times \mathrm{f}(\mathrm{Q}) < 0$$

The minimum optimal virtual resource requirement Q is obtained from Eq. (3). Since the amount of virtual resources generated by the IaaS instance is deterministic Y (i.e., the supply is deterministic), IaaS calculates the minimum optimal allocation of the instance as follow: $X = Q/Y$ (4).

Algorithm 1 ORAS_UC

Input: the unit price of IaaS resource lease C, each user instance needs to consume R CIPU virtual resources, probability distribution table T of virtual resource demand u as a random quantity, each user service charge Pu, IaaS virtual resource shortage cost L, IaaS virtual resource idle cost H, IaaS instance produces virtual resource quantity Y.

Output: optimal virtual resource demand Q and IaaS resource allocation amount X

Steps:

Step 1: Obtain the sales price of virtual resources according to $P = \mathrm{Pu}/R$.

Step 2: According to Eq. (3) and check the probability distribution table T, get the minimum optimal virtual resource demand Q.

Step 3: Calculate the minimum optimal IaaS resource allocation amount X according to Eq. (4).

B. OPTIMAL RESOURCE ALLOCATION STRATEGY WITH CERTAIN DEMAND AND UNCERTAIN SUPPLY (ORAS_CU)

The following calculates the amount of IaaS resources allocated under supply uncertainty. In this case, user access D is determined, the demand for virtual resources $\mathrm{D1} = D\times R$, the price per unit of virtual resources sold $P = \mathrm{Pu}/R$, the SaaS provider leases as many IaaS resources as possible to meet the market demand while obtaining the maximum revenue for itself, the decision variable IaaS computing instance allocation amount X at this time. The virtual computing instances that can be supported per IaaS computing instance is a random variable y, obeying the probability density function, cumulative distribution function $\mathrm{f}(y), \mathrm{F}(y)$, respectively, that is, the amount of virtual output resources $Q = X\times y$, with the following formula:

$$\begin{aligned} max\,\{P\times E\,[min\,(\mathrm{D1}, Q)] - C\times X - L\times E\left[(\mathrm{D1}-Q)^+\right] \\ -H\times E\left[(Q-\mathrm{D1})^+\right]\} \end{aligned} \tag{4}$$

Due to the uncertainty of virtual resource availability, the objective function is transformed into the desired objective function when dealing with uncertain variable Q in the objective function, as shown in Eq. (5):

$$\begin{aligned} &\mathrm{G}(\mathrm{X}) \\ &= \int_0^{\frac{\mathrm{D1}}{X}} [P\times Q - L\times(\mathrm{D1}-Q)]\,\mathrm{f}(\mathrm{y})\,dy \\ &\quad + \int_{\frac{\mathrm{D1}}{X}}^{+\infty} [P\times \mathrm{D1} - H\times(Q-\mathrm{D1})]\,\mathrm{f}(\mathrm{y})\,dy - C\times X \end{aligned} \tag{5}$$

Proposition 2: The optimal leasing strategy for a single service period X satisfies Eq. (6) under uncertain supply and deterministic demand conditions.

$$\int_0^{\frac{\mathrm{D1}}{X}} y\times \mathrm{f}(\mathrm{y})dy = (C+1)\big/(P+L+H) \tag{6}$$

Proof: Expand G (X)

$$\begin{aligned} &G(X) \\ &= \int_0^{\frac{\mathrm{D1}}{X}} [P\times X\times y - L\times(\mathrm{D1}-X\times y)]\,\mathrm{f}(\mathrm{y})\,dy \\ &\quad + \int_{\frac{\mathrm{D1}}{X}}^{+\infty} [P\times\mathrm{D1} - H\times(X\times y-\mathrm{D1})]\,\mathrm{f}(\mathrm{y})\,dy - C\times X \\ &= \int_0^{\frac{\mathrm{D1}}{X}} P\times X\times\mathrm{f}(\mathrm{y})\,dy - \int_0^{\frac{\mathrm{D1}}{X}} L\times D1\times\mathrm{f}(\mathrm{y})\,dy \\ &\quad + \int_0^{\frac{\mathrm{D1}}{X}} L\times X\times y\times\mathrm{f}(\mathrm{y})\,dy + \int_{\frac{\mathrm{D1}}{X}}^{+\infty} P\times D1\times\mathrm{f}(\mathrm{y})\,dy \\ &\quad - \int_{\frac{\mathrm{D1}}{X}}^{+\infty} H\times X\times y\times\mathrm{f}(\mathrm{y})\,dy \\ &\quad + \int_{\frac{\mathrm{D1}}{X}}^{+\infty} H*\mathrm{D1}*\mathrm{f}(\mathrm{y})\,dy - C*X \end{aligned}$$

Finding the first -order derivative of the function G (X) concerning X:

$$\begin{aligned} \frac{\partial\mathrm{G}(\mathrm{X})}{\partial X} &= \int_0^{\frac{\mathrm{D1}}{X}} P\times y\times\mathrm{f}(\mathrm{y})\,dy \\ &\quad + P\times X\times\frac{\mathrm{D1}}{X}\times f\left(\frac{\mathrm{D1}}{X}\right) \\ &\quad \times\left(-\frac{\mathrm{D1}}{X^2}\right) - L\times\mathrm{D1}\times\mathrm{f}\left(\frac{\mathrm{D1}}{X}\right)\times\left(-\frac{\mathrm{D1}}{X^2}\right) \\ &\quad + \int_0^{\frac{\mathrm{D1}}{X}} L\times y\times\mathrm{f}(\mathrm{y})\,dy + L\times X\times\frac{\mathrm{D1}}{X}\times\mathrm{f}\left(\frac{\mathrm{D1}}{X}\right) \\ &\quad \times\left(-\frac{\mathrm{D1}}{X^2}\right) - P\times\mathrm{D1}\times\ \mathrm{f}\left(\frac{\mathrm{D1}}{X}\right)\times\left(-\frac{\mathrm{D1}}{X^2}\right) \\ &\quad - \int_{\frac{\mathrm{D1}}{X}}^{+\infty} H\times y\times\mathrm{f}(\mathrm{y})\,dy + H\times X \\ &\quad \times\frac{\mathrm{D1}}{X}\times\mathrm{f}\left(\frac{\mathrm{D1}}{X}\right)\times\left(-\frac{\mathrm{D1}}{X^2}\right) \\ &\quad - H\times\mathrm{D1}\times\mathrm{f}\left(\frac{\mathrm{D1}}{X}\right)\times\left(-\frac{\mathrm{D1}}{X^2}\right) - C \end{aligned}$$

Reduce and we have

$$\begin{aligned} \frac{\partial\mathrm{G}(\mathrm{X})}{\partial X} &= \int_0^{\frac{\mathrm{D1}}{X}} P\times y\times\mathrm{f}(\mathrm{y})\,dy + \int_0^{\frac{\mathrm{D1}}{X}} L\times y\times\mathrm{f}(\mathrm{y})\,dy \\ &\quad - \int_{\frac{\mathrm{D1}}{X}}^{+\infty} H\times y\times\mathrm{f}(\mathrm{y})\,dy - C \end{aligned}$$

Let the first-order derivative function be 0, and the optimal solution X satisfies the following equation

$$\int_0^{\frac{\mathrm{D1}}{X}} y\times\mathrm{f}(\mathrm{y})dy = (C+1)\big/(P+L+H)$$

If the optimal solution X is proved to satisfy the above equation, it is sufficient to prove that it is a convex function by finding the second-order derivative of the function G (X) concerning X:

$$\frac{\partial^2\mathrm{G}(\mathrm{X})}{\partial X^2} = (P+L+H)\times\frac{\mathrm{D1}}{X}\times\mathrm{f}\left(\frac{\mathrm{D1}}{X}\right)\times\left(-\frac{\mathrm{D1}}{X^2}\right) < 0$$

The function is obtained as a convex function, with the point of maximum value taken to be the point where the first- order derivative is 0.

Eq. (6) is an increasing function. There is only one root, which is very suitable for using an efficient dichotomous search algorithm to solve. This section uses the dichotomous search method to solve, such as Algorithm 2.

Algorithm 2 Finding Function Solution by Binary Search (FFS_BS)

```
Input: Solve $\mathrm{g}(\mathrm{x}) = 0$
Output: Approximate solution $x^*$
Steps:
Step 1: Determine the initial interval $(l, r)$ of the objective function $\mathrm{g}(\mathrm{x})$, where $\mathrm{g}(l) < 0$ and $\mathrm{g}(r) > 0$, the termination condition $\varepsilon$ (i.e., $|\mathrm{l}-\mathrm{r}| < \varepsilon$).
Step 2: Calculates $\mathrm{g}\left((l+r)\big/2\right)$.
Step 3: if $\mathrm{g}\left((l+r)\big/2\right) < 0$, the approximate solution of function is within the interval $\left[(l+r)\big/2, r\right]$, then $l = (l+r)\big/2$.
Step 4: if $\mathrm{g}\left((l+r)\big/2\right) > 0$, indicating that the extreme value point is within the interval $\left[l, (l+r)\big/2\right]$, $r = (l+r)\big/2$.
Step 5: if $|\mathrm{l}-\mathrm{r}| < \varepsilon$, $x^* = 2\times\mathrm{D1}\big/(l+r)$, terminate the iteration; otherwise, return to Step 2.
```

Based on the above modelling, solution and algorithm implementation, the following ORAS_UC algorithm is designed, as shown in Algorithm 3.

C. OPTIMAL RESOURCE ALLOCATION STRATEGY WITH UNCERTAIN DEMAND AND UNCERTAIN SUPPLY (ORAS_UU)

The uncertain demand and uncertain supply refer to the uncertainty in demand for virtual resources and the uncertainty in the supply of virtual resources due to the uncertainty in user access. Therefore, when calculating the minimum optimal allocation of IaaS instances, both uncertainty in demand

```
Algorithm 3 ORAS_CU
Input: IaaS resource rental unit price $C$, each user instance needs to consume virtual resources $R$ CIPU, virtual resource demand is $D$, per user service fee Pu, IaaS virtual resource shortage cost $L$, IaaS virtual resource idle cost $H$, the amount of IaaS instance supporting virtual resources $y$ is a random variable with probability density function $\mathrm{f}(\mathrm{y})$.
Output: Optimal allocation amount of IaaS instance $X$
Steps:
Step 1: Yield the virtual resource sales price based on $P = \mathrm{Pu}/R$
Step 2: Yield the virtual resource demand based on $\mathrm{D1} = D \times R$
Step 3: Call the FFS_BS algorithm to find the solution to equation (6) and obtain the minimum optimal IaaS resource allocation $X$
```

for virtual resources and uncertainty in supply need to be considered.

```
Algorithm 4 ORAS_UU
Input: IaaS resource rental unit price $C$, each user instance needs to consume virtual resources $R$ CIPU, each user service fee Pu, the uncertainty of user volume leads to uncertainty of virtual resource demand, the virtual resource demand $u$ is a random variable obeying the probability distribution table $T$; IaaS virtual resource shortage cost $L$, IaaS virtual resource idle cost $H$, the amount of virtual resources supported by IaaS instance is a random variable $y$ with probability density function $\mathrm{f}(\mathrm{y})$.
Output: Optimal allocation amount of IaaS instance $X$
Steps:
Step 1: Call ORAS_UC($C, R, T$, Pu,$L, H$) to get $Q$.
Step 2: Call ORAS_CU($C, R, Q$, Pu,$L, H$, $\mathrm{f}(\mathrm{y})$) to get $X$.
```

VI. EXPERIMENTAL ANALYSES

A. ALGORITHM PERFORMANCE ANALYSIS

In this section, experiments show that the Time complexity and Space complexity of the algorithms in this paper are constant orders. The environment and parameter settings are as follows. The hardware configuration is Intel (R) Core (TM) i7-10750H with 2.60 GHz CPU, 16.0 GB of RAM, Windows 10 operation system, and the algorithm is implemented in Python. The minimum optimal resource allocation in ORAS_UC is independent of the number of user accesses, so the impact of user variation on the algorithm's execution time does not need to be considered in the experiments. In ORAS_CU and ORAS_UU, the dichotomous search method is used to obtain the minimum optimal allocation, which is related to the amount of provisioning, so the impact of the change in the mean value of the random variable provisioning on the performance of the algorithm is set. Figure 2 shows that the time complexity of all three algorithms is constant. The storage space mainly involves the need for temporary space to establish a probability distribution table. The Space complexity of the three algorithms is O (n), and usually n is a constant. For example, the complexity of the normal distribution is O (320).

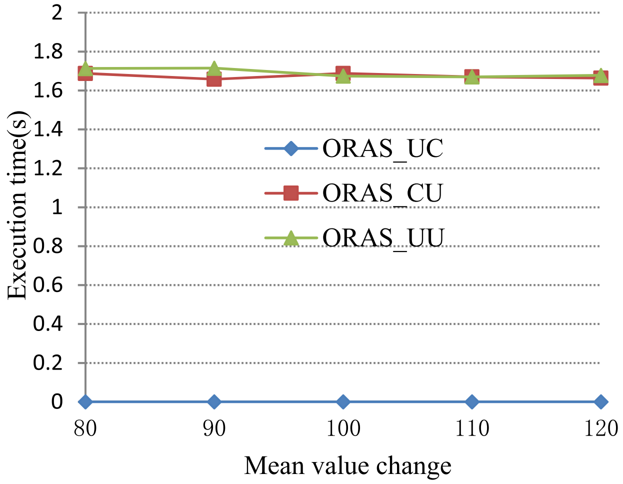


FIGURE 2. Time complexity analysis of the three algorithms.

B. NUMERICAL EXAMPLES

This section describes the calculation process of optimal resource allocation for the three algorithms using specific examples.

(1) **Basic parameters setting.** Our methods are suitable for scenarios where user access is random and cloud instance support capacity is random. We selected a literature reading service from a certain university as the experimental object, which runs on a shared cloud platform. We selected a literature reading service from a certain university as the experimental object, which runs on a public cloud platform. The service provider provides literature reading service to readers, service price Pu $=$ 4 ¥, the amount of virtual resources consumed per user accessing the service $R = 10$ CIPU, virtual resource price $P = 0.4$ ¥, IaaS resource unit rental cost $C = 10$ ¥, virtual resource unit out-of-stock cost $L = 0.3$ ¥; virtual resource unit resource idle cost $H = 0.1$ ¥; The amount of users obeys a normal distribution $u \sim N(5000, 500)$; The amount of virtual resources available per IaaS computing instance follows a normal distribution $y \sim N(100, 9)$. The basic parameters setting is shown in Table 3.

TABLE 3. Basic parameters.

R	$\mathrm{P_u}$	P	C	L	H	u	y
10	4	0.4	10	0.3	0.1	$N(5000{,}500)$	$N(100{,}9)$

(2) **ORAS_UC numerical example.** In this example, only the case of uncertainty in demand is considered, which is given by Eq. (3),

$$\int_0^Q \mathrm{f}(\mathrm{u})du = \frac{0.4 + 0.3 - 0.1}{0.4 + 0.3 + 0.1} = 0.75$$

A table of the normal distribution gives,

$$\int_0^{0.67} \mathrm{f}\,(\mathrm{u1})d\mathrm{u1} < 0.75 < \int_0^{0.68} \mathrm{f}\,(\mathrm{u2})du2$$

$\frac{\frac{Q}{10}-5000}{500} \in [0.67, 0.68]$, which is $Q \in [53350, 53400]$. As the supply is determined in this example, $Y = 100$. According to Eq. (4), the IaaS instance optimal allocation $X \in [533.5, 534.0]$ can be obtained $\mathrm{X} \in [533.5, 534.0]$, we can take the average value of 533.75 CIPU.

(3) **ORAS_CU numerical example.** In this example, only supply uncertainty is considered, and demand is determined, i.e., the number of user visits is 5000, then the virtual resource demand D1 $= D \times R = 50000$ is obtained. from Eq. (6), the following equation can be got

$$\int_0^{\frac{\mathrm{D1}}{X}} y \times \mathrm{f}\,(\mathrm{y})dy = (C+1) \big/ (P+L+H) = 13.75$$

The solution of the equation is calculated using the FFS_BS algorithm, and $X = 543.4783$ is obtained, the optimal allocation of IaaS resources under demand-determined and supply-uncertain is 543.4783 CIPU, and the iterative process of FFS_BS algorithm is shown in Figure 3.

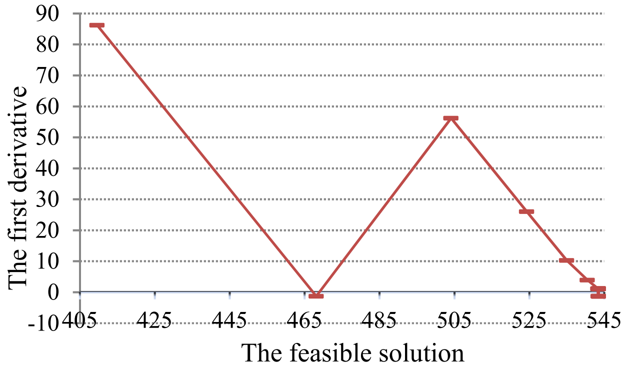


FIGURE 3. The iterative process of FFS_BS.

(4) **ORAS_UU numerical example.** In this example, the uncertainty of demand and the uncertainty of supply are considered. The uncertainty in the number of user accesses is considered first, and the demand for virtual resources $Q = 53375$ is calculated from Eq. (3). Then, considering the uncertainty of the number of virtual resources that IaaS computing instances can support, the optimal IaaS computing instance allocation $X = 580.1630$ CIPU is solved by Eq. (6) and FFS_BS.

C. SENSITIVITY ANALYSIS

This section evaluates the sensitivity of the output results of the three algorithms to input parameters by changing the values of service sales price, IaaS resource leasing cost, virtual resource out of stock cost, and virtual resource idle cost.

ORAS_UC parameters analysis: based on Table 3, the service sales price is varied from 0.2 to 0.6, the IaaS resource leasing cost is varied from 8 to 12, the out-of-stock cost of virtual resources varies from 0.1 to 0.5, the idle cost of virtual resources varies from 0.08 to 0.12, and the SaaS provider revenue and the optimal amount of resources allocated are shown in table 4 (1) and 4 (2). From the table, the sales price and out of stock cost have the greatest impact on the revenue of SaaS provider.

TABLE 4. (1) Sales price and leasing cost analysis for ORAS_UC. (2) out of stock cost and idle cost analysis for ORAS_UC.

(1)

Sales price changes				*Leasing cost changes*			
P	*Q*	*X*	G(Q)	*C*	*Q*	*X*	G(Q)
0.2	52150	521.5	3909.20	8	53800	538.0	14800.3
0.3	52800	528	8810.31	9	53550	535.5	14263.6
0.4	53350	533.5	13728.9	10	53350	533.5	13728.9
0.5	53800	538	18659.9	11	53200	532.0	13196.1
0.6	54200	542	23600.2	12	53000	530.0	12665.3

(2)

Sales price changes				*Leasing cost changes*			
L	*Q*	*X*	G(Q)	*H*	*Q*	*X*	G(Q)
0.1	52150	521.5	3909.20	0.08	53650	536.5	13813.5
0.2	52800	528	8810.31	0.09	53500	535.0	13770.6
0.3	53350	533.5	13728.9	0.10	53350	533.5	13728.9
0.4	53800	538	18659.9	0.11	53200	532.0	13688.2
0.5	54200	542	23600.2	0.12	53050	530.5	13648.6

ORAS_CU parameters analysis: based on the parameters in Table 3 and set $D = 5000$, the service sales price varies from 0.2 to 0.6, the IaaS resource leasing cost varies from 8 to 12, the out-of-stock cost of virtual resources varies from 0.1 to 0.5, the idle cost of virtual resources varies from 0.08 to 0.12, and the SaaS provider revenue and the optimal amount of resources allocated are shown in table 5 (1) and (2). From the table, the sales price and Leasing cost have the greatest impact on the revenue of SaaS provider.

TABLE 5. (1) sales price and leasing cost analysis for ORAS_CU. (2) out of stock cost and idle cost analysis for ORAS_CU.

(1)

Sales price changes				*Leasing cost changes*			
P	*Q*	*X*	G(Q)	*C*	*Q*	*X*	G(Q)
0.2	53763	537.6	3894.96	8	55556	555.6	14852.8
0.3	54348	543.5	8780.46	9	54945	549.5	14242.3
0.4	54945	549.5	13692.9	10	54945	549.5	13692.9
0.5	55556	555.6	18587.9	11	54347	543.5	13187.0
0.6	55556	555.6	23554.4	12	54347	543.5	12643.5

(2)

out-of-stock cost changes				*idle cost changes*			
L	*Q*	*X*	G(Q)	*H*	*Q*	*X*	G(Q)
0.1	53763	537.6	13895.0	0.08	54945	549.5	13799.7
0.2	54348	543.5	13780.5	0.09	54945	549.5	13746.3
0.3	54945	549.5	13692.9	0.10	54945	549.5	13692.9
0.4	55555	555.6	13587.9	0.11	54945	549.5	13639.5
0.5	55555	555.6	13554.4	0.12	54945	549.5	13586.0

ORAS_UU parameter analysis: based on Table 3, the service sales price varies from 0.2 to 0.6, the IaaS resource leasing cost varies from 8 to 12, the out-of-stock cost of virtual resources varies from 0.1 to 0.5, the idle cost of virtual resources varies from 0.08 to 0.12, and the SaaS provider revenue and the optimal amount of resources allocated are

TABLE 6. **(1) sales price and leasing cost analysis for ORAS_UU. (2) out of stock cost and idle cost analysis for ORAS_UU.**

(1)

Sales price changes				*Leasing cost changes*			
P	*Q*	*X*	G(Q)	*C*	*Q*	*X*	G(Q)
0.2	*52150*	*560.8*	*4062.44*	*8*	*53800*	*597.8*	*15852.0*
0.3	*52800*	*573.9*	*9272.16*	*9*	*53550*	*588.5*	*15253.5*
0.4	*53350*	*586.3*	*14610.3*	*10*	*53350*	*586.3*	*14610.3*
0.5	*53800*	*597.8*	*20000.5*	*11*	*53200*	*578.3*	*14030.9*
0.6	*54200*	*602.2*	*25532.9*	*12*	*53000*	*576.1*	*13402.1*

(2)

out-of-stock cost changes				*idle cost changes*			
L	*Q*	*X*	G(Q)	*H*	*Q*	*X*	G(Q)
0.1	*52150*	*560.8*	*14492.4*	*0.08*	*53650*	*589.6*	*14807.1*
0.2	*52800*	*573.9*	*14552.2*	*0.09*	*53500*	*587.9*	*14708.6*
0.3	*53350*	*586.3*	*14610.3*	*0.10*	*53350*	*586.3*	*14610.3*
0.4	*53800*	*597.8*	*14620.5*	*0.11*	*53200*	*584.6*	*14512.4*
0.5	*54200*	*602.2*	*14692.9*	*0.12*	*53050*	*583.0*	*14414.8*

shown in table 5 (1) and 5 (2). It can be seen from the table that the sales price has the greatest impact on the revenue of SaaS provider, followed by the lease cost.

D. ALGORITHM COMPARISON ANALYSIS

This section compares the algorithm proposed in this paper with existing algorithms in terms of revenue, QoS violation rate, and resource utilization.

Our methods are based on the perspective of SaaS providers, and the goal is to maximize SaaS revenue while not violating QoS constraints and high IaaS resource utilization. There are few research results on resource allocation based on the perspective of SaaS providers. In this section, SS_MaCM [27] (cloud model-based SaaS selection algorithm) and AFERM [25] (the application feature-based elastic resource manager) are selected to compare with the methods presented in this paper (the relevant introduction of the two algorithms can be found in Sec. II-B.). SS_MaCM considers the randomness of QoS, while AFERM considers the randomness of the number of users in the application. If the revenue of ORAS_UC are verified to be superior to the two algorithms mentioned above, and then the benefits of ORAS_UU algorithm are verified to be superior to ORAS_UC and ORAS_CU, it can be verified that the algorithm proposed in this paper maximizes the benefits of SaaS providers in a stochastic supply and demand environment.

Regardless of the impact of the service itself and dynamic environment on QoS violation rates, the best QoS accuracy predicted by SS_MaCM is used as a reference for QoS violation rates; SS_MaCM did not consider the impact of user access on the service, and set its IaaS resource configuration to meet the maximum user access required by QoS constraints. The actual resource usage is proportional to the expected user access. The AFERM approach only considers the effect of experiments on services with relatively good request rate patterns as a reference, as it weighs resource overhead and deployment flexibility. The overhead limit for virtual resources is set at 80%, and the amount of virtual resources remaining without reaching the overhead limit is not considered. In the ORAS_UC, the SaaS provider proposes to configure the amount of IaaS resources based on the probability distribution of user visits. From the perspective of the IaaS provider, there is no violating QoS constraints. Therefore, the QoS violation rate of the ORAS_UC is setting to 0. Figure 4 shows a comparison of QoS constraint violations and resource utilization. From Figure 4, ORAS_UC has a high resource utilization rate and a low QoS constraint violation rate.

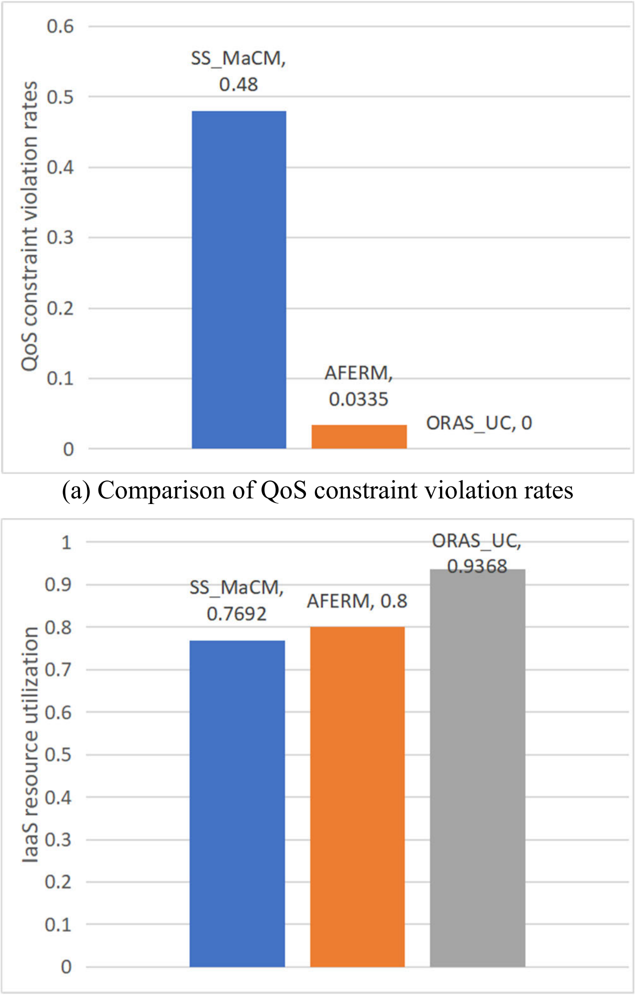
SS_MaCM does not consider the impact of user accesses on resource and service QoS, assuming that user accesses are mean values (i.e., $u = 5000$, $Q = 50000$, $X = 500$), the number of users accesses in the example parameters (see Table 3). Instead, AFERM considers the impact of user accesses on resource demand and service QoS, the basic idea being to satisfy all concurrent user requests and minimize violations of QoS constraints by taking the maximum number of user accesses (i.e., $u = 5000 + 3 \times 500 = 6500$, $Q = 65000$, $X = 650$). Then, the revenues of SaaS providers obtained from SS_MaCM and AFERM were calculated by Eq. (2). Figure 5 shows the revenue comparison, which considers only the revenue of a SaaS provider with uncertain demand. From Figure 5, ORAS_UC can achieve higher revenue for SaaS providers.

In stochastic environment of supply and demand (the parameters are shown in Table 3), we compare the revenue of our three algorithms. Figure 6 shows the revenue comparison, which considers the revenue of a SaaS provider with uncertain demand and supply. From Figure 6, it can be seen that ORAS_UU can achieve higher revenue for SaaS providers.

Comparison with existing algorithms: in response to the uncertainty of user access and IaaS load, the existing IaaS provisioning solutions based on IaaS and PaaS levels either predict user access and IaaS load for provisioning, or monitor user access and load changes for adaptive adjustment, or reserve IaaS resources to reduce the number of QoS constraint violations. The prediction method is challenging to guarantee the accuracy, and the monitoring and adaptive method cannot achieve timely response and often violates QoS constraints; the reserved IaaS method can reduce the frequency of QoS constraint violations but reduces the resource utilization of the cloud system and the cost of the SaaS provider. The algorithm proposed in this study obtains the optimal number of IaaS allocations that maximize the SaaS provider's benefit under the uncertainty of user access and virtual resource provisioning and no QoS constraint violation at the IaaS and PaaS levels.

E. SUMMARY

This section validates the advantages of our methods from the following four aspects. (1) Algorithms performance analysis proves that the time and space complexity of three algorithms are constant orders. (2) The numerical examples clearly describe the calculation process of the optimal resource allocation for the three algorithms, and indicate that their calculations are reasonable. (3) In sensitivity analysis,



(a) Comparison of QoS constraint violation rates

(b) Comparison of IaaS resource utilization

FIGURE 4. Comparison of QoS constraint violations and resource utilization.

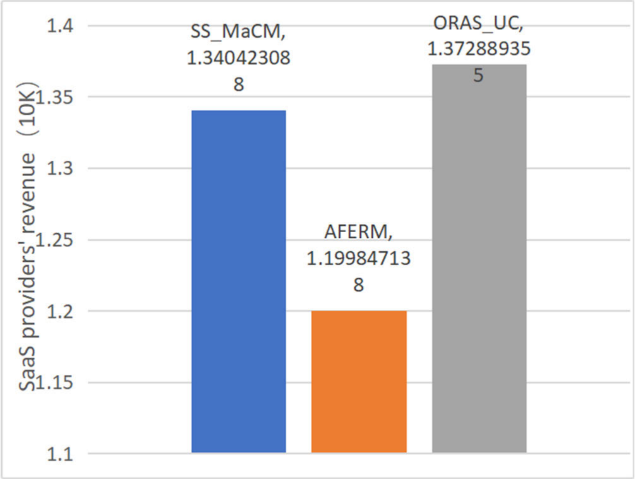


FIGURE 5. SS_MaCM, AFERM, ORAS_UC Revenue comparison.

it was found that ORAS_UC is more sensitive to sales prices and out of stock costs, ORAS_CU is more sensitive to sales prices and leasing costs, and ORAS_UU is more sensitive to sales prices. (4) In algorithm comparison analysis, it has been proven that our methods can maximize the revenue of SaaS providers, while having lower QoS violation rates and higher resource utilization.

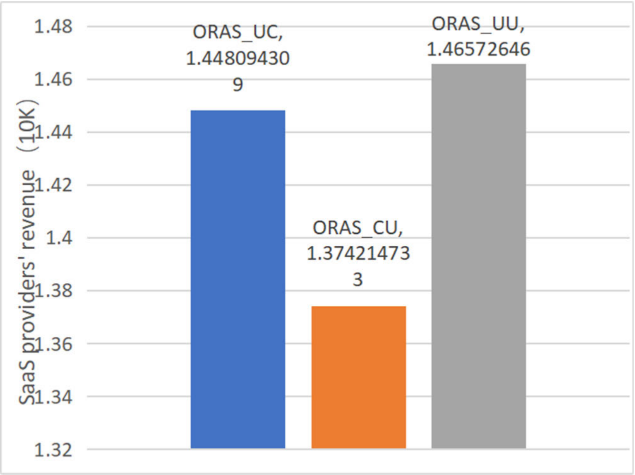


FIGURE 6. ORAS_UC, ORAS_CU, ORAS_UU Revenue comparison.

VII. DISCUSSION

Our method's adaptability: (1) The main input parameters involved in our method include service price, rental cost, out of stock cost, and idle cost. The service price can also be the service value, if it can be measured in currency. The cost parameters can be added or deleted according to actual scenarios, and the basic idea of the method remains unchanged. (2) For the two random variables of demand and supply, they can be either discrete or continuous. For the sake of intuitive calculation, the continuous type is used in the text. The distribution types of random variables can be binomial distribution, Poisson distribution, uniform distribution, exponential distribution, normal distribution, etc., but not limited to those. If the random variables with stable distribution tables, the optimal allocation of resources can be calculated by our methods. (3) The probability distribution table of virtual resource demand is obtained by calculating the probability distribution of user volume. It is necessary to establish a mapping relationship between user and virtual resource demand based on actual scenarios (which is common). In the paper, a constant is used to describe the virtual resource required by a user. Functions, constants, or other types of relationships are applicable to our methods.

The limitations of our methods: In addition to meeting the assumptions in Section C of Part 3, our method also has some limitations. (1) Being able to maximize the expected revenue of SaaS providers, cannot maximize the revenue of SaaS providers in every resource allocation, and cannot maximize the revenue for PaaS and IaaS providers. (2) Suitable for environments with random demand and supply, not suitable for deterministic environments. (3) Due to the lack of a dynamic adaptive mechanism, it is not suitable for scenarios where the distribution characteristics of random variables change dynamically. (4) Suitable for single resource configuration, not suitable for multi-resource joint configuration.

VIII. CONCLUSION

In the existing studies on IaaS resource allocation under service QoS constraints at the IaaS and PaaS levels, the problems of QoS constraint violation and low resource utilization under uncertainty in user access and virtual resource provisioning (i.e., IaaS resource load) are still highlighted, and the revenue of SaaS providers is not considered. Regarding the above difficulties, this study proposes a minimum optimal allocation strategy for IaaS resources that maximizes the expected revenue of SaaS providers and designs three minimum optimal allocation strategies for IaaS resources with uncertain demand and supply, demand and uncertain supply, and both uncertain demand and uncertain supply, to achieve the goal of maximizing the expected revenue of SaaS providers. The experimental analysis proves that the method in this study can effectively obtain the optimal allocation of IaaS resources and has high operational efficiency, with the following advantages: (1) it can effectively determine the minimum optimal allocation of IaaS resources in the three cases that maximize the expected revenue of the SaaS provider (2) the SaaS provider determines the IaaS allocation, and there is no violation of QoS constraints, which is also conducive to its accurate allocation of IaaS resources and improvement of system resource utilization; (3) it can effectively cope with the problem of complicated IaaS configuration caused by uncertainty in user access and IaaS resource load.

As this study's research on IaaS configuration for maximizing the revenue of SaaS providers is still in its initial stage, there are still many shortcomings that need further improvement. (1) There is still room for improvement in SaaS provider revenue; (2) How to ensure the QoS requirements of users who have been transferred services; (3) How to fully utilize idle resources to further enhance SaaS provider revenue; (4) How to allocate resources under multiple resource requirements; (5) How to allocate resources under different resource leasing price models; (6) How to allocate resources for multiple infrastructure providers; (7) How to establish an adaptive resource allocation model under stochastic supply and demand conditions.

APPENDIX A
SOURCE CODE OF ALGORITHMS

The algorithm source code proposed in this article can be downloaded from the following link: https://pan.baidu.com/s/1j4TbrD-DovZRJGuILbvFVg, and the extraction code is 1234.

ACKNOWLEDGMENT

The authors are grateful to the Senior Editor and three anonymous referees for their constructive comments and suggestions, which have helped them significantly improve the quality and exposition of this paper.

REFERENCES

[1] A. Zhou, S. Wang, Z. Zheng, C.-H. Hsu, M. R. Lyu, and F. Yang, "On cloud service reliability enhancement with optimal resource usage," *IEEE Trans. Cloud Comput.*, vol. 4, no. 4, pp. 452–466, Oct. 2016, doi: 10.1109/TCC.2014.2369421.

[2] Q. Zhu and G. Agrawal, "Resource provisioning with budget constraints for adaptive applications in cloud environments," *IEEE Trans. Services Comput.*, vol. 5, no. 4, pp. 497–511, 4th Quart., 2012, doi: 10.1109/TSC.2011.61.

[3] H. Shi, H. Xu, and X. Xu, "Service composition considering QoS fluctuations and anchoring cost," in *Proc. ICWS*, Chicago, IL, USA, 2021, pp. 370–380, doi: 10.1109/ICWS53863.2021.00056.

[4] C. Wu, A. N. Toosi, R. Buyya, and K. Ramamohanarao, "Hedonic pricing of cloud computing services," *IEEE Trans. Cloud Comput.*, vol. 9, no. 1, pp. 182–196, Jan. 2021, doi: 10.1109/TCC.2018.2858266.

[5] F. Alzhouri, A. Agarwal, and Y. Liu, "Maximizing cloud revenue using dynamic pricing of multiple class virtual machines," *IEEE Trans. Cloud Comput.*, vol. 9, no. 2, pp. 682–695, Apr. 2021, doi: 10.1109/TCC.2018.2878023.

[6] Q. Li, Q.-F. Hao, L.-M. Xiao, and Z.-J. Li, "Adaptive management and multi-objective optimization for virtual machine placement in cloud computing," *Chin. J. Comput.*, vol. 34, no. 12, pp. 2253–2264, Mar. 2012.

[7] D. Sun, G. Chang, and F. Li, "Optimizing multi-dimensional QoS cloud resource scheduling by immune clonal with preference," *Acta Electronica Sinica*, vol. 39, no. 8, pp. 1824–1831, Aug. 2011.

[8] S. K. Addya, A. Satpathy, B. C. Ghosh, S. Chakraborty, S. K. Ghosh, and S. K. Das, "CoMCLOUD: Virtual machine coalition for multi-tier applications over multi-cloud environments," *IEEE Trans. Cloud Comput.*, vol. 11, no. 1, pp. 956–970, Jan. 2023, doi: 10.1109/TCC.2021.3122445.

[9] A. Belgacem, K. Beghdad-Bey, and H. Nacer, "Dynamic resource allocation method based on symbiotic organism search algorithm in cloud computing," *IEEE Trans. Cloud Comput.*, vol. 10, no. 3, pp. 1714–1725, Jul. 2022, doi: 10.1109/TCC.2020.3002205.

[10] W. Guo, K. Q. Zhang, and L. Z. Cui, "A cloud resources placement method supporting SaaS applications with multi-dimensional and heterogeneous requirements," *Chin. J. Comput.*, vol. 41, no. 6, pp. 1225–1237, Jun. 2018.

[11] G. J. Kuang, G. S. Zeng, and J. Cao, "Satisfactory marriage method between cloud tasks and resources based on graph theory," *Acta Electronica Sinica*, vol. 42, no. 8, pp. 1582–1586, Aug. 2014.

[12] Y. W. Wu, H. Wu, and J. Ren, "Heuristic based resource provisioning approach for big data analytics in cloud environment," *J. Softw.*, vol. 31, no. 6, pp. 1860–1874, Jun. 2020.

[13] Y. M. Yan, B. Zhang, and J. Guo, "A reinforcement learning-based self-adaptation performance optimization approach for SBS cloud application," *Chin. J. Comput.*, vol. 40, no. 2, pp. 464–480, Feb. 2017.

[14] J. J. Sun, X. W. Wang, and C. X. Gao, "Resource allocation scheme based on neural network and group search optimization in cloud environment," *J. Softw.*, vol. 25, no. 8, pp. 1858–1873, Aug. 2014.

[15] P. Zhou, B. Yin, X. S. Qiu, S. Y. Guo, and L. M. Meng, "Service reliability oriented cloud resource scheduling method," *Acta Electronica Sinica*, vol. 47, no. 5, pp. 1036–1043, 2019.

[16] C. Liu, F. Tang, Y. Hu, K. Li, Z. Tang, and K. Li, "Distributed task migration optimization in MEC by extending multi-agent deep reinforcement learning approach," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 7, pp. 1603–1614, Jul. 2021, doi: 10.1109/TPDS.2020.3046737.

[17] M. Chen, S. Huang, X. Fu, X. Liu, and J. He, "Statistical model checking-based evaluation and optimization for cloud workflow resource allocation," *IEEE Trans. Cloud Comput.*, vol. 8, no. 2, pp. 443–458, Apr. 2020, doi: 10.1109/TCC.2016.2586067.

[18] L. Yu, Y. Xie, and B. Chen, "Towards runtime dynamic provision of virtual resources using feedforward and feedback control," *J. Comput. Res. Develop.*, vol. 52, no. 4, pp. 889–897, Apr. 2015.

[19] Y. Ying, W. Cuirong, and W. Cong, "An uncompleted information game based resources allocation model for cloud computing," *J. Comput. Res. Develop.*, vol. 53, no. 6, pp. 1342–1351, 2016.

[20] Y. Wang, J.-T. Zhou, and X. Song, "A utility game driven QoS optimization for cloud services," *IEEE Trans. Services Comput.*, vol. 15, no. 5, pp. 2591–2603, Sep. 2022, doi: 10.1109/TSC.2021.3062383.

[21] V. Nallur and R. Bahsoon, "A decentralized self-adaptation mechanism for service-based applications in the cloud," *IEEE Trans. Softw. Eng.*, vol. 39, no. 5, pp. 591–612, May 2013, doi: 10.1109/TSE.2012.53.

[22] P. Haratian, F. Safi-Esfahani, L. Salimian, and A. Nabiollahi, "An adaptive and fuzzy resource management approach in cloud computing," *IEEE Trans. Cloud Comput.*, vol. 7, no. 4, pp. 907–920, Oct. 2019, doi: 10.1109/TCC.2017.2735406.

[23] A. Alsarhan, A. Itradat, A. Y. Al-Dubai, A. Y. Zomaya, and G. Min, "Adaptive resource allocation and provisioning in multi-service cloud environments," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 1, pp. 31–42, Jan. 2018, doi: 10.1109/TPDS.2017.2748578.

[24] Z. A. Wu, J. Z. Luo, and A. B. Song, "Data centers spanning dynamic resource co-reservation," *Chin. J. Comput.*, vol. 37, no. 11, pp. 2395–2407, Nov. 2014.

[25] H. Wei, S. R. Zhou, and R. Zhang, "Application feature based elastic resource management mechanism on PaaS," *Chin. J. Comput.*, vol. 39, no. 2, pp. 223–236, Feb. 2016.

[26] L. Qi, W. Dou, C. Hu, Y. Zhou, and J. Yu, "A context-aware service evaluation approach over big data for cloud applications," *IEEE Trans. Cloud Comput.*, vol. 8, no. 2, pp. 338–348, Apr. 2020, doi: 10.1109/TCC.2015.2511764.

[27] H. Ma, H. Zhu, K. Li, and W. Tang, "Collaborative optimization of service composition for data-intensive applications in a hybrid cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 5, pp. 1022–1035, May 2019, doi: 10.1109/TPDS.2018.2879603.

[28] Alibaba Cloud. *Alibaba Cloud Infrastructure Supply Chain Inventory Management Decision Dataset*. Accessed: Oct. 8, 2022. [Online]. Available: https://tianchi.aliyun.com/dataset/138679?t=1676603960552

[29] Z. Zhou and F. M. Liu, "An energy-efficient incentive mechanism for resource recycling in multi-tenant datacenters," *Scientia Sinica Informationis*, vol. 51, no. 5, pp. 735–749, May 2021.

LONGCHANG ZHANG was born in Xiuyan, Liaoning, China, in 1977. He received the B.S. degree in computer science and technology from Bohai University, Jinzhou, China, in 2000, and the M.S. and Ph.D. degrees in computer science and technology from the Beijing University of Posts and Telecommunications, Beijing, in 2008 and 2011, respectively.

From 2011 to 2012, he was a Lecturer with the College of Information Science and Technology, Bohai University. From 2013 to 2018, he was an Associate Professor with the College of Information Science and Technology, Bohai University. From 2016 to 2017, he was a Visiting Associate Professor with the Department of Computer Science, The University of Hong Kong. From 2019 to 2022, he was a Professor with the College of Information Science and Technology, Bohai University. Since 2022, he has been a Professor with the Shenzhen Research Institute, Beijing University of Posts and Telecommunications, Shenzhen, China. Since 2023, he has been a Professor with School of Information Engineering, Suqian University, Suqian, China. He is the author of two books, more than 60 articles, and more than ten inventions. His research interests include service computing, cloud computing, and intelligent manufacturing.

JING BAI was born in 1987. She received the master's degree in business administration from the Beijing University of Posts and Telecommunications, in 2020. She is currently pursuing the Ph.D. degree with the School of Management Science and Engineering, Dongbei University of Finance and Economics, China. Her paper has appeared in *Computer Integrated Manufacturing Systems*. Her current research interests include cloud computing and inventory optimization.

JIANJUN XU (Member, IEEE) received the Ph.D. degree in operations management from Nanyang Technological University. He is currently a Professor with the International Business College and the Institute of Supply Chain Analytics, Dongbei University of Finance and Economics, China. His papers have appeared in *Operations Research*, *Production and Operations Management*, *PNAS*, *IISE Transactions*, *European Journal of Operational Research*, *International Journal of Production Economics*, *Naval Research Logistics*, and *Operations Research Letters*. His research interests include dynamic inventory control, stochastic optimization, and data-driven operations management.