A Blockchain Platform of Crowdsensing for Cloud Reallocation

Yisong Chen[1, *], Jin Li[2]

[1] College of Computing, Georgia Institute of Technology, Atlanta, US 30332
[2]School of Computing and Artificial Intelligence, Beijing Technology and Business University, Beijing, China 102401
[*]Email: yisongchen97@gmail.com

Abstract

This paper proposes a decentralized blockchain-based crowdsensing platform for dynamic cloud resource reallocation and pricing. By integrating supervised linear regression with auction theory, the system enables fair and efficient cloud trading based on user reputation values. In this model, secondary users (buyers) bid for idle cloud resources offered by primary users (sellers), while crowd sensors detect resource availability and feed real-time data into the blockchain. A smart contract-driven incentive mechanism ensures high-quality data collection and trustworthy transactions. The proposed model employs supervised learning to classify and allocate resources, using a modified Vickrey-Clarke-Groves (VCG) pricing mechanism based on critical value theory. Experimental results demonstrate the model's high accuracy, fairness, and resource utilization. It has potential to optimize cloud allocation while performing economic efficiency and algorithmic truthfulness.

Keywords: Cloud allocation, blockchain, auction theory, smart contract

Article History:
Received November 20, 2024
Revised January 25, 2025
Accepted March 20, 2025
Available Online March 30, 2025

 https://doi.org/10.63646/JZEI5817

A Blockchain Platform of Crowdsensing for Cloud Reallocation

1. Introduction

With the development of information communication technologies, cloud computing and its relevant functions are becoming quickly integrated and are increasingly being applied in business and daily activities. As a shared and configurable computing resource, cloud computing is becoming an inevitable process of the decision-making system and it plays important roles in business (Gupta, Seetharaman, &Raj, 2013; Hsu, Ray, & Li-Hsieh, 2014). Considering cloud allocation and pricing jointly together is a challenging task in the extant study, since there is fixed pricing, dynamic pricing, game theory-based auction design, and machine learning-based mechanisms. All of the different cloud pricing strategies have common goals to distribute the available cloud computing efficiently and fairly, as well as to profit participants effectively and optimally.

As a commercial utility, cloud computing has its own characteristics associated with its value. Specifically, the intrinsic and the extrinsic features are the two components that jointly determine cloud pricing. As a computing and internet relevant service, cloud computing is becoming more and more reliant on extrinsic features (Bayramusta & Nasir, 2016). The leading cloud companies, such as Amazon and Microsoft, have introduced their cloud services with different pricing strategies and saving portfolios; however, it can still be difficult for buyers to recognize what factors impact cloud value and how to precisely estimate that value. This study adopts a classical auction mechanism (VCG) in which both sellers and buyers bid, based on reputation value, which is an important indicator in the cloud pricing model. Our design illustrates an alternative approach that allows participants to pay attention to a unique indicator (e.g., reputation value) instead of to the many varied features of cloud computing.

Depending upon each participant's reputation value, the study proposes a viable and efficient blockchain-based cloud reallocation and pricing system that is adaptable to the optimization of cloud resources. The buyers (secondary users) will compete for cloud resources based on their own reputation values, and the sellers (primary users) will seek buyers with higher reputation values and will guarantee buyers the winning resources. In the blockchain trading system, the miners (crowd sensors) will take the responsibility of sensing the potential available cloud resources as well as the usage status of the primary user's cloud resources. All useful information will be sent, verified, and posted in the block. Then, the buyers and sellers will have a chance to complete a transaction (Samimi, Teimouri & Mukhtar, 2016).

The supervised linear regression algorithm is used to classify and to analyze the information from crowdsensing to reallocate cloud resources based on the reputation value of buyer. It is used to study and to find an optimal model from the training dataset, to reallocate cloud resources. Under critical value theory, the pricing algorithm ensures the truthfulness of the auction mechanism. From the perspectives of social welfare, allocation accuracy, execution time, and resource utilization, the proposed reallocation and pricing mechanisms offer strong performance (Lehmann, Oćallaghan, & Shoham, 2002).

See: https://inatgi.in/index.php/jtis/index for more information. https://doi.org/10.63646/JZEI5817

The proposed blockchain-based cloud trading system has the potential to improve both economic efficiency and optimal profits. Also, the blockchain system can set up a relatively free trading environment without unnecessary intervention. The integration of blockchain and cloud computing is a good example of a blockchain scenario in managerial decisions. Blockchain has the potential to build a reputation-based pricing system for cloud and to provide a decentralized trading network for cloud transactions. In addition, crowdsensing is embedded into the system to locate available cloud resources for potential trading between sellers and buyers.

This paper is structured as follows: Section II provides a literature review of the extant studies regarding cloud allocation and pricing. Section III depicts the decentralized cloud trading platform. Section IV proposes a machine learning mechanism for cloud pricing: the supervised linear regression. Section V illustrates the pricing algorithm. Section VI addresses and verifies the procedures for training and testing the supervised linear regression. Section VII discusses several potentials for future research. The conclusion is found in Section VIII.

2. The Extant Study of Cloud Allocation and Pricing

2.1 Fixed Pricing Mechanisms

Fixed pricing strategies currently are the major mechanisms that cloud providers are using to conduct cloud business. Two fixed pricing strategies are pay-as-you-go and subscription (Li et al., 2016). Pay-as-you-go allows users to pay for what they have used, without promotions. Subscription, another type of fixed pricing, requires users to pay the service fees in advance with discounts (Al-Roomi et al. 2013; Luong et al. 2017). For example, Amazon AWS employs “On-Demand”, “Saving Plans”, and “Dedicated Hosts.” Microsoft Azure uses “Pay as you go” and “Reserved Virtual Machine Instances.” Prices are stable and are easy to follow in the fixed pricing strategies; however, the price cannot reflect the relationship between supply and demand, under a given circumstance. Both sellers and buyers have the potential for loss, with fixed pricing. Dynamic pricing strategies can resolve this problem, with varied pricing, according to the fluctuation in the markets.

2.2 Dynamic Pricing Mechanism

No-auction Pricing Mechanism

The dynamic pricing algorithm can estimate the price based on the market status and can present a reasonable price, both for cloud providers and for customers. Models from other disciplines can also be employed in the pricing of cloud resources. One study (Sharma et al., 2015) presents fuzzy and genetic algorithms to keep the cloud price in a certain range, based on Financial Option Theory and on Moore’s Law. This approach reflects the impacts of the start time of the resource, the QoS (quality of service), the rate of depreciation and inflation, and capital investment. By applying the Lyapunov Optimization Model, He et al. (2013) addressed the problem of the revenue maximization of cloud allocation. This study is a good attempt to balance between the operational costs and the QoE (Quality of Experience). Other studies have proposed the Genetic Model (Macias & Guitart, 2011), or the Markov Decision Process (Truong-Huu & Tham, 2014), for pricing in cloud computing markets.

 https://doi.org/10.63646/JZEI5817

When considering heuristic resource allocation based on greedy algorithms as the major algorithm (Zaman & Grosu, 2013), an approximate algorithm can solve the issue of resource allocation, but it can also bring computational inefficiency and allocation inaccuracy. Although the heuristic algorithm based on the greedy approach can satisfy monotonicity, it cannot achieve good results in solving the problem of multi-resource allocation. Compared to the optimal solution, its profits, accuracy, and utilization are lower. Among other items, user evaluation and operation cost are critical items to consider when thinking of cloud pricing. Zhang et al. (2018) proposed a heuristic and truthful auction mechanism for the allocation and pricing of cloud computing resources that is based upon these two features. Studies (Chatterjee, Ladia, & Misra, 2015 and Do et al., 2016) employed the dynamic optional scheme to resolve provision and pricing from the perspective of different services, e.g., hardware; both cloud service providers and customers were satisfied to some extent. Another study (Jin et al., 2014) used an optimized fine-grained algorithm to leverage the potential costs of cloud operation.

Auction Pricing Mechanism

Basically, in an auction pricing mechanism, all of the participants know the nature of the available resources and the distribution of these resources. Each participant, i.e., a provider or a customer, needs a bidding strategy (Milgrom & Weber, 1982). Only when the resource allocation is an optimal solution and the allocation algorithm satisfies the monotonicity can the auction mechanism be guaranteed to be reliable (Lehmann, Oćallaghan, & Shoham, 2002). In an auction design, it should benefit at least one of the two parties, the customer and/or the provider. In terms of optimal allocation strategy, one study (Nejad, Mashayekhy, & Grosu, 2014) used integer programming, and another one (Mashayekhy, Fisher, & Grosu, 2015) employed dynamic programming to solve the optimal solution for cloud allocation. In another study, a combinatorial auction was treated as a decision-making problem for the winning bidder and was solved by the maximum group model (Wu & Hao, 2016).

As the scale of resources and the quantity of requests increase, the time will increase exponentially to calculate the optimal results. Hence, when the allocation scale is large, the polynomial time approximation scheme (PTAS) or the heuristic algorithm is the more effective approach to use in dealing with cloud allocation. Furthermore, Mashayekhy, Nejad, & Grosu (2014) proposed a PTAS algorithm for multi-task scheduling. Shi et al. (2015) transformed the online auction into a continuous static multi-round static resource allocation. Liu, Li, & Zhang, (2017) focused on heterogeneous physical machines that solve resource management, and designed multi-dimensional and multi-mapping mechanisms based on combinatorial auction and corresponding high-efficiency algorithms to obtain approximate solutions.

To achieve computational efficiency, bidding truthfulness, and competing fairness, other researchers (Wang, Ren, & Meng, 2012) employed an auction-style pricing model, which helped providers lift the overall revenue through cloud trading. A combinatorial auction was designed to dynamically provision multiple cloud resources to achieve social welfare approximation and bidding truthfulness (Shi et al., 2015; Samimi, Teimouri & Mukhtar, 2016).

 https://doi.org/10.63646/JZEI5817

3. The Blockchain-based Cloud Trading Platform

3.1 Incentive Mechanism in Crowdsensing

Crowdsensing & Incentive Mechanism

Crowdsensing refers to the formation of an interactive and participatory sensing network through the use of devices such as smart phones, laptops, intelligent wearable devices, and the release of information to the participants. Crowdsensing has the potential to achieve data collection, information analysis, and resource sharing. As a new mode of sensing the environment, crowdsensing is an important channel that participants can use to obtain data and services (Ganti, Ye, & Lei, 2011). An incentive mechanism is a means to realize crowdsensing by designing incentive methods to motivate and attract participants to join in sensing tasks. Crowd sensors will get rewards if they can provide high-quality and reliable information. More importantly, incentive mechanisms will be adopted to solve the problems faced by providers and participants, respectively, in maximizing the utility. It is necessary to increase the level of participation and to ensure that information perceived by participants is of high quality and reliability (Yang et al., 2015).

The cloud server seeks to recruit more sensors at the lowest or most reasonable cost. The incentive mechanism of the server platform should be able to motivate participants to provide high-quality data, not just to consider the cost of payment. Participants may falsify data or personal information to obtain more returns. One applicable way to resolve this issue is by conducting an incentive mechanism to stimulate participants based on their reputation values. Participants can process the auction without unnecessary intervention in a manner of decentralization and privacy, based on their own reputation rankings. The incentive mechanism will offer participants more rewards to trade cloud resources via the proposed auction platform. In this study, the reallocation of the sensing cloud is determined by the server. The server will formulate an appropriate reputation value according to the difficulty of a task to ensure the completion of that task. In addition, once the crowd sensor completes the sensing task, the reward will be automatically released through the smart contract. This mechanism will effectively protect the rights and interests of the crowd sensor (Capponi, et al., 2019).

3.2 The Procedure of Cloud Sensing

In this study, the blockchain platform is a consortium system that consists of a primary user (PU, the seller), a secondary user (SU, the buyer), the cloud server (S), and the cloud sensor (CS). The distribution of perception tasks is handled by S. S will formulate an appropriate reputation value, according to the difficulty of the task, to ensure the completion of the task.

The blockchain-based cloud sensing system consists of three layers: the physical layer, the transmission layer, and the application layer. The physical layer is the bottom layer of the system where the CS and the PU are. The PU has the priority of using cloud resources, and the CS is used to detect the cloud usage status of PU and to update the information to the transmission layer. The transmission layer is in the middle, and it connects the physical and application layers. The blockchain network is in this layer. Any transaction can happen after the verification of the original sensing data and the transaction to be added to the block. The application layer is the top layer of the system, including the SU and the S. The

 https://doi.org/10.63646/JZEI5817

SU is the potential buyer who seeks the PU's idle cloud resources, while the S processes the SU's cloud request, if it is verified by the system.

(1) Multiple SUs sends requests to the S. The winning SU needs to pay the S service fees and the PU access fees. The S's service fees include the mining fee, the sensing fee, and the operating fee.

(2) The S formulates the corresponding cloud sensing tasks, according to the SU's request and reputation value. The higher the reputation value, the more difficult the task, and the more rewards that the CS can obtain.

(3) The S publishes the sensing tasks in the form of smart contracts. The S specifies the task reputation value and the corresponding reward of each SU. A smart contract is an unmodifiable code that runs on the blockchain.

(4) CS provides quotes for different sensing tasks. The higher the sensing reputation value of the crowd sensor, the more sensing tasks it can perform. Assuming that the reputation value of a CS is R_{CS}, the reputation values of the crowd sensor for different sensing tasks should satisfy $\sum_{i=1}^{n} R_{T_i} \leq R_{CS}$.

(5) The S selects a certain number of CSs to complete the sensing tasks and broadcasts the selection results to all of the CSs in the system. The result is packaged by the miner and added to the blockchain. For any SU sensing task with a reputation value of R_{SU}, in order to complete the task, the sum of all quotations of all sensing tasks should satisfy $\sum_{j=1}^{m} R_{T_j^{cs}} \geq R_{SU}$. If the sum is $\sum_{j=1}^{m} R_{T_j^{cs}} < R_{SU}$, the task cannot be completed.

(6) The CS completes the process of the sensing cloud and transmits the unprocessed data to the miners. The miners receive sensing data from the CSs and process the data that can be used by S. The sensing result of each CS is recorded on the blockchain.

(7) CS receives the task rewards from S, and SU can access the available cloud services. Once the sensing data of the CS is verified, a smart contract will automatically reward the CS and will update the reputation value of the CS. After receiving the message that the task has been closed, the transaction will be completed.

4. The Supervised Linear Regression Algorithms

Aiming to distribute the sensing rewards fairly, the rewards to each CS are based on the reputation value; a CS will obtain more rewards if the sensor works more. This will effectively stimulate CS's willingness to join in cloud sensing.

4.1 Hedonic Regression Model of Cloud Pricing

The pricing metrics include the reputation, the cloud features, and the time of the same item from different customers' bids. The proposed hedonic regression model is:

$$P_{it} = f\,(R_{it}, C_{it}, T_t) \tag{1}$$

Where, P_{it} is price, R_{it} is the reputation of a certain cloud service, C_{it} is cloud feature attributes, T_{it} is time trend, *i* represents a certain cloud service, *t* represents a specific time.

See: https://inatgi.in/index.php/jtis/index for more information. https://doi.org/10.63646/JZEI5817

4.2 The Allocation Model of Cloud

Assume that there are m users in the set U. U = {1, 2, …, m}, user $i \in$ U. User i proposes resources requests $K_i^{k_r}$. k_r is a certain cloud resource, r = 1, 2, …, n. User i's resource request $K_i^{k_r} = (k_r, R_r^i)$, R_r^i is the reputation value of user i on resource k_r.

Supervised Auction Design

A complete auction design includes two components: resource allocation and price estimation. A reliable auction fulfills two requirements: truthfulness and accuracy. Truthfulness means that users cannot obtain benefits by falsified bid, and accuracy means that the allocation strategy should be the optimal solution, or at least very close to the optimal solution (Milgrom & Weber, 1982). The resource allocation in cloud computing is an NP-hard problem. If possible, an algorithm will be used to obtain the optimal solution; alternatively, an approximate or a heuristic algorithm will be used as a feasible solution to resolve the issue of the NP-hard problem. In this section, cloud reallocation and pricing will be introduced and explained in detail.

Definition 1. Monotonicity

If the request submitted by a user (B_r^i) can be allocated, any other request ($B_r^{i'}$) from the same user will be allocated on the condition of $B_r^{i'} > B_r^i$. This is the monotonicity of resource allocation.

Definition 2. Critical Value

If the request submitted by a user is allocated, there exists a critical value (CV_r). If the user bid $B_r^i > CV_r$, the request can be satisfied; otherwise, it cannot be satisfied.

Lemma 1.

If the resource allocation in the auction mechanism satisfies the monotonicity, and the final price satisfies the critical value, the mechanism is truthful.

The VCG auction mechanism, based on the optimal allocation solution, is truthful. But the final price using VCG cannot be calculated within polynomial time. This study employs supervised learning classification and regression to design cloud allocation. The basic principle is to select some requests from all user requests, and to estimate the optimal allocation and price. By fitting the optimal strategy, the training model is applied to all users to predict resource allocation. In this section, we design resource allocation based on linear regression algorithm (LN) and we construct the price algorithm based on the critical value theory.

The LN Algorithm of Cloud Allocation

In the auction design, the hypothesis function ($h_\theta(k^i)$) is constructed according to the SU's request for different resources.

$$h_\theta(k^i) = \theta_0 + \theta_1 k_1^i + \theta_2 k_2^i + \cdots + \theta_n k_n^i + \theta_{n+1}\sqrt{k_1^i} + \theta_{n+2}\sqrt{k_2^i} + \cdots + \theta_{2n}\sqrt{k_n^i} \tag{2}$$

The goal of the supervised linear regression is to find the rules that SU wins, it is $\theta_{LN} =$

See: https://inatgi.in/index.php/jtis/index for more information. https://doi.org/10.63646/JZEI5817

$(\theta_0\theta_1 \ldots \theta_{2n}) \in \mathbb{R}^{2n+1}$. The optimal strategy and the final price of each winning SU will be calculated. p_F^i is the final price from SU_i. If SU_i wins, $p_F^i > 0$, otherwise, $p_F^i = 0$. According to the optimal solution and all SUs' requests, the matrix of SU request is K = $[k^1, k^2, \ldots, k^m]^T$. The vector of the optimal allocation is X = $(x^1, x^2, \ldots, x^m)^T$. The vector the SUs' bidding is B = $(b^1, b^2, \ldots, b^m)^T$. The vector of the final price is P = $(p^1, p^2, \ldots, p^m)^T$. We have F (θ),

$$\mathrm{F}(\theta) = \frac{1}{2m}[\textstyle\sum_{i=0}^{m} x^i \left(h_\theta\left(k^i\right) - p^i\right)^2 + \lambda \sum_{i=0}^{m} \theta_j^2] \tag{3}$$

To get Min $F(\theta)$, θ can be solved by the normal equation based on the above function. Hence,

$$\theta = (K^T K - \lambda L)^{-1} K^T P \tag{4}$$

$$\mathrm{L} = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}$$

Then, we use the Sigmoid Function to estimate whether an SU's bid wins or not. It is,

$$P_i^W = \frac{1}{1+e^{-(b^i - h_\theta(k^i))}}, W^i \in (0,1) \tag{5}$$

P_i^W is the probability of an SU winning the bid. If an SU wants to win the bid, $b^i > h_\theta\left(k^i\right)$. Hence, $W^i \geq 0.5$ means that SU has a strong chance of winning the resources being bidded for.

5. The Pricing Algorithms

5.1 The Modified VCG Pricing Algorithm

A reliable auction mechanism will ensure that the price paid by buyers is optimal. The final price proposed in this study is based on the critical value. The estimated winning set W^i is used to calculate the final price. Specifically, $p_F^{i^+}$ is the maximum and $p_F^{i^-}$ is the minimum. When $p_F^{i^+} - p_F^{i^-} > \delta$, the final price that SU needs to pay is $p_F^{i^+}$.

5.2 Proof of Truthfulness

Lemma 2.

The supervised linear regression is monotonic.

Proof.

Suppose P_L^W is the probability that the last SU can be satisfied for the expected request. Based on the estimation function (Last Function), we have

$$P_i^W = \frac{1}{1+e^{-(b^i - h_\theta(k^i))}} > P_L^W \tag{6}$$

So,

$$b^i > h_\theta\left(k^i\right) - \ln[(1 - P_L^W)/P_L^W] \tag{7}$$

See: https://inatgi.in/index.php/jtis/index for more information. https://doi.org/10.63646/JZEI5817

This indicates that SU will win the bidding if the SU's bid satisfies the above function. If an SU's bid is greater than b^i, the SU can be guaranteed to obtain the requested cloud resources. Definition 1 presents the identical tone of monotonicity.

Lemma 3.

The final price algorithm satisfies the theory of critical value.

Proof.

When $b^i > p_F^{i+}$, an SU wins the bid; when $b^i < p_F^{i-}$, an SU will lose the bid.

There exists a critical value (Definition 2). Thus, the final price that the SU needs to pay is p_F^{i+}, if $p_F^{i+} - p_F^{i-} > \delta$.

Theorem1.

The cloud allocation design proposed is truthful.

Proof.

According to Lemma 1, the supervised linear regression algorithm satisfies resource allocation monotonicity.

Also, the final price algorithm satisfies the theory of critical value.

Therefore, the proposed cloud allocation design is truthful.

6. The Procedure of Training and Testing

6.1 Resource Allocation Algorithm Training

In training and testing, an open-source data set DAS-2 [1] was used as test data to simulate the SUs' requests. The configuration of the experimental platform is CPU Intel CoreI7 6500U, 16 GB memory, 1 TB DDR storage. The experimental conditions are: (1) For each valid record, CPU, memory, and storage information are used to simulate SU's requests; (2) For any request, an integer value from 1 to 100 is randomly generated and used to simulate an SU's bidding and to preset the reputation value of each resource; (3) The optimal allocation plan is solved by IBM CPLEX; (4) The optimal payment price based on the VCG mechanism is solved; (5) The implementation of the algorithms is programmed.

The test selected 5000 records as SU requests and generated corresponding bids. We calculated the resource density of each SU (d_i):

$$d_i = \frac{b_i}{\sqrt{\sum_{r=1}^{n}(\frac{1}{c_r} * k_r^i)}}, \forall i = 1,2,\ldots,m \tag{8}$$

According to the resource density, the SU requests are sorted in descending order, to form a total sample. The systematic sampling method is to extract 500 samples each time as a sample set; a total of 20 sample sets were set up. Among them, 17 samples were used as the training sets, and 3 samples were used as the cross-validation sets. This test substitutes the prediction model output by each training set into the cross-validation set for verification. Finally, the best model is selected as the final prediction model, which is used to estimate all SU requests. The prediction accuracy rate (PA) and the prediction

[1] Grid Workloads Archives [OL]. http://gwa.ewi.tudelft. nl,2018.2.

See: https://inatgi.in/index.php/jtis/index for more information. https://doi.org/10.63646/JZEI5817

error rate (PE) are applied to evaluate the model, PA + PE = 1. PA is defined as the number of SUs who have the same feasible and optimal solution divided by the total number of SUs:

$$P_i^W = \begin{cases} 1, & P_i^W \geq V_L \\ 0, & P_i^W < V_L \end{cases}$$

$$PA = \frac{1}{m}\sum_{i=1}^{m}(P_i^W = x_i) \tag{9}$$

V_L represents the predicted value of the last allocated SU, P_i^W represents the probability of a SU winning in the resource reallocation. The greater the value P_i^W, the higher the probability that a SU wins the bid. To solve θ in LN, the coefficient λ needs to be adjusted appropriately to ensure higher prediction accuracy in the cross-validation set. Figure 1 shows the change in the prediction error rate with λ when fitting the model in the training set to the cross-validation set. For LN, when λ=3, the prediction error rate is the smallest at 1.4% (Figure 1(a)). Similarly, for LG, when λ= {30, 40}, the prediction error rate is the smallest at 3.2%, (Figure 1(b)).

 https://doi.org/10.63646/JZEI5817

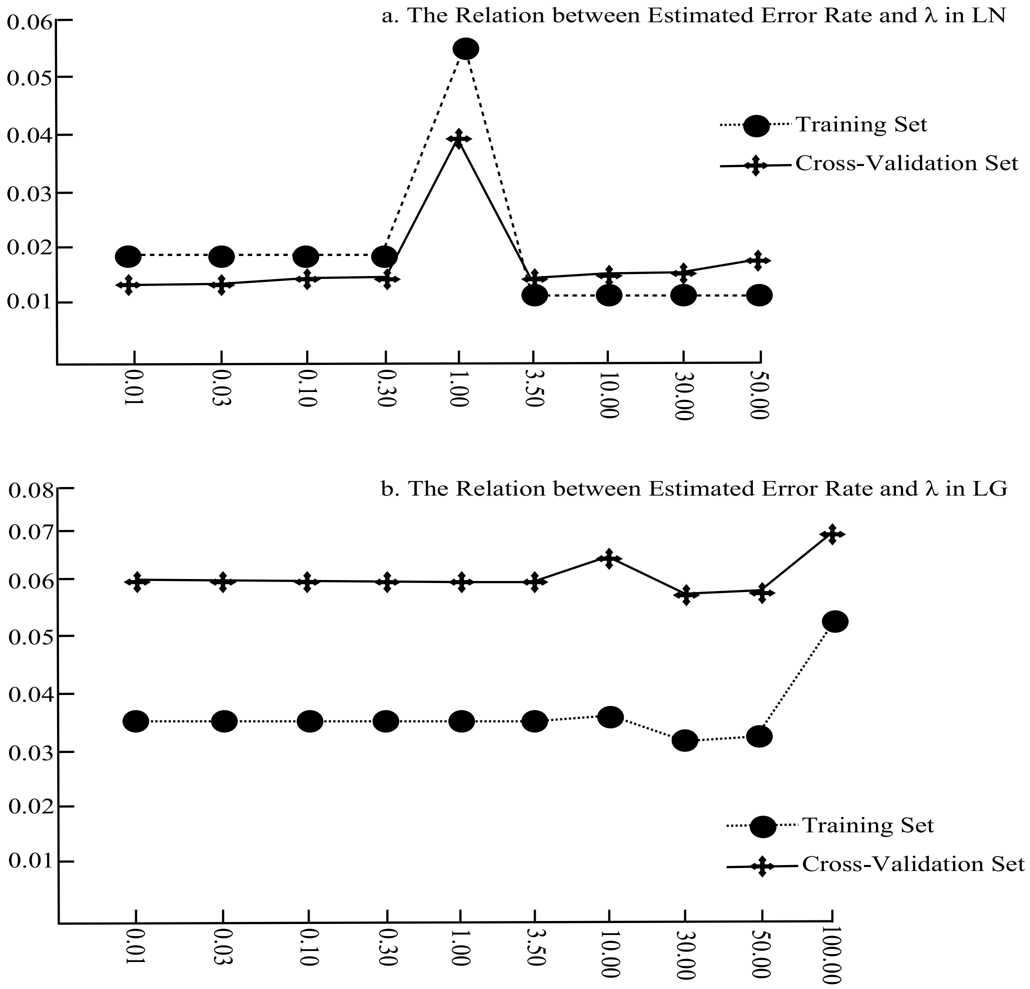

Figure 1. Comparison of Estimated Error Rate between LN and LG

6.2 Comparison of training Time among the Three Algorithms

The Algorithm of Logistic Regression (LG)

Different from LN, LG (Logistic Regression) does not calculate the final price, which is an important factor of the hypothesis function ($h_\theta(k^i)$) that is constructed according to the SU's request for different resources. $\theta_{LG} = (\theta_0\theta_1 \dots \theta_{n+1}) \in \mathbb{R}^{n+2}$, the conditions are:

$$f_\theta(k^i) = \theta_0 + \theta_1 k_1^i + \theta_2 k_2^i + \cdots + \theta_n k_n^i + \theta_{n+1}(b^i)^2$$

$$g(z) = \frac{1}{1+e^{-z}} \tag{10}$$

$$h_\theta(k^i) = g(f_\theta(k^i))$$

The function F (θ) is,

$$\mathrm{F}(\theta) = \frac{1}{m}\sum_{i=0}^{m}[-x^i \lg\left(h_\theta\left(k^i\right)\right) - (1 - x^i)\lg\left(1 - h_\theta\left(k^i\right)\right)] + \frac{\lambda}{2m}\sum_{i=0}^{n}\theta_j^2 \quad (11)$$

Then, the estimated winning probability is,

$$P_i^W = \frac{1}{1+e^{-(h_\theta(k^i))}}, W^i \in (0,1) \quad (12)$$

The Algorithm of Support Vector Machine (SVM)

When the resource capacity is small and the number of users is large, there are only a few users who can be selected and who can obtain the expected resources. The issue of skewness will appear in the training set. If the resource allocation algorithm of linear regression or logistic regression is still used for training, it will affect the prediction accuracy of the resource reallocation. Thus, the alternative algorithm used is SVM (Support Vector Machine). The algorithms are as follows,

$$\min \mathrm{C}\sum_{i=1}^{m}[x^i COST_1\left(\theta^T f^i\right) + \left(1 - x^i\right) * COST_0(\theta^T f^i)] + \frac{1}{2}\sum_{j=1}^{m}\theta_j^2 \quad (13)$$

$$COST_1(x) = \begin{cases} -\frac{11}{16}x + \frac{11}{16}, & x < 1 \\ 0, & x \geq 1 \end{cases}$$

$$COST_0(x) = \begin{cases} \frac{11}{16}x + \frac{11}{16}, & x > -1 \\ 0, & x \leq -1 \end{cases}$$

$$x^i = \begin{cases} 1, & \theta^T f^i \geq 0 \\ 0, & \theta^T f^i < 0 \end{cases}$$

$$f^i = (1 f_1^i f_2^i \dots f_m^i)$$

$$f_j^i = \exp\left(-\frac{\left|R^i - R^j\right|^2}{2\sigma^2}\right)$$

$$\forall j = 1,2,\dots,m$$

When using SVM to predict resource reallocation, C and σ are important parameters. C indicates the accuracy of the estimation boundary, and σ indicates the range of influence of each value. The advantage of SVM is that when a small sample training set is used for training, it can also obtain good estimation accuracy,

$$\theta = (\theta_0 \theta_1 \dots \theta_m) \quad (14)$$

According to SU requests, the estimation function is,

$$P_i^W = \theta_0 + \theta_1 f_1^i + \theta_2 f_2^i + \cdots + \theta_m f_m^i, P_i^W \in \mathbb{R} \tag{15}$$

P_i^W is the probability of an SU winning the bid. $P_i^W > 0$ means that an SU has a strong chance to win the bidding resources.

We compared the training times among the three algorithms (Figure 2). When the training set size was the same, the SVM's training time was the longest, and the LG's time was the shortest. The LN's speed was in the middle. Among the three algorithms, the LN had the smallest error rate. The main reason is that its cost function has the characteristics of the optimal payment price. Compared to LG and SVM, an extra factor needs to be considered in LN to give a higher prediction accuracy. Overall, the proposed LN was found to be qualified and can be implemented in the decentralized auction design.

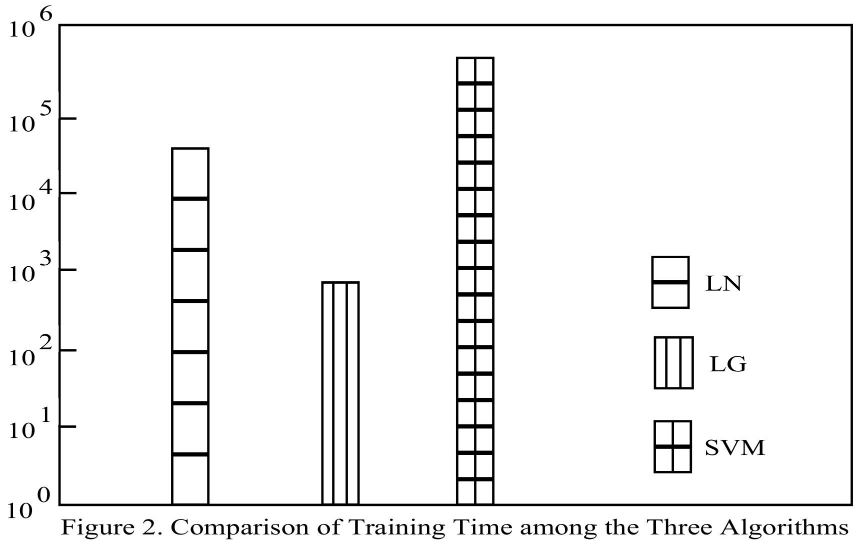

Figure 2. Comparison of Training Time among the Three Algorithms

6.3 Analysis of Resource Allocation Forecast Results

After obtaining the optimal prediction models of the three algorithms, 5 test sets were randomly generated, for instance, 1000, 2000, 3000, 4000, and 5000 SU requests. The social welfare obtained by the two algorithms (LG and SVM), based on supervised learning, was lower than the proposed optimal strategy, but was very close (Figure 3). This shows that the optimal allocation solution has specific pattern. It can be classified by a supervised learning algorithm and can be fitted by a regression model.

See: https://inatgi.in/index.php/jtis/index for more information. https://doi.org/10.63646/JZEI5817

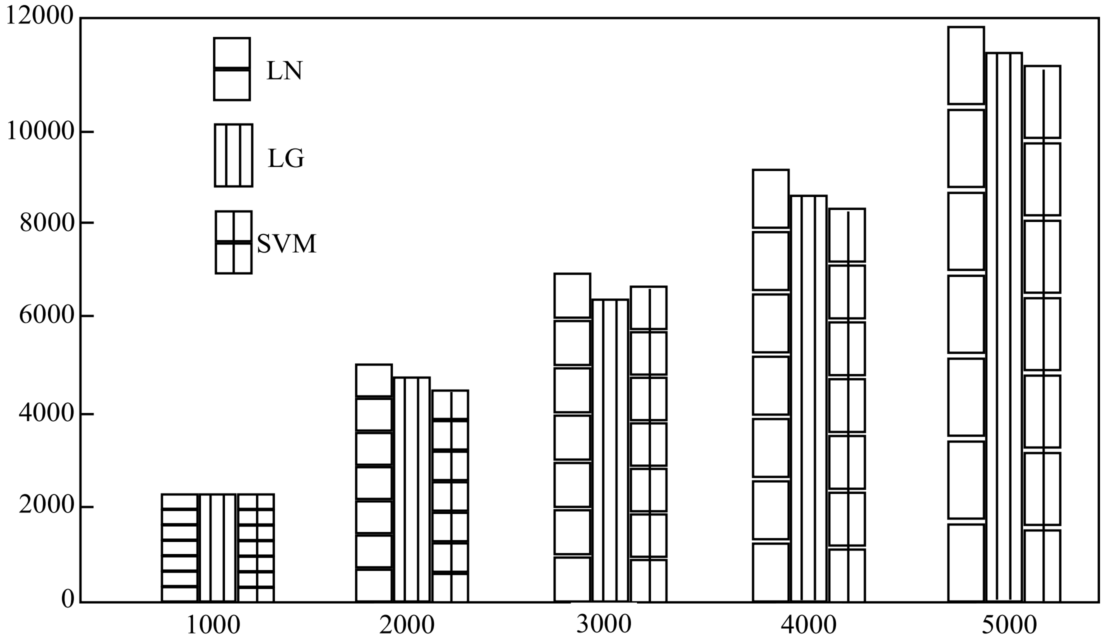

Figure 3. Comparison of Social Welfare

Figure 4 shows the prediction accuracy of different algorithms relative to the optimal allocation. The accuracy can reflect the fairness of an important indicator of the algorithm in resource reallocation. All predictions based on supervised learning algorithms had a high accuracy rate (above 95%). Among them, the accuracy of the LN algorithm was above 97%, and the proposed optimal strategy had the highest accuracy among all of the testing sets.

Figure 5 shows three resource utilizations in different algorithms, given the resource capacities of the CPU, the RAM, and the storage. Based on the results, LN and LG had similar good performances, but the optimal strategy performed the best among the three resource utilizations -- in detail, CPU (100%), RAM (60%), and storage (100%).

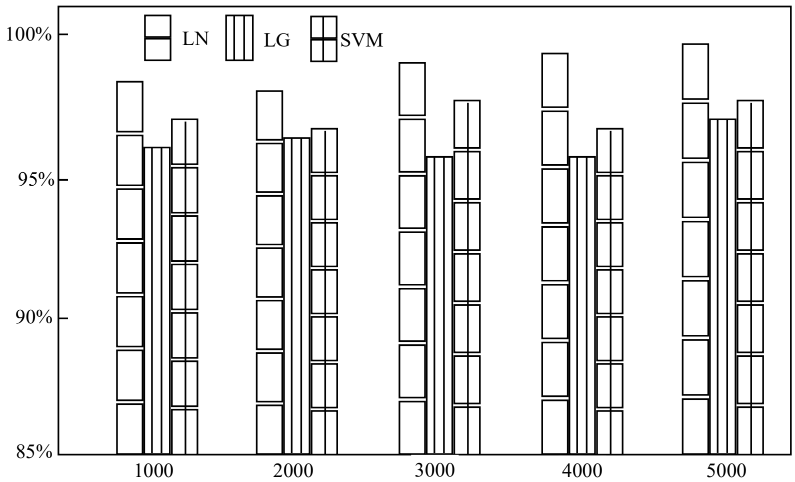

Figure 4. Comparison of Prediction Accuracy

The supervised linear regression (LN) performed very well in the test. Consistent with the previous theoretical study, the hypothesis function of the linear regression had the characteristics of the optimal payment. Also, the linear regression had the extra dimension of variable in the calculation. Hence, the accuracy of the prediction was higher.

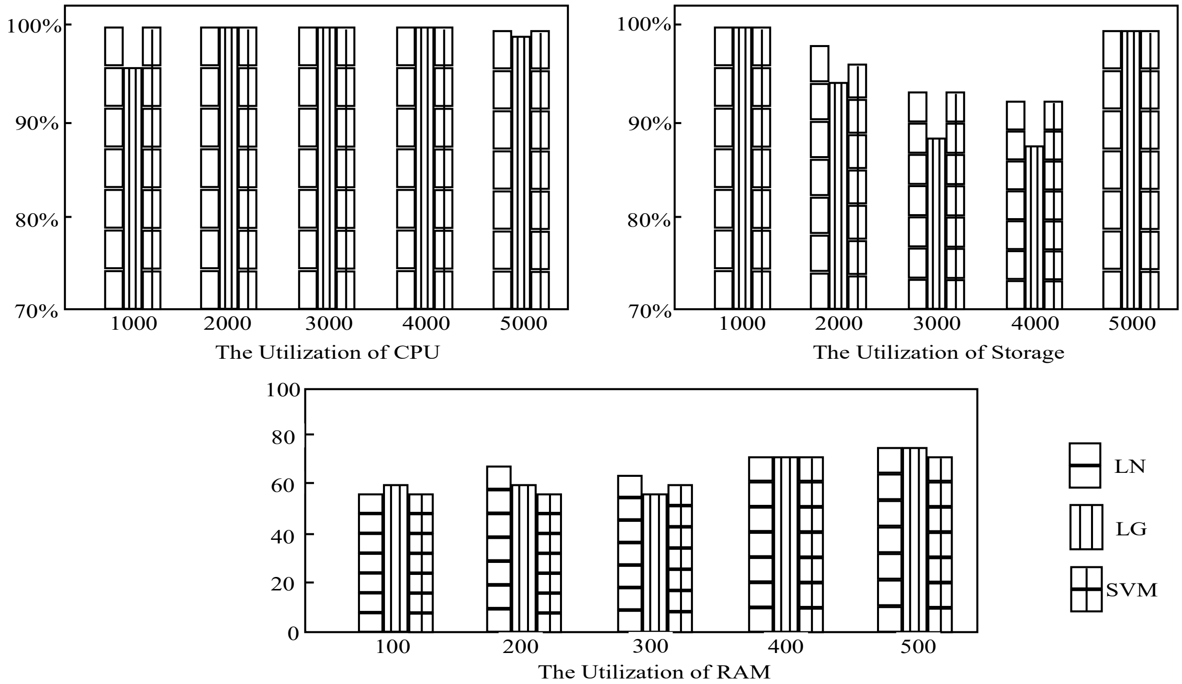

Figure 5. Comparison of Resources Utilization

7. Discussions

7.1 Computational Complexity

Buyers (users) will compete and submit bids of reputation values. The mathematical issue (NP-Hardness) and the computational complexity should be carefully considered, as well (Lu, et al., 2020). Another direction for future study is to examine alternative metrics. We employed reputation value as the indicator of pricing in this study. Multiple indicators, such as security and network externalities, can be added to represent the relationship between price and cloud computing. The more parameters added into the auction algorithm, the more accurate and practical the estimates will be. This is worth investigating in real-world industries, whereas the involvement of more cloud metrics may lead to a complexity of algorithms (Bayramusta and Nasir, 2016).

7.2 Blockchain-based Transaction Mechanism

The following are several issues related to blockchain mechanisms (Lu, 2019).
(1) The cost of blockchain node. The decentralized trading system needs to integrate a number of information-based tools and facilities. A simplified blockchain-based

framework could be better, in order to reduce the cost of the node.
(2) The storage mode of data in blockchain. Not all data needs to be recorded in the blockchain. Business-relevant data can be separated from common recorded data. In particular, common data could be stored off-chain.
(3) Improvement of communication efficiency. The communication efficiency and the performance of the underlying channel are poor. According to different requirements, the blockchain verification mechanism can be simplified, or the node's recording can be distinguished, thereby improving the communication efficiency of the entire blockchain system.
(4) Selection of different blockchain platforms. There are three popular blockchain platforms: private, public, and the consortium blockchain. This study focuses on the design of the consortium blockchain trading system. Future research could compare the three blockchain platforms to seek an appropriate one to fit a cloud transaction. For instance, the openness to participants or the requests for security level will lead to a different blockchain infrastructure for implementing cloud resources.

8. Conclusion

In a cloud environment, the dominant pricing strategies of leading companies that market the cloud are to use certain pricing models to sell their services. In our work, we address the reallocation of cloud resources to secondary users who are flexible in their bids for their preferred cloud services in a blockchain-based crowdsensing system. We highlight the decentralized crowdsensing process, the reputation incentive mechanism, and the supervised linear regression algorithm. This study converts cloud reallocation and pricing into a training and classification problem analyzed by a supervised linear regression algorithm. Based on the optimal allocation and pricing portfolios in the training set, the appropriate model was determined. The parameters learned from the supervised linear algorithm reflect the optimal strategy for cloud resources in an auction design. The estimation model from the cross-validation set ensures that the selection of winning-users is precise. The proposed model has a good performance in accuracy, truthfulness, social welfare, and resource utilization.

Reference

Al-Roomi, M., Al-Ebrahim, S., Buqrais, S., & Ahmad, I. (2013). Cloud computing pricing models: a survey. International Journal of Grid and Distributed Computing, 6(5), 93-106. DOI: 10.14257/ijgdc.2013.6.5.09.
Bayramusta, M., & Nasir, V. A. (2016). A fad or future of IT?: A comprehensive literature review on the cloud computing research. International Journal of Information Management, 36(4), 635-644. DOI: 10.1016/j.ijinfomgt.2016.04.006.
Capponi, A., Fiandrino, C., Kantarci, B., Foschini, L., Kliazovich, D., & Bouvry, P. (2019). A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities. IEEE communications surveys & tutorials, 21(3), 2419-2465. DOI: 10.1109/COMST.2019.2914030.

Chatterjee, S., Ladia, R., & Misra, S. (2015). Dynamic optimal pricing for heterogeneous service-oriented architecture of sensor-cloud infrastructure. IEEE Transactions on Services Computing, 10(2), 203-216. DOI: 10.1109/TSC.2015.2453958.

Do, C. T., Tran, N. H., Huh, E. N., Hong, C. S., Niyato, D., & Han, Z. (2016). Dynamics of service selection and provider pricing game in heterogeneous cloud market. Journal of Network and Computer Applications, 69, 152-165. DOI: 10.1016/j.jnca.2016.04.012.

Ganti, R. K., Ye, F., & Lei, H. (2011). Mobile crowdsensing: current state and future challenges. IEEE communications Magazine, 49(11), 32-39. DOI: 10.1109/MCOM.2011.6069707.

Gupta, P., Seetharaman, A., & Raj, J. R. (2013). The usage and adoption of cloud computing by small and medium businesses. International journal of information management, 33(5), 861-874. DOI: 10.1016/j.ijinfomgt.2013.07.001.

He, J., Wen, Y., Huang, J., & Wu, D. (2013). On the cost–QoE tradeoff for cloud-based video streaming under Amazon EC2's pricing models. IEEE Transactions on Circuits and Systems for Video Technology, 24(4), 669-680. DOI: 10.1109/TCSVT.2013.2283430.

Hsu, P. F., Ray, S., & Li-Hsieh, Y. Y. (2014). Examining cloud computing adoption intention, pricing mechanism, and deployment model. International Journal of Information Management, 34(4), 474-488. DOI: 10.1016/j.ijinfomgt.2014.04.006.

Jin, H., Wang, X., Wu, S., Di, S., & Shi, X. (2014). Towards optimized fine-grained pricing of IaaS cloud platform. IEEE Transactions on cloud Computing, 3(4), 436-448. DOI: 10.1109/TCC.2014.2344680.

Lehmann, D., Oćallaghan, L. I., & Shoham, Y. (2002). Truth revelation in approximately efficient combinatorial auctions. Journal of the ACM (JACM), 49(5), 577-602. DOI: 10.1145/585265.585266.

Li, Z., Zhang, H., O'Brien, L., Jiang, S., Zhou, Y., Kihl, M., & Ranjan, R. (2016). Spot pricing in the Cloud ecosystem: A comparative investigation. Journal of Systems and Software, 114, 1-19. DOI: 10.1016/j.jss.2015.10.042.

Liu, X., Li, W., & Zhang, X. (2017). Strategy-proof mechanism for provisioning and allocation virtual machines in heterogeneous clouds. IEEE Transactions on Parallel and Distributed Systems, 29(7), 1650-1663. DOI: 10.1109/TPDS.2017.2785815.

Luong, N. C., Wang, P., Niyato, D., Wen, Y., & Han, Z. (2017). Resource management in cloud networking using economic analysis and pricing models: A survey. IEEE Communications Surveys & Tutorials, 19(2), 954-1001. DOI: 10.1109/COMST.2017.2647981.

Mashayekhy, L., Fisher, N., & Grosu, D. (2015). Truthful mechanisms for competitive reward-based scheduling. IEEE Transactions on computers, 65(7), 2299-2312. DOI: 10.1109/TC.2015.2479598.

Milgrom, P. R., & Weber, R. J. (1982). A theory of auctions and competitive bidding. Econometrica: Journal of the Econometric Society, 1089-1122. DOI: 10.2307/1911865.

Macías, M., & Guitart, J. (2011, March). A genetic model for pricing in cloud computing markets. In Proceedings of the 2011 ACM Symposium on Applied Computing (pp. 113-118). DOI: 10.1145/1982185.1982216.

Mashayekhy, L., Nejad, M. M., & Grosu, D. (2014). A PTAS mechanism for provisioning and allocation of heterogeneous cloud resources. IEEE Transactions on Parallel and Distributed Systems, 26(9), 2386-2399. DOI: 10.1109/TPDS.2014.2355228.

Nejad, M. M., Mashayekhy, L., & Grosu, D. (2014). Truthful greedy mechanisms for dynamic virtual machine provisioning and allocation in clouds. IEEE transactions on parallel and distributed systems, 26(2), 594-603. DOI: 10.1109/TPDS.2014.2308224.
Samimi, P., Teimouri, Y., & Mukhtar, M. (2016). A combinatorial double auction resource allocation model in cloud computing. Information Sciences, 357, 201-216. DOI: 10.1016/j.ins.2014.02.008.
Sharma, B., Thulasiram, R. K., Thulasiraman, P., & Buyya, R. (2014). Clabacus: A risk-adjusted cloud resources pricing model using financial option theory. IEEE Transactions on Cloud Computing, 3(3), 332-344. DOI: 10.1109/TCC.2014.2382099.
Shi, W., Zhang, L., Wu, C., Li, Z., & Lau, F. C. (2014). An online auction framework for dynamic resource provisioning in cloud computing. ACM SIGMETRICS Performance Evaluation Review, 42(1), 71-83. DOI: 10.1145/2637364.2591980.
Truong-Huu, T., & Tham, C. K. (2014). A novel model for competition and cooperation among cloud providers. IEEE Transactions on Cloud Computing, 2(3), 251-265. DOI: 10.1109/TCC.2014.2322355.
Wang, Q., Ren, K., & Meng, X. (2012, March). When cloud meets ebay: Towards effective pricing for cloud computing. In 2012 Proceedings IEEE INFOCOM (pp. 936-944). IEEE. DOI: 10.1109/INFCOM.2012.6195844.
Yang, D., Xue, G., Fang, X., & Tang, J. (2015). Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones. IEEE/ACM transactions on networking, 24(3), 1732-1744. DOI: 10.1109/TNET.2015.2421897.
Zaman, S., & Grosu, D. (2013). Combinatorial auction-based allocation of virtual machine instances in clouds. Journal of parallel and distributed computing, 73(4), 495-508. DOI: 10.1016/j.jpdc.2012.12.006.
Zhang, J., Xie, N., Zhang, X., & Li, W. (2018). An online auction mechanism for cloud computing resource allocation and pricing based on user evaluation and cost. Future Generation Computer Systems, 89, 286-299. DOI: 10.1016/j.future.2018.06.034.