Pricing and Bidding Strategies for Cloud Spot Block Instances

Branka Mikavica* and Aleksandra KostiĆ-LjubisavljeviĆ*
* University of Belgrade - Faculty of Transport and Traffic Engineering, Belgrade, Serbia
b.mikavica@sf.bg.ac.rs, a.kostic@sf.bg.ac.rs

***Abstract* - Cloud providers offer idle resources - Spot instances, in an auction-like mechanism. Spot instances provide a dynamic and efficient manner for renting cloud resources. However, failures are often due to Spot price fluctuations. Recently introduced variety of Spot instances - Spot Block instances, run uninterrupted for a predefined duration. In this paper, we propose a model for Spot Block prices determination. Analysis of different bidding strategies in creating Spot Blocks requests is provided. Two auction-based pricing mechanisms are analyzed: Uniform price auction and Generalized Second-price auction. Cloud provider's revenue under these pricing mechanisms is also addressed.**

Keywords - Cloud resources*; *Spot Block instance; pricing; bidding; auctions

I. INTRODUCTION

Cloud computing is a paradigm for enabling network access to a scalable and elastic pool of shareable physical or virtual resources with self-service provisioning and administration on-demand [1]. Resources in a cloud are generally purchased as cloud instances, i.e. Virtual Machines (VMs) with the required CPU, memory and I/O resources. In addition, different types of VMs provide different levels of guarantees in terms of availability and termination. The price that cloud customers pay for required resources is determined by instance type and pricing mechanism. As the pioneer of cloud computing, Amazon Elastic Compute Cloud (EC2) uses three different purchasing options to improve revenue management: Reserved instances, On-demand instances and Spot instances. Reserved instances are purchased with long-term (usually a year or three years) availability guarantees, On-demand instances are purchased for specific time periods and are guaranteed to be available once assigned, and Spot instances are purchased through an auction mechanism, but their availability depends on the time-varying spot price. Reservation pricing mechanism incurs a relatively high fixed cost for cloud customers. However, it provides guaranteed access upon request, and the instances are not revocable once provisioned. With On-demand instances, cloud customers do not have guaranteed access upon request, but the instances are not revocable once provisioned. Hence, Reservation and On-demand pricing mechanisms provide guarantees on minimum sustained availability [2]. These pricing mechanisms are considered as static, since prices for cloud resources cannot be increased or decreased for under and over-utilization of the resources, respectively. With Spot Instances, cloud customers can bid in an auction mechanism for idle resources. However, this purchasing option offers no guarantees on minimum sustained availability. Cloud customers submit their requests by specifying the maximum hourly price they are willing to pay for the required resources. Spot prices are dynamically set by cloud provider depending on the availability of the idle resources and current demand. Spot instances are allocated to the customers if their bid is higher than the Spot price. When the Spot price exceeds the maximum bid price, the Spot instance is terminated. Cloud provider can also revoke Spot Instances when the supply of available Spot Instances decreases due to increased demand for On-demand and Reserved instances, and the maximum bid price is equal to the Spot price [3]. Recently, Amazon introduced a new variety of Spot instances, namely Spot Block instances. These instances run continuously for a finite duration (1 to 6 hours). Pricing is based on the requested duration and the available resources. Spot Block prices are typically 30% to 45% less than On-demand prices [4]. Submitting the bid, cloud customer specifies the maximum price willing to pay per hour along with the Block duration parameter (the number of hours that Spot Block instance will run). When there are enough available resources for the requested duration, Spot Block instance will launch and run continuously for a flat hourly price. At the end of the Spot Block duration, the instance is terminated. This model of purchasing cloud resources is suitable for tasks that need to run uninterrupted for up to 6 hours.

Main contribution of this paper is the development of a novel model for Spot Block price determination. A unit bid is introduced in order to enable comparison of cloud customers bids. In addition, we analyze three different bidding strategies under two auction mechanisms: Uniform price auction and Generalized Second-price auction. Cloud provider's revenues under various scenarios are addressed. The remainder of the paper is organized as follows. Section II presents a brief literature review. Modelling of Spot Block prices, bidding strategies and auction setup for two observed auction mechanisms are introduced in Section III. Experimental evaluations and results are given in Section IV. Section V contains discussion on obtained results Finally, concluding remarks are presented in Section VI.

II. Literature Review

Cloud providers introduced dynamic forms of pricing mechanisms in order to maximize their profits and balance supply and demand for cloud resources [5-7]. Auction mechanisms can be effective and promising solution, since they support cloud customers for truthful bidding (submitting bids that reflect real willingness to pay). The auctions ensure that cloud resources are assigned to the customers that value them the most. In addition, these mechanisms provide price variation depending on the changes in supply and demand by creating competition among cloud customers [8]. Various auction-based allocation and pricing mechanisms can be used for cloud resources [9-12]. Uniform price auction is a widespread method for Spot pricing mechanism [13]. The concept of the marginal bid (the highest unsuccessful bid) for cloud resources allocation and pricing under Second-price auction is proposed in [14]. Combinatorial auctions are often used for pricing and allocation of cloud resources. A comparison of combinatorial one-to-many auction and fixed pricing mechanisms is presented in [15], where the two mechanisms based on the combinatorial auctions are proposed. The auction-based mechanisms show better performances than fixed pricing mechanisms. A periodical auction mechanism, based on the limited English combinatorial model for allocation of cloud resources is proposed in [16], where the genetic algorithm with the objectives to maximize the provider's profit and to reduce customers' requests execution times is used. A fair multi-attribute combinatorial double auction for allocation and pricing of cloud resources is proposed in [17], where additional parameters such as fairness, reputation and imposing of penalty on the providers that make a false Quality of Service (QoS) assurance in order to win the auction are included in the analysis.

Bidding strategies for allocation and pricing of Spot instances are widely addressed [2, 18-23]. Guidelines for cloud customers in creating bids for Spot instances with consideration of the tradeoffs between cost, wait time and interruption rates are presented in [18]. A descriptive statistics approach for the analysis of Spot pricing along with consideration of typical pricing patterns including the presence of seasonal components, extremes and trends is presented in [2]. The availability and cost aware bidding framework based on the formalization of Spot instance failure model and non-linear programming model is proposed in [19]. Bidding and pricing strategies for optimizing cloud customers' utility and cloud provider's revenue are analyzed in [20].

In this paper, the two well known auctions: Uniform price auction and Generalized Second-price auction are used for Spot Block pricing. Three bidding strategies within four different scenarios are addressed. The aim of the analysis is to get an insight into cloud provider's revenues, cloud customers' payoffs and potential savings in Spot Block instances provisioning. To the best of our knowledge, this is the first paper addressing this issue.

III. Problem Statement

Let us consider a setting with a single cloud provider, where the provider offers its idle resources, i.e. Spot instances, in an auction-like manner. We assume that all cloud instances have the same characteristics in terms of computing and storage performances. Analysis can be easily extended to various types of cloud instances. Without loss of generality, our analysis is executed into N consecutive time slots of one hour duration. The number of available idle instances, that may be offered in the form of Spot Block instances, varies according to the number of initiated and released Reserved, On-demand, Spot instances and Spot Block instances that already run. This largely depends on the demand for cloud resources. The total number of cloud instances is denoted as M. In this paper, we assume that each instance in each time slot (unless already initiated as Spot Block instance) can be initiated as Reserved, On-demand or Spot instance, i.e. this instance is unavailable for initiation as Spot Block instance. The probability that cloud instance is unavailable is denoted as Q. Considering that sufficient number of the participants in the market is needed for revenue maximization in long term [17], we assume that number of the cloud customers, denoted as B, is always greater than the number of the available Spot Block instances.

The number of the customers that initiate request for Spot Block instance can be modelled using Poisson distribution [24]. In this paper, the two parameters for Poisson distribution, λ_1 and λ_2, are used for the periods of high and low traffic load, respectively.

In order to initiate a certain Spot Block, cloud customer creates a bid. Submitting a bid, cloud customer defines two parameters: the value of the bid, i.e. the maximum price per hour that customer is willing to pay for a given Spot Block, and the Spot Block duration, expressed in hours (up to 6 hours). Bids are submitted at the beginning of each time slot. Cloud provider allocates the available Spot Block instances, denoted as M_i, $M_i \le M$, to customers that value them the most. Customers with the greatest bid values can initiate instances that will run for predefined duration without termination. The value that customer pay for Spot Block initiation is not the value of the bid, but the value of Spot Block price per hour. Depending on applied auction mechanism, these prices can have different values.

A. Modelling of Spot Block prices

Spot Block prices differ depending on the duration of Spot Block usage. The Spot Block price per hour for Spot Block instance in time slot $i \in [1, N]$ for the duration of t is denoted as: $p_{i,t}$, $t = \{1, 2, 3, 4, 5, 6\}$. These prices are not known in advance. Their values for current time slot are set depending on cloud customers bids and applied auction mechanism. However, Spot Block prices for the previous time slot $i-1$, $p_{i-1,1}$ and $p_{i-1,6}$, are publicly available [25]. The greatest Spot Block price is set for the maximum Spot Block duration of 6 hours, while the lowest Spot Block price is set for Spot Block duration of one hour. Therefore, it applies:

$$p_{i-1,1} < p_{i-1,2} < p_{i-1,3} < p_{i-1,4} < p_{i-1,5} < p_{i-1,6} . \quad (1)$$

Spot Block prices for Spot Block duration of 2, 3, 4 and 5 hours in time slot $i-1$ can be easily calculated from $p_{i-1,1}$ and $p_{i-1,6}$. We assume that

$$p_{i-1,6} = (1+\theta)\cdot p_{i-1,1}, \quad \theta \in (0,1). \tag{2}$$

The parameter θ is introduced in order to evaluate the difference between the values of Spot Block prices per hour for Spot Block duration of one hour and six hours. Therefore, Spot Block prices for duration t can be presented as follows:

$$p_{i-1,t} = \left(1+\frac{(t-1)\cdot\theta}{5}\right)\cdot p_{i-1,1}, \quad t = \{2, 3, 4, 5\}. \tag{3}$$

B. Bidding strategies

Cloud customers submit their bids in time interval $i \in [1, N]$ based on information about on-demand prices and Spot Block prices per hour in previous time interval. On-demand price for equivalent on-demand instance (instance with the same computing and storage performances offered on-demand) is denoted as p_o. Set of all bidders in time interval $i \in [1, N]$ is denoted as B_i. The bid of the cloud customer $j \in B_i$ in time slot $i \in [1, N]$ can be expressed as follows:

$$b_{i,j} = \left(\upsilon^i_{j,t_j}, t^i_j\right), \tag{4}$$

where υ_{j,t_j} represents the value of the bid for the duration of t_j.

Depending on their willingness to pay, customers may perform the following bidding strategies: bidding close to the Spot Block price for one hour in previous time interval, $p_{i-1,1}$; bidding close to the Spot Block price for six hours in previous time interval, $p_{i-1,6}$ and bidding close to the on-demand price. These strategies are chosen according to a discrete probability distribution.

When cloud customers place bids close to the Spot Block price for one hour in previous time interval, their bids take values

$$b_{i,k} \in \left[p_{i-1,1}-\delta_1,\ p_{i-1,1}+\delta_1\right], \tag{5}$$

where $i \in [1, N]$, $k \in [1, |B_i|]$. δ_1 denotes a small variation of the Spot Block price for one hour in previous time interval. The probability of choosing the first bidding strategy is denoted as $q_1, q_1 \in (0,1)$.

For the second bidding strategy, it applies:

$$b_{i,l} \in \left[p_{i-1,6}-\delta_6,\ p_{i-1,6}+\delta_6\right] \tag{6}$$

where $i \in [1, N]$, $l \in [1, |B_i|]$, $l \neq k$, δ_6 denotes a small variation of the Spot Block price for one hour in previous time interval. The probability of choosing the second bidding strategy is denoted as $q_2, q_2 \in (0,1)$.

The third bidding strategy is characterized by bid values from the range

$$b_{i,m} \in \left[p_o-\delta_o,\ p_o\right], \tag{7}$$

where $i \in [1, N]$, $m \in [1, |B_i|]$, $m \neq l \neq k$. δ_o is used to denote the lower bound in variation of the on-demand price per hour. The probability of choosing the third bidding strategy is denoted as $q_3, q_3 \in (0,1)$. Obviously, $q_1+q_2+q_3=1$.

C. Auction Mechanisms

Comparison of bids is not as straightforward as in the case of bidding for regular Spot instances. The complexity is introduced by allowing that Spot Block instance runs without termination even if the value of the bid is already exceeded. Considering that Spot Block prices per hour are different for different Spot Block duration, bids cannot be simply sorted depending on their value. In this paper, we propose a model for translation of bids into unit bids in order to enable their comparison in an auction mechanism. Each bid is represented by a new value of the bid and by the same Spot Block duration, as follows:

$$\overline{b_{i,j}} = \left(\overline{\upsilon^i_{j,t_j}}, t_j\right), \tag{8}$$

where $\overline{\upsilon^i_{j,t_j}}$ represents the unit bid value of cloud customer j in time slot $i \in [1, N]$. In accordance with (3), the unit bid value can be calculated as:

$$\overline{\upsilon^i_{j,t_j}} = \begin{cases} \dfrac{5\cdot\upsilon^i_{j,t_j}}{5+\left(t^i_j-1\right)\cdot\theta}, & t^i_j = \{2, 3, 4, 5, 6\}. \\ \upsilon^i_{j,t_j}, & t^i_j = 1 \end{cases} \tag{9}$$

Once unit bids in certain time slot are determined, the allocation of Spot Block instances is performed based on the set of winning bids, i.e. cloud customers with the winning bids can initiate required Spot Block instances. Cloud customers payoffs differ depending on the applied auction mechanism. In this paper, we analyze Uniform price auction and Generalized Second-price auction in order to obtain cloud provider's revenues in provisioning of Spot Block instances. The set of the winning bids contains M_i the highest unit bids,

$$W_i^I = \left\{ \overline{\upsilon_{1,t_1}^i}, \overline{\upsilon_{2,t_2}^i}, \cdots, \overline{\upsilon_{M_i,t_{M_i}}^i} \right\}, \tag{10}$$

where M_i denotes the number of the available Spot Block instances in time interval $i \in [1, N]$. This number varies since Spot Block instances can be occupied without termination up to 6 hours.

In Uniform price auction, each winning bidder pays the same price equal to the lowest winning bid. The set of the cloud customers payoffs in time interval $i \in [1, N]$ under Uniform price auction is:

$$P_i^I = \left\{ \overline{\upsilon_{1,t_1}^i}, \overline{\upsilon_{2,t_2}^i}, \cdots, \overline{\upsilon_{M_i,t_{M_i}}^i} \right\}, \tag{11}$$

where $\overline{\upsilon_{1,t_1}^i} > \overline{\upsilon_{2,t_2}^i} > \cdots > \overline{\upsilon_{M_i,t_{M_i}}^i}$. Therefore, Spot Block price for Spot Block duration of one hour in time slot $i \in [1, N]$ is $p_{i,1}^I = \overline{\upsilon_{M,t_{M_i}}^i}$. According to (3), Spot Block prices in general can be represented as:

$$p_{i,t}^I = \left(1 + \frac{(t-1)\cdot\theta}{5}\right)\cdot p_{i,1}^I, \tag{12}$$

where $t = \{2, 3, 4, 5, 6\}$.

Cloud provider's revenue under Uniform price auction in time slot $i \in [1, N]$ can be expressed as:

$$R_i^I = \sum_{q=1}^{|W_i^I|} p_{i,t_q}^I + \sum_{h=1}^{|W_{i-1}^I|} p_{i-1,t_h}^I + \sum_{z=1}^{|W_{i-2}^I|} p_{i-2,t_z}^I + \sum_{e=1}^{|W_{i-3}^I|} p_{i-3,t_e}^I + \sum_{g=1}^{|W_{i-4}^I|} p_{i-4,t_g}^I + \sum_{r=1}^{|W_{i-5}^I|} p_{i-5,t_r}^I, \tag{13}$$

where $t_h \geq 2$, $t_z \geq 3$, $t_e \geq 4$, $t_g \geq 5$, $t_r = 6$. W_{i-1}^I denotes the set of winning bids in time interval $i-1$, W_{i-2}^I denotes the set of winning bids in time interval $i-2$, etc. Equation (13) implies that cloud provider's revenue in one time slot depends on the Spot Block prices determined in current time slot and the prices of Spot Block instances initiated in five previous time intervals with corresponding duration (in time slot $i-5$, relevant requests are those with duration of 6 hours, in time slot $i-4$ relevant requests are those with duration that is equal or greater than 5 hours, etc).

When cloud provider applies Generalized Second-price auction winning bidders pay the value of the next highest bid. This means that each cloud customer has different set of possible payoffs. For each bidder, the set of unit bids is determined. However, Spot Block prices for each winning bidder are determined based on the unit bid of the next highest unit bidder. The set of the cloud customers payoffs in time interval $i \in [1, N]$ under Generalized Second-price auction is:

$$P_i^{II} = \left\{ \overline{\upsilon_{2,t_2}^i}, \overline{\upsilon_{3,t_3}^i}, \cdots, \overline{\upsilon_{M_i+1,t_{M_i+1}}^i} \right\}, \tag{14}$$

where $\overline{\upsilon_{2,t_2}^i} > \overline{\upsilon_{3,t_3}^i} > \cdots > \overline{\upsilon_{M+1,t_{M_i+1}}^i}$.

In this case, the Spot Block price per hour of cloud customer $q \in W_i^{II}$ can be expressed as follows:

$$p_{i,t_q}^{II} = \left(1 + \frac{(t_q-1)\cdot\theta}{5}\right)\cdot \overline{\upsilon_{q+1,\ t_{q+1}}^i} \tag{15}$$

where $t_q = \{2, 3, 4, 5, 6\}$ and $\overline{\upsilon_{q+1,\ t_{q+1}}^i}$ is the next highest unit bid. Since cloud customers pay different Spot Block prices, we assume that the lowest winning unit bid in previous time interval is known in advance. These information provide decision support in choosing appropriate bidding strategies.

In accordance with (13), cloud provider's revenue under Generalized Second-price auction in time slot $i \in [1, N]$ can be expressed as:

$$R_i^{II} = \sum_{q=1}^{|W_i^{II}|} p_{i,t_q}^{II} + \sum_{h=1}^{|W_{i-1}^{II}|} p_{i-1,t_h}^{II} + \sum_{z=1}^{|W_{i-2}^{II}|} p_{i-2,t_z}^{II} + \sum_{e=1}^{|W_{i-3}^{II}|} p_{i-3,t_e}^{II} + \sum_{g=1}^{|W_{i-4}^{II}|} p_{i-4,t_g}^{II} + \sum_{r=1}^{|W_{i-5}^{II}|} p_{i-5,t_r}^{II}. \tag{16}$$

IV. Performance Evaluation

In order to analyze proposed scenarios, simulations in open source programming language Python 2.7 are performed in 1000 iterations. Time period of 5 days is simulated (Monday-Friday). Each day is divided into $N = 24$ time slots of one hour duration. Time slots from $i = 7$ up to $i = 20$ belong to the period of high traffic load [26]. The number of customers initiating requests in this the period of high traffic load is modelled by Poisson distribution parameter $\lambda_1 = 1.5$, while the period of low traffic load is modelled by Poisson distribution parameter $\lambda_1 = 0.5$. The number of cloud instances is $M = 50$. The average number of cloud customers is $B = 250$. The assumed probability of cloud instance unavailability (due to reserved, on-demand or spot instance initiation) is $Q = 0.33$.

The initial Spot Block prices for Spot Block duration of one hour and six hours are chosen from publicly available data for Amazon Spot Block instance *m4.10xlarge* in EU (Frankfurt) region and Windows operating system [25]. Spot Block prices for duration of one hour and six hours are $p_{0,1}$=3.04 $/*hour* and $p_{0,6}$=3.4 $/*hour*, respectively. According to (2) and (3), θ=0.12 while Spot Block prices for Spot Block duration of 2, 3, 4 and 5 hours are: $p_{0,2}$=3.112 $/*hour*, $p_{0,3}$=3.184 $/*hour*, $p_{0,4}$=3.256 $/*hour* and $p_{0,5}$=3.328 $/*hour*, respectively. On-demand price per hour of equivalent on-demand cloud instance is p_o=4.24 $/*hour* [25].

Considering these prices, it is reasonable to assume that values for parameters determining boundaries in bidding for appropriate bidding strategy are: $\delta_1 = \delta_6 = \delta_o = 0.2$.

We observe four scenarios depending on dominant bidding strategy. Scenarios 1, 2 and 3 analyze setting where cloud customers predominantly choose the first (q_1=0.50, q_2=0.30, q_3=0.20), the second (q_1=0.30, q_2=0.50, q_3=0.20), and the third bidding strategy (q_1=0.20, q_2=0.30, q_3=0.50), respectively. The fourth scenario analyzes the setting where all available bidding strategies are equally possible, q_1=q_2=q_3=0.33. After the first round of auctions, the Spot Block price for Spot Block duration of 1 hour (unit bid) is known publicly in the case of Uniform price auction. In the case of Generalized Second-price auction, the lowest winning unit bid is known.

Table 1 shows the average winning bids (expressed in $) for all observed scenarios in the periods of high and low traffic loads under both auction mechanisms. These values are approximately the same, i.e. both auctions have nearly the same impact on cloud customers in the terms of bidding. Although customers pay different Spot Block prices under these auction mechanisms, their bid values are nearly the same. This means that in the terms of cloud customers' incentive to bid truthfully (submitting bids that reflect their actual willingness to pay), the applied mechanisms have the same effect.

TABLE I. AVERAGE WINNING BIDS [$]

Scenario	**Uniform price auction**		**Generalized Second-price auction**	
	High traffic load	***Low traffic load***	***High traffic load***	***Low traffic load***
1	4.090	3.942	4.090	3.941
2	4.112	3.978	4.112	3.977
3	4.131	4.018	4.131	4.018
4	4.116	3.990	4.116	3.989

Table 2 shows the average revenue per instance (expressed in $) that cloud providers achieves in the Spot Block market. This revenue depends on the Spot Block prices defined in an auction mechanism in current time slot and the Spot Block prices defined in previous five intervals. This is due to the fact that Spot Block instances can be run in 6 time slots without termination. The results in Table 2 show that Generalized Second-price auction provide slightly greater revenues in comparison with Uniform price auction.

TABLE II. AVERAGE CLOUD PROVIDER'S REVENUE IN TIME SLOT PER INSTANCE [$]

Scenario	**Uniform price auction**		**Generalized Second-price auction**	
	High traffic load	***Low traffic load***	***High traffic load***	***Low traffic load***
1	3.142	3.268	3.197	3.360
2	3.090	3.235	3.137	3.318
3	3.039	3.189	3.083	3.263
4	3.081	3.226	3.128	3.305

The average savings (expressed in %) over On-demand instances are shown in Table 3. These values present the potential savings that cloud customers obtain for using Spot Block instead On-demand instances. It is notable that all scenarios provide savings greater than 20%. The greatest savings (28.334%) are achieved in the scenario where the dominant strategy is bidding close to On-demand price under Uniform price auction in the period of high traffic load.

TABLE III. AVERAGE SAVINGS OVER ON-DEMAND PRICES [%]

Scenario	**Uniform price auction**		**Generalized Second-price auction**	
	High traffic load	***Low traffic load***	***High traffic load***	***Low traffic load***
1	25.907	22.917	24.600	20.746
2	27.122	23.708	26.011	21.755
3	28.334	24.794	27.283	23.051
4	27.331	23.915	26.221	22.062

V. DISSCUSSION

Recently introduced Spot Block instances provide several advantages over classic Spot instances for both cloud providers and cloud customers. Cloud providers have better utilization of idle resources, while cloud customers can initiate these instances up to 6 hours without termination. In addition, the determined Spot Block price per hour for required Spot Block duration is fixed for this specified time interval. Therefore, once determined, the given price is fixed regardless on changes in supply and demand in cloud instances market. This is a great improvement in comparison with Spot instances, where cloud provider can terminate the instance whenever the spot price exceeds customer's bid. In general, Spot Block prices are greater that Spot prices, but significantly lower that On-demand instances. The results show that regardless of the chosen bidding strategy, greater revenues in average are achieved under Generalized Second-price auction. However, from cloud customers' perspective, Uniform price auction is more convenient, since it provides greater savings over On-demand price. This approves that Spot Block instances are cost-effective solution for occasional workloads that require uninterrupted provisioning in duration less than 6 hours.

VI. CONLUSION

This paper presents analysis of auction-based pricing mechanisms for pricing cloud provider's idle resources in the form of recently introduced Spot Block instances. Uniform price auction, as a widespread pricing mechanism for cloud resources, is compared with Generalized Second-price auction. Initiating request for the access to the Spot Block instance, cloud customer choose one of the three possible bidding strategies. Depending on dominant bidding strategy, we observe several scenarios. In this paper, a model for Spot Block prices determination under these pricing mechanisms is proposed. This model provides an appropriate insight into price dependences of chosen auction mechanism. However, it requires that Spot Block prices in previous time interval are known in advance. We analyze average winning bids, average revenues per instance and average potential savings over On-demand pricing mechanism. In all observed scenarios, the significant savings over On-demand instances are achieved. In addition, Uniform price auction provides slightly less Spot Block prices and greater savings over On-demand instances. The results show that regardless on the chosen auction mechanism and regardless on chosen bidding strategy Spot Block instances are cost-effective solution that embodies advantages of On-demand instances and Spot instances.

ACKNOWLEDGMENT

This work was supported by the Ministry of Education, Science and Technological Development of Serbia [grant number TR 32025].

REFERENCES

[1] ITU-T Recommendation Y.3500, Information technology ï Cloud computing ï Overview and vocabulary, (2014).

[2] M. Lumpe, M. B. Chhetri, Q. B. Vo and R. Kowalczyk, ñOn Estimating Minimum Bids for Amazon EC2 Spot Instancesò, in *Proc. 2017 17th IEEE/ACM Int. Simp. on Cluster, Cloud and Grid Computing (CCGRID)*, pp. 391-400, Madrid, Spain, 2017.

[3] R. Wolski and J. Brevik, ñProviding Statistical Reliability Guarantees in the AWS Spot Tierò, in *Proc. of the 24th High Performance Computing Symposium. Society for Computer Simulation International*, p. 13, Pasadena, USA, 2016.

[4] Amazon EC2 Spot instances, 2018. Available Online: https://aws.amazon.com/blogs/aws/new-ec2-spot-blocks-for-defined-duration-workloads/

[5] Z. Cai, X. Li, R. Ruiz, Q. Li, , ñPrice Forecasting for Spot Instances in Cloud Computingò, *Future Generation Computer Systems*, vol. 79(1), pp. 38-53, 2018.

[6] N. A. B. M. Shaari, T. Fong Ang, L. Yee Por, C. Sun Liew, ñDynamic Pricing Scheme for Resource Allocation in Multi-Cloud Environmentò, *Malaysian Journal of Computer Science*, vol. 30(1), pp. 1-11, 2018.

[7] S. G. Domanal, G. R. Mohana Reddy, ñAn Efficient Cost Optimized Scheduling for Spot Instances in Heterogeneous Cloud Environmentò, *Future Generation Computer Systems*, vol. 84, pp. 11-21, 2018.

[8] A. N. Toosi, K. Vanmechelen, F. Khodadadi and R. Buyya, ñAn Auction Mechanism for Cloud Spot Marketsò, *ACM Transactions on Autonomous and Adaptive Systems*, vol. 11(1), pp. 25-57, 2016.

[9] D. Kumar, G. Baranwal, Z. Raza, D. P. Vidyarthi, ñA Systematic Study of Double Auction Mechanisms in Cloud Computingò, *J. of Systems and Software*, vol. 125, pp. 234-255, 2017.

[10] D. Kumar, G. Baranwal, Z. Raza, D. P. Vidyarthi, ñA Truthful Combinatorial Double Auction-Based Marketplace Mechanism for Cloud Computingò, *J. of Systems and Software*, vol. 140, pp. 91-108, 2018.

[11] L. Lu, J. Yu, Y. Zhu, M. Li, ñA Double Auction Mechanism to Bridge Usersô Task Requirements and Providersô Resources in Two-Sided Cloud Marketsò, *IEEE Transaction on Parallel and Distributed Systems*, vol. 29(4), pp. 720-733, 2018.

[12] X. Li, R. Ding, X. Liu, X.Liu, E. Zhu, Y. Zhong, ñA Dynamic Pricing Reverse Auction-Based Resource Allocation Mechanism in Cloud Workflow Systemsò, *Scientific Programming*, vol. 2016, pp. 1-13, 2016.

[13] Q. Zhang, E. Gurses and R. Boutaba, ñDynamic Resource Allocation for Spot Markets in Cloudsò, in *Proc. of the 4th IEEE Int. Conf. on Utility and Cloud Computing (UCC)*, pp. 178-185, Victoria, Australia, 2011.

[14] W.-Y. Lin, G.-Y. Lin and H.-Y. Wei, ñDynamic Auction Mechanism for Cloud Resource Allocationò, in *Proc. of the 10th IEEE/ACM Int. Conf. on Cluster, Cloud and Grid Computing (CCGrid)*, pp. 591-592, Melbourne, Australia, 2010.

[15] S. Zaman and D. Grosu, ñCombinatorial Auction-Based Allocation of Virtual Machine Instances in Cloudsò, *J. of Parallel and Distributed Computing*, vol. 73(4), pp. 495-508, 2013.

[16] W. Xing-Wei, W. Xue-Yi and H. Min, ñA Resource Allocation Method Based on the Limited English Combinatorial Auction Under Cloud Computing Environmentò, in *Proc. of the IEEE Int. Conf. on Fuzzy Systems and Knowledge Discovery (FSKD)*, pp. 905-909, Chongqing, China, 2012.

[17] G. Baranwal and D. P. Vidyarthi, ñA Fair Multi-Attribute Combinatorial Double Auction Model for Resource Allocation in Cloud Computingò, *J. of Systems and Software*, vol. 108, pp. 60-76, 2015.

[18] S. Karunakaran and R. P. Sundarraj, ñBidding Strategies for Spot Instances in Cloud Computing Marketsò, *J. of IEEE Internet Computing*, vol. 19(3), pp. 32-40, 2015.

[19] W. Guo, K. Chen, Y. Wu and W. Zheng, ñBidding for Highly Available Services with Low Price in Spot Instance Marketò, in *Proc. of 24th Int. Symp. on High-Performance Parallel and Distributed Computing*, pp. 191-202, Portland, USA, 2015.

[20] J. Song and R. Guerin, ñPricing and Bidding Strategies For Cloud Computing Spot Instancesò, in *Proc. of the IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 647-653, Atlanta, USA, 2017.

[21] J. Wank, X. Gui, R. Zhang, ñDynamic Bidding in Spot Market for Profit Maximization in the Public Cloudò, *J. of Supercomputing*, vol. 73(10), pp. 4245-4274, 2017.

[22] Z. Li, M. Kihl, A. Robertsson, ñOn a Feedback Control-based Mechanism of Bidding for Cloud Spot Serviceò, in *Proc. of the 7th IEEE Int. Conf. on Cloud Computing Technology and Science (CloudCom)*, pp. 290-297, Vancouver, Canada, 2015.

[23] L. Zheng, C. Joe-Wong, C. Wei Tan, M. Chiang, X. Wang, ñHow to Bid the Cloudò, *ACM Sigcomm Computer Communication Review*, vol. 45(4), pp. 71-84, 2015.

[24] B. Mikavica and A. Kostiĺ-Ljubisavljeviĺ, ñInterconnection Contracts between Service and Content Provider with Partial Cloud Migrationò, *Elektronika Ir Elektrotechnika*, vol. 22(6), pp. 92-98, 2016.

[25] Amazon EC2 Spot pricing, 2018. Available Online: https://aws.amazon.com/ec2/spot/pricing/

[26] Y. Jiang, X. Ma and W. Chen, ñCost-Effective Resource Configuration for Cloud Video Streaming Servicesò, in *Proc. of the 21th IEEE Int. Conf. on Parallel and Distributed Systems (ICPADS)*, pp. 430-439, Melbourne, Australia, 2015.