G Model
IJLEO 57375 1–6
ARTICLE IN PRESS
Optik xxx (2016) xxx–xxx
Contents lists available at ScienceDirect
Optik
journal homepage: www.elsevier.de/ijleo

# Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism

Weiwei Kong[a,b,*], Yang Lei[c], Jing Ma[d]

[a] Xijing University, Xi'an 710123, China
[b] Department of Information Engineering, Engineering University of Armed Police Force, Xi'an 710086, China
[c] Department of Electronics Technology, Engineering University of Armed Police Force, Xi'an 710086, China
[d] Key Laboratory of Information Assurance Technology, Beijing 100072, China

ARTICLE INFO

*Article history:*
Received 18 January 2016
Accepted 28 February 2016
Available online xxx

*Keywords:*
Cloud computing
Virtual machine
Auction mechanism
Queuing theory

ABSTRACT

To overcome the problem of virtual machine (VM) scheduling in cloud computing, a novel adaptive VM resource scheduling algorithm based on auction mechanism is presented by considering multiple factors including network bandwidth and auction deadline. First, the sequencing of the clients' bids is conducted in the given competition deadline. Second, the client group is screened and corresponding VM resource is configured according to the minimum costs of the cloud service providers. Finally, the final payment price can be figured by considering the levels of average payments and competitive payments, so that the tasks clients request can be completed with the given VM resource. The simulation experimental results show that the proposed algorithm can effectively enhance the quality of service of the cloud environment, the profits of cloud service providers and the resource utilization rate of VM.



## 1. Introduction

Cloud computing [1–3] is the integration product between the traditional computing model such as distributed computing, virtualization and the developing network technology. Based on the above computing model, the computing, storage, platform and service resource pools can be realized, which are abstract, dynamic extending and manageable. Moreover, the resource pools can be provided to external clients on demand via Internet.

Virtual machine (VM) is one of the most typical resources in cloud computing. The resource scheduling and allocation not only are related to quality of service (QoS) [4–6], but also can affect the profits of the cloud service provider directly. At present, resource scheduling has become the hot issue in the field of cloud computing, and a series of resource scheduling algorithms [7–21] have been presented by the scholars both at home and abroad.

Commonly, the scheduling issue boils down to certain mathematical problem first. Then, the corresponding mathematical model can be constructed. Finally, the solution to the related model is figured out. Traditional cloud computing algorithms include first-in-first-out (FIFO) scheduling algorithm, greedy algorithm, fair scheduling [7,8] algorithm, and so on. However, these algorithms belong to the static type, thus there is no adaptive and dynamic adjustment mechanism. Unfortunately, the cloud computing resources are usually dynamically allocated and released, therefore, traditional algorithms cannot satisfy the practical requirements of the cloud computing resource scheduling, and the resources are wasted seriously. There is a great deal of research that shows that resource scheduling is not only a multi-constrained multi-objective optimizing issue in essence, but also a NP problem. To this end, many artificial intelligence algorithms are introduced into the solution to the cloud computing resource scheduling, such as genetic algorithm (GA), particle swarm optimization [9], clone selection algorithm, shuffled frog leaping (SFL) [10] algorithm, and so on. Due to the high efficiency of the newly proposed algorithms, they have been the mainstreamed research basis. Ref. [11] integrated GA into the scheduling model, and the original fitness function was improved via QoS, but the result easily trapped in a local optimum. Yang et al. [12] proposed a novel heuristic optimizing algorithm called bat algorithm (BA) in 2010, and a variety of improved versions were devised to deal with the resource scheduling of cloud computing. Similarly, SFL model and its modifications were also utilized to conduct the resource scheduling.

* Corresponding author at: Xijing University, Xi'an 710123, China.
Tel.: +86 15609231541.
*E-mail address:* kwwking@163.com (W. Kong).

Please cite this article in press as: W. Kong, et al., Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism, Optik - Int. J. Light Electron Opt. (2016), http://dx.doi.org/10.1016/j.ijleo.2016.02.061

Ref. [13] proposed a resource evaluation model based on entropy optimization and dynamic weighting. He et al. [14] presented a management model called elastic application container (EAC) to deal with the lightweight resource scheduling. In Ref. [15], an algorithm was devised to estimate the quantity of VM resources the service provider offers according to the demands of the clients require. Gong et al. [16] evaluated the VM resource demands based on the theories of both the signal and statistics. In Ref. [17], the differential calculation method was used to simulate the optimal dynamic VM resource sets. Experimental results demonstrated that the performance of the algorithm is good, but its computational costs were high. Different from most ones, Refs. [18–21] all involved the cloud resource distributing algorithms based on the auction mechanism. However, the algorithm in [18] needs the step of prior prediction on the workloads. In Ref. [19], we didn't have to conduct the prior prediction, but we paid for it with high computational complexities. By combining the superiorities of [18,19], Refs. [20,21] took many factors such as VM resources and network bandwidth into consideration, and improved the traditional auction mechanism a lot. However, due to the ignorance of the factor of competition deadline, the wasting of VM resources may appear to some extent.

Based on the background mentioned above, this paper proposes an adaptive VM resource scheduling algorithm which considers both auction mechanism and many factors such as network bandwidth and auction deadline. This algorithm consists of two phases. The one is the cloud property evaluation and VM configuration, and the other one is the client auction payment mechanism. The former is responsible for the sequencing of the clients' bids in the given competition deadline, so that the potential client group can be determined. The latter can realize the payment for the network resources and figure out the final payment price of the clients. The simulations of the algorithms involved in this paper are conducted on the CloudSim platform, and the experimental results demonstrate that the proposed algorithm can effectively enhance QoS of the cloud environment, the profits of the cloud service providers and the resource utilization rate of VM.

The rest of the paper is organized as follows. In Section 2, we give the traditional cloud computing model and four dynamic resource description indexes. The auction mechanism based VM resource scheduling algorithm and its concrete steps are given in Section 3. In Section 4, experimental results and related analysis are described. Finally, Section 5 draws a conclusion for this paper.

## 2. Traditional cloud computing model

Cloud computing can provide large batches of task requests for a large number of clients simultaneously. Once receiving the service requests, cloud service providers will distribute corresponding computing resources based on different requests from the clients or the dispenses of the cloud computing resources the clients pay for. The traditional cloud computing model is shown in Fig. 1.

The cloud computing resource especially the VM resource performance is an important index in the cloud computing service, which could measure the service probability and the level of service ability of the cloud service providers. Usually, VM resource performance evaluation is mainly based on the information of the VM resource available. The description of the current used and available VM resources can be realized by the evaluation of VM resource availability, which is beneficial for resource scheduling, distribution and transferring.

In order to efficiently describe the information of dynamic resources in the cloud computing, four dynamic resource description indexes are proposed in this paper, including request index (RI), dealing index (DI), loading index (LI) and cost index (CI).


**Fig. 1.** Traditional cloud computing model.

Suppose the VM resource set denoted by $\boldsymbol{S} = \{S_1, S_2, \ldots, S_n\}$, and $n$ is the number of VM resources.

(1) *RI*: RI is the service request quantity the cloud service providers receive in unit time. The larger the RI value is, the more resource quantity of the cloud service providers the clients request. If the service request the VM resource $S_i(1 \le i \le N)$ receives in unit time is $\mathrm{RI}_i(1 \le i \le N)$, then the RI of the set $\boldsymbol{S}$ is:

$$\mathrm{RI} = \sum_{i=1}^{n} \mathrm{RI}_i \tag{1}$$

(2) *DI*: DI is the service request quantity VM resources accomplish in unit time. The value of DI reflects the dealing ability to the clients' requests of the cloud service resources. The larger DI is, the higher the dealing ability is. If the service request quantity the VM resource $S_i(1 \le i \le N)$ accomplishes in unit time is $\mathrm{DI}_i(1 \le i \le N)$, then the DI of the set $\boldsymbol{S}$ is:

$$\mathrm{DI} = \sum_{i=1}^{n} \mathrm{DI}_i \tag{2}$$

(3) *LI*: LI is the ratio between the average time required to complete a client's service for the VM resource and the clients' adjacent service request interval time. Similar to the above two indexes, the value of LI is proportional to the loading of the cloud service resources. With regard to the VM resource $S_i(1 \le i \le N)$, if the average time required to complete a client's service is $t_i(1 \le i \le N)$ and the clients' adjacent service request interval time is $t_d$, then the LI is:

$$\mathrm{LI} = \frac{t_i}{t_d} \tag{3}$$

(4) *CI*: CI is used to weigh the cost required to maintain the normal running of the VM resource. If the running maintenance cost of the VM resource $S_i(1 \le i \le N)$ is $C_{iR}$ and the idle maintenance cost of the VM resource $S_i(1 \le i \le N)$ is $C_{iI}$, then the total maintenance CI of the cloud service provider is in the following range:

$$\mathrm{CI} = \left[ \sum_{i=1}^{n} C_{iI}, \sum_{i=1}^{n} C_{iR} \right] \tag{4}$$

Please cite this article in press as: W. Kong, et al., Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism, Optik - Int. J. Light Electron Opt. (2016), http://dx.doi.org/10.1016/j.ijleo.2016.02.061

Obviously, the competitive payments $P_i$ of the clients of the unit resource must be in the interval $[C_{iI}, C_{iR}]$ to guarantee that the cloud service provider can obtain the basic profit. If $P_i < C_{iR}$, then the loss of the cloud service provider is $C_{iR} - P_i$ and it must satisfy the inequality $C_{iR} - P_i < C_{iI}$, which means that the loss of the running VM resource is less than that of idle VM resource. As a result, the competitive payments $P_i$ of the clients must satisfy $C_{iI} \le P_i \le C_{iR}$.

## 3. Auction mechanism based VM resource scheduling algorithm

In this paper, the cloud computing VM resource scheduling algorithm is based on the competitive payments mechanism and composed of two parts. One is the cloud property evaluation and VM configuration, and the other one is the client auction payment mechanism.

After considering the network bandwidth and auction deadline, the basic models of the cloud service provider and the client group are constructed as follows.

(1) Cloud service provider $S_{\text{cloud}}$:
  (a) The number of VM resource types is $K$.
  (b) The total VM resource number is $N$.
  (c) The running and the idle maintenance costs of the VM resource $S_{k,i}$ $(1 \le k \le K,\ 1 \le i \le N)$ are $C_{k,i,R}$ and $C_{k,i,I}$, respectively.
  (d) The competition deadline of the VM resource at a certain time is $[t_{\min}, t_{\max}]$.
(2) Client group $C_g$:
  (a) The client number in the group is $n$.
  (b) The service request $S_{\text{cloud}}$ receives from unit client $C_g(p)$ $(1 \le p \le n)$ is $(S_{k,i,P}, b_p, t_p, P_P)$. $S_{k,i,P}$ denotes that the quantity demand for the $k$th VM resource the client $C_g(p)$ requires is $i$. $b_p$ is the network bandwidth demand of the client $C_g(p)$. $t_p$ is the competitive payments moment of the client $C_g(p)$. $P_P$ denotes the highest price the client $C_g(p)$ can afford as for the corresponding service request. It is noteworthy that $t_p$ must locate the competition deadline which $S_{\text{cloud}}$ announces, else the service request of $C_g(p)$ is invalid. Moreover, the final payment $P_{P'}$ must satisfy $P_{P'} \le P_P$.

Different from the auction mechanisms involved in the past references, the auction deadline is added into the model proposed in this paper, which is helpful to enhance the utilization rate of the VM resource from the cloud service provider. There is a type of clients called "zombie clients". Zombie clients often occupy the bandwidth resource and send the service requests to the cloud service provider, but the cloud service provider cannot get any response for a long time from them. As a result, the cloud service provider in this paper is able to announce and refresh the competition deadline about its own resource in the network in time, so that the zombie clients can be deleted. Of course, the network service quality, the profit of the cloud service provider in unit time, and the VM resource utilization rate can be enhanced accordingly.

Based on the model constructed in our proposed algorithm, the profit of the cloud service provider can be expressed as follows.

$$\sum_{p=1}^{n} P_{P'}\lambda_P - \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{p=1}^{n} C_{k,i,R}\lambda_P - \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{p=1}^{n} C_{k,i,I}(1-\lambda_P) \tag{5}$$

where $\lambda_P$ denotes the service response coefficient. If the cloud service provider accepts the service request from the client, $\lambda_P = 1$, else $\lambda_P = 0$.

Obviously, the client group $C_g$ should also satisfy the following requirements.

$$t_p \in [t_{\min}, t_{\max}] \tag{6}$$

$$\sum_{p=1}^{n} b_P \le B \tag{7}$$

### 3.1. Cloud property evaluation and VM configuration

The four dynamic resource description indexes including RI, DI, LI and CI proposed in Section 2 can be used to conduct the property evaluation on the cloud network, and the related results can provide scientific basis for the VM resource configuration. Based on the four indexes, we can obtain another composite indicator called performance index (PI) whose value is in the range [0,1]. The larger PI is, the less the available VM resource of the cloud service provider is, and the overall performance is worse. The mathematical expression of PI is as follows:

$$\mathrm{PI} = \begin{cases} 1, & \text{if } \mathrm{RI}_i = R^* \text{ or } \mathrm{DI}_i \ge D^* \text{ or } \mathrm{LI}_i \ge L^* \text{ or } \mathrm{CI}_i \ge C^* \\ \max\left[\dfrac{\mathrm{RI}_i}{R^*}, \dfrac{\mathrm{DI}_i}{D^*}, \dfrac{\mathrm{LI}_i}{L^*}, \dfrac{\mathrm{CI}_i}{C^*}\right], & \text{else} \end{cases} \tag{8}$$

where $\mathrm{RI}_i$, $\mathrm{DI}_i$, $\mathrm{LI}_i$, $\mathrm{CI}_i$ are the four indexes of the current VM resource. $R^*$, $D^*$, $L^*$, $C^*$ denote the largest quantities of the service request, request dealing, loading and the cost of the current VM resource can afford, respectively. Note that the four parameters in the max function required to be normalized in advance.

Eq. (8) indicates that, if any index of the four ones reaches the maximum, then PI is 1, which implies the VM resource in the cloud network is very scarce, otherwise the VM resource is still in surplus.

The cloud property evaluation and VM configuration are determined by both the service request of the client group and the cloud property indexes. The concrete steps are as follows.

*Inputs:* The cloud property index $\mathrm{PI}_i$, the total VM resource quantity of the cloud service provider is $N$, The running and the idle maintenance costs of the unit VM resource $S_i$ $(1 \le i \le N)$ are $C_{i,R}$ and $C_{i,I}$, respectively. The competition deadline of the VM resource at a certain time is $[t_{\min}, t_{\max}]$. The total network bandwidth resource is $B$. The service request the client $C_g(p)$ sending to the cloud service provider is $(S_{i,P}, b_p, t_p, P_P)$.

*Outputs:* The client group set $S_C$ auctioning successfully, and corresponding VM number $\mathrm{VM}_k$, $1 \le k \le K$.

*Initial state:* The after-sifting auctioning client group set $S_C = \Phi$, $\mathrm{VM}_k = 0$, $1 \le k \le K$. The competition deadline of the VM resource at a certain time is $[t_{\min}, t_{\max}]$. The parameter $q$, the individual counter of the after-screening auctioning client group, equals zero.

*Step1:* Compute the cloud performance index $\mathrm{PI}_i$.
*Step2:* $N = N \times (1 - \mathrm{PI}_i)$; $B = B \times (1 - \mathrm{PI}_i)$;
*Step3:* For ($p = 1$; $p \le n$; $p$++); $q$++;
*Step4:* If $t_p \le t_{\max}$
*Step5:* Delete the client $C_g(p)$
*Step6:* Else $C'_g(q) \leftarrow C_g(p)$
*Step7:* End if else; end for

*Step8:* Utilize the highest price $P$ announced in the client service request to filter the client group, and sort the surplus $q$ clients.

*Step9:* For ($x = 1$; $x \le q$; $x$++);

*Step10:* If $P_x < C_{k,i,R} - C_{k,i,I}$ or $b_x > B$ or $S_{i,P} > N$

*Step11:* Delete the client $C'_g(x)$.

*Step12:* Retain the client $C'_g(x)$.

*Step13:* Else $S_C = S_C \cup C'_g(x)$, $B = B - b_x$, $N = N - S_{i,P}$

*Step14:* End if else; end for

*Step15:* As for the final surplus $x$ successfully auctioning clients, conduct the resource distribution according to the highest price $P$ announced in the client service request.

During the above algorithm, the cloud service provider conducts three siftings in all. The first sifting happens at the auctioning moment, the main object is to deal with the possible issue that the interval between the order request moment and the final auctioning moment may be too long. In the past VM resource scheduling algorithms, the auction deadline often was not considered. It easily leads to the deadlock phenomenon that the zombie clients occupy the potential resource, but other hurry clients cannot apply for the resource successfully. The second sifting mainly focuses on the competitive payment. Since the cloud service provider has to face the maintenance cost of the VM resource, the point that the competitive payment of the client group must be higher than the maintenance cost becomes the premise the cloud service provider can offer the service. As a result, it is necessary to sift the prices the clients are willing to pay who have satisfied the requirements of the auction deadline. The third sifting is conducted towards the total resource quantity the cloud service provider can offer. If the client's request of the bandwidth or the number of the VM resource is larger than that the cloud service provider can offer, then this request must be denied. Moreover, by evaluating the cloud performance to obtain the cloud computing index PI, the rationally adjustment of the resource from the cloud service provider can be realized to further optimize the cloud service performance.

### 3.2. Client auction payment mechanism

After being an element of the client group set $S_C$, the client has to pay for the service offered by the cloud service provider subsequently. Of course, the client needs to take multiple factors into consideration, such as the service price another client agrees to pay who competes with me for the VM resource, and the average payment in the client group set $S_C$. In order to guarantee that the client is able to obtain the required resource and the cloud service provider has a good profit level, the client auction payment mechanism is given here. Suppose the number of the elements in the client group set $S_C$ is $x$, the concrete steps are as follows.

*Inputs:* The client group set $S_C$ auctioning successfully, the element in $S_C$ is $S_c (1 \le c \le x)$, the auction prices of $S_c$ is $P_C$ $(1 \le C \le x)$, the performance index $\mathrm{PI}_i$, the total quantity of the VM resource the cloud service provider has is $N$.

*Outputs:* The final price $P_{c'}$ the client auctioning successfully pays for.

*Step1:* For ($c = 1$; $c \le x$; $c$++).

*Step2:* As for the client $S_c$, $S_{C-c} = S_C - S_c$, compute the average payment of the set $S_{C-c}$.

*Step3:* Obtain the price $P_z$ the client $S_z$ who is the most serious rival towards $S_c$ pays for.

*Step4:* $P_{c'} = \max(P_{C-c}, P_z)$.

*Step5:* End for.

During the above steps, the cloud service provider does the research on the elements in the client group set $S_C$ auctioning successfully. The mechanism involves two aspects. One is the average payment level of $S_{C-c}$, the other is the price of the most serious rival towards the client. Obviously, the latter reflects the macro payment capacity of the client group, which can describe the payment level of whole group set to a considerable extent. Moreover, the most serious rival in the group set is the client most likely to obtain the corresponding VM resource except the client itself, so the rival's payment level should be the focus of the discussion. Naturally, in order to guarantee the client can get the VM resource and the profit level is acceptable simultaneously, the final payment price must be the maximum between the average payment level and the payment price of the most serious rival.

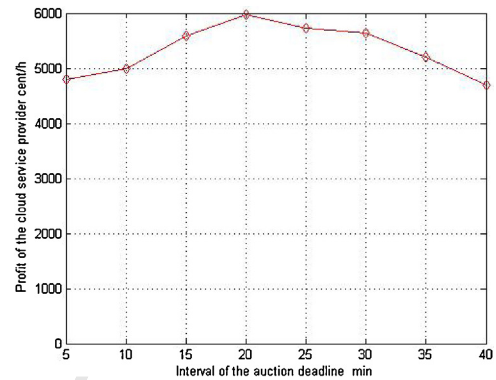


**Fig. 2.** Relation between the profit of the cloud service provider and the interval of auction deadline.

## 4. Experimental results and analysis

### 4.1. Experiment setups

In order to verify the effectiveness of the proposed algorithm, we choose the CloudSim software to complete the experiments. The platform is a PC whose configurations are as follows: 2.4 GHz CPU, 4 GB Memory, 512 GB hard disk.

As Ref. [20], the parameters setting of the experiments are as follows. The total VM resource quantity $N$ the cloud service provider offers is 40,000, the number of the clients denoted by $n$ is 20,000, the total of network bandwidth resource denoted by $B$ is 11 Mbit/s, the running and the idle maintenance costs of the unit VM resource are 0.5 cent/h and 0.15 cent/h, respectively.

Moreover, three typical algorithms have been chosen to compare with the proposed one, the detailed information of which is as follows.

**Algorithm 1 (A1).** Auction-based dynamic allocation and supply of the VM resource [20].

**Algorithm 2 (A2).** Fixed-price-based VM resource allocation.

**Algorithm 3 (A3).** VM resource dynamic scheduling algorithm.

It is noteworthy that the auction deadline mechanism is introduced into the proposed algorithm, so it is necessary to use the related software to fix the auction deadline $[t_{\min}, t_{\max}]$ of the VM resource in the course of the simulation experiments. In order to describe Algorithm 2 (A2) more objectively, we will offer two fixed values including 0.45 cent/h and 0.30 cent/h when conducting the experiments.

### 4.2. Simulation experiments

As for the proposed algorithm, the relation between the profit of the cloud service provider and the interval of auction deadline is shown in Fig. 2. It can be easily observed that, when the number of the clients is few, the higher the profit of the cloud service provider is, the larger the interval of the auction deadline is. The reason for the phenomenon lies in that these few clients are able to take much richer time and chances to apply for the VM resource. However,



Please cite this article in press as: W. Kong, et al., Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism, Optik - Int. J. Light Electron Opt. (2016), http://dx.doi.org/10.1016/j.ijleo.2016.02.061

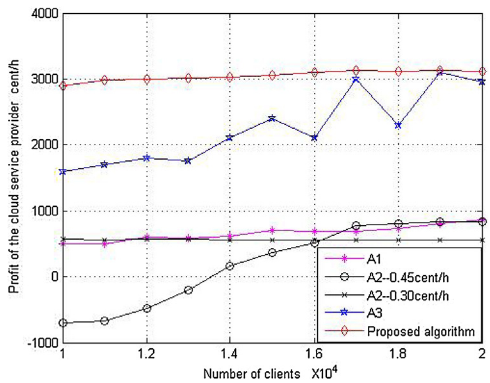


**Fig. 3.** Relation between the profit of the cloud service provider of the four algorithms and the number of the clients.

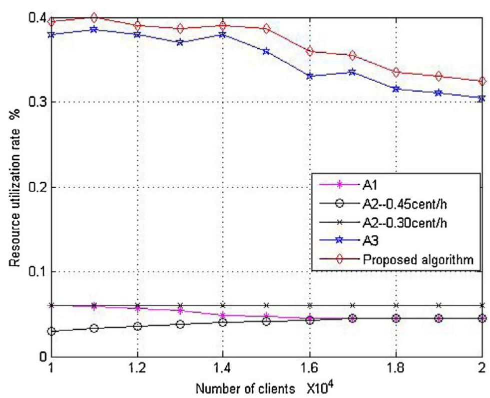


**Fig. 4.** Relation between the resource utilization rate of the four algorithms and the number of the clients.

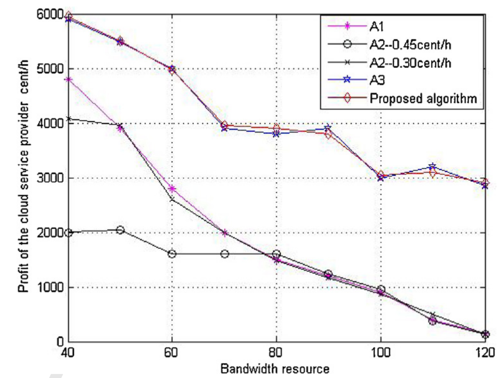


**Fig. 5.** Relation between the bandwidth resource and the profit of the cloud service provider.

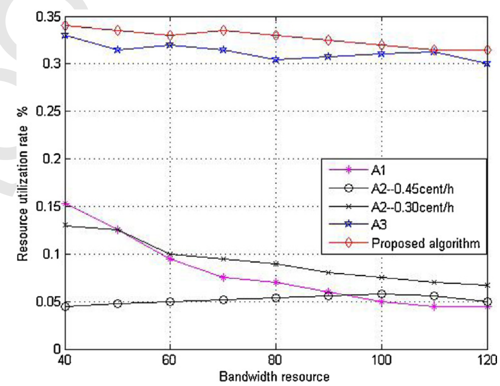


**Fig. 6.** Relation between the bandwidth resource and the resource utilization rate.



with the number of the clients increasing, there may be some clients locating at the application queue so long cannot pay for the price in time, so large auction deadline interval will have a negative impact on the profit of the cloud service provider to some extent. According to the final experimental results, it is appropriate to set the auction deadline interval denoted by $t_{max} - t_{min}$ to be 20 min. As a result, the default auction deadline interval is 20 min in this paper unless otherwise noted.

The relation between the profit of the cloud service provider of the four algorithms including the proposed one and the number of the clients is shown in Fig. 3. The profit trends of the cloud service provider between Algorithm 1 (A1) and Algorithm 2 (A2) whose fixed price is 0.30 cent/h are very approximate. The curves are stable and gentle, reflecting good robustness and network adaptivity. In comparison, the trend of the profit at the fixed price 0.45 cent/h is sharply different form that at the price 0.30 cent/h especially when the number of the clients is less than 14,000, the reason for which is that the number of the clients who are willing to afford the price 0.45 cent/h is too few so that the idle maintenance cost of the VM resource is too high. Due to the introduction of the critical payment in Algorithm 3 (A3), the variation extent of the profit of the cloud service provider is sharp. Oppositely, by taking the average payment level of the client group and the price of the most serious rival into consideration, the fluctuation of the proposed algorithm is much smoother than that of Algorithm 3 (A3). Furthermore, the proposed algorithm enhances the overall profit of the cloud service provider.

The relation between the resource utilization rate of the four algorithms and the number of the clients is reported in Fig. 4. In the case of the fixed price 0.45 cent/h in Algorithm 2 (A2), the utilization rate tends to be influenced by the number of the clients. When the number of the clients is comparatively few, the utilization rate is also low, whereas the utilization rate will increase and keeps stable. When the fixed price is 0.30 cent/h, the utilization rate of Algorithm 2 (A2) always remains stable. Algorithm 1 (A1) is obsessed with the high price the clients pay that often require enormous demand for VM resources, so the utilization rate of the cloud service provider will decline with the number of the clients increasing. Algorithm 3 (A3) places great emphasis on the factor of the bandwidth resource, so if there are only few clients, the cloud service provider will supply the VM resource to the client with low demand for bandwidth. At the beginning, the utilization rate of the cloud service provider has a tendency to increase. However, with the number of the clients increasing, the clients with higher demand for bandwidth resources, VM resources, and higher prices are more and more, so the utilization rate of the cloud service provider will be a downward trend in the late. The proposed algorithm in this paper has taken multiple factors including the bandwidth, payment prices and the auction deadline into consideration, so its utilization rate of the cloud service provider is much higher than that of other three algorithms.

Fig. 5 describes the relation between the profit of the cloud service provider and the bandwidth resource. Fig. 6 shows the relation between the utilization rate of the cloud service provider and the bandwidth resource.

Similar to the phenomena in Fig. 3, Algorithms 1 (A1) and 2 (A2) in Fig. 5 still show good robustness, but the curve fluctuation extent of Algorithm 3 (A3) is sharper. It is noteworthy that the profit trends of the cloud service provider in four algorithms all have a tendency to decline, the reason for which is that the enhancement of the bandwidth demand of the clients necessarily results in the decline of the distributable VM resource; likewise, the profit will also be affected. The factor of the bandwidth resource is both taken into

Please cite this article in press as: W. Kong, et al., Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism, Optik - Int. J. Light Electron Opt. (2016), http://dx.doi.org/10.1016/j.ijleo.2016.02.061

account in Algorithm 3 (A3) and the proposed algorithm, so they are prone to replying to the service requests with lower demand for bandwidth resources, fewer VM resources and higher payment prices.

In Fig. 6, with the bandwidth resource increasing, the resource utilization rates of the cloud service provider of the fur algorithms all have a tendency to decline, the reason for which is that the bandwidth demand of the clients increase. Moreover, if the bandwidth $b \leq 90$ kbit/s, the resource utilization rates of the cloud service provider at the fixed price 0.45 cent/h in Algorithm 2 (A2) still has a tendency to increase, because the current total of bandwidth resource demand is still not saturated. Moreover, the resource utilization rate of the cloud service provider of the proposed algorithm is larger than that of other three ones. The fundamental reason is that the proposed algorithm has taken multiple factors including the bandwidth, payment prices and the auction deadline into account.

From the above, compared with other algorithms, the proposed one has remarked priorities in terms of the service quality of the cloud environment, the profits of cloud service providers and the resource utilization rate of VM.

## 5. Conclusions

Based on the traditional VM resource scheduling algorithms in cloud computing, an adaptive VM resource scheduling algorithm is proposed in this paper, which takes the network bandwidth resource, the auction deadline and the payment prices into consideration. Experimental results indicate that the proposed algorithm has remarked priorities in terms of the service quality of the cloud environment, the profits of cloud service providers and the resource utilization rate of VM. How to further optimize the performance of the algorithm will be the focus of our future work.

## Acknowledgments

The authors thank all the reviewers and editors for their valuable comments and works. The work was supported in part by the National Natural Science Foundation of China under Grant 61309008, 61309022 and 61473237; Natural Science Foundation of Shannxi Province of China under Grant 2014JQ8049; Foundation of Science and Technology on Information Assurance Laboratory under Grant KJ-15-102; Natural Science Foundations of the Engineering University of the Armed Police Force of China under Grant WJY-201414.

## References

[1] A. Iosup, S. Ostermann, M.N. Yigitbasi, et al., Performance analysis of cloud computing services for many-tasks scientific computing, IEEE Trans. Parallel Distrib. Syst. 22 (6) (2011) 931–945.
[2] A.S. Prasad, S. Rao, A mechanism design approach to resource procurement in cloud computing, IEEE Trans. Comput. 63 (1) (2014) 17–30.
[3] J. Li, X. Tan, X.F. Chen, et al., OPoR: enabling proof of retrievability in cloud computing with resource-constrained devices, IEEE Trans. Cloud Comput. 3 (2) (2015) 195–205.
[4] J.W. Lin, C.H. Chen, J.M. Chang, QoS-aware data replication for data-intensive applications in cloud computing systems, IEEE Trans. Cloud Comput. 1 (1) (2013) 101–115.
[5] D. Bruneo, A stochastic model to investigate data centre performance and QoS in IaaS cloud computing systems, IEEE Trans. Parallel Distrib. Syst. 25 (3) (2014) 560–569.
[6] P.B. Si, Q. Zhang, F.R. Yu, et al., QoS-aware dynamic resource management in heterogeneous mobile cloud computing networks, Chin. Commun. 11 (5) (2014) 144–159.
[7] G. Baranwal, D.P. Vidyarthi, A fair multi-attribute combinatorial double auction model for resource allocation in cloud computing, J. Syst. Softw. 108 (10) (2015) 60–76.
[8] H. Zhao, X.X. Liu, X.L. Li, Towards efficient and fair resource trading in community-based cloud computing, J. Parallel Distrib. Comput. 74 (11) (2014) 3087–3097.
[9] M.H. Eawna, S.H. Mohammed, E.S.M. El-horbaty, Hybrid algorithm for resource provisioning of multi-tier cloud computing, Procedia Comput. Sci. 65 (2015) 682–690.
[10] Z. Yang, C.Q. Yin, Y. Liu, A cost based resource scheduling paradigm in cloud computing, in: Proceedings of the 12th International Conference on Parallel and Distributed Computing, vol. 1, (15), 2011, pp. 417–422.
[11] K. Zhu, H.G. Song, L.J. Liu, Hybrid genetic algorithm for cloud computing applications, IEEE Asia-Pacific Serv. Comput. Conf. 1 (2011) 182–187.
[12] X.S. Yang, A new metaheuristic bat-inspired algorithm, Science 284 (2010) 65–74.
[13] L.Y. Zuo, Z.B. Cao, S.B. Dong, Virtual resource evaluation model based on entropy optimized and dynamic weighted in cloud computing, J. Softw. 24 (8) (2013) 1937–1946 (In Chinese).
[14] S.J. He, L. Guo, Y.K. Guo, et al., Elastic application container: a lightweight approach for cloud resource provision, in: Proc of the 2012 IEEE 26th Int'l Conf. on Advanced Information Networking and Applications (AINA), IEEE, 2012, pp. 15–22.
[15] A. Verma, L. Cherkasova, R.H. Campbell, Resource provisioning framework for MapReduce jobs with performance goals, Lect. Notes Comput. Sci. 70 (9) (2011) 165–186 (2011).
[16] Z.H. Gong, X.H. Gu, J. Wlkes, PRESS: predictive elastic resource scaling for cloud systems, in: Proc of the 2010 Int'l Conf. on Network and Service Management (CNSM), 2010, pp. 9–16.
[17] Z.J. Hu, The Resource Availability Evaluation in Service Grid Environment for QoS, Central South University, Changsha, 2010.
[18] N. Chohan, C. Castillo, M. Spreitzer, et al., See spot run: using spot instances for mapreduce workflows, in: Proc of the 2nd USENIX Conference on Hot topics in Cloud Computing, 2010.
[19] S. Zaman, D. Grosu, Combinatorial auction-based dynamic VM provisioning and allocation in clouds, in: Proc of the 3rd IEEE International Conference on Cloud Computing Technology and Science (CloudCom), 2011.
[20] S. Zaman, D. Grosu, Combinatorial auction-based allocation of virtual machine instances in clouds, J. Parallel Distrib. Comput. 73 (4) (2013) 495–508.
[21] X. Liu, C.W. Yuan, Z. Yang, et al., VM dynamic scheduling algorithm for mobile cloud computing, Syst. Eng. Electron. 37 (9) (2015) 2176–2181.