ARAM: A New Auction-based Resource Allocation Model in Cloud Computing

Nafiseh Sharghivand* and Farnaz Derakhshan

Computer Engineering Department, Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran; n.sharghivand@tabrizu.ac.ir, derakhshan@tabrizu.ac.ir

Abstract

Cloud computing is an on-demand network access model to a shared pool of configurable computing resources. One of the main challenges in cloud computing is the efficiency of the pricing and resource allocation models adopted by cloud providers. In this paper, we proposed a new resource allocation model called *ARAM*, which is based on reverse auctions and provides dynamic pricing policies. In this model, independent reverse auctions are hold for each cloud service request. ARAM has been compared with other previously proposed approaches, considering several issues such as the types of the services it supports, the necessity for configuring service instances, and also those related to load balancing. The main advantage of ARAM, compared to previous forward auction based models, is that it eliminates the necessity for configuring service instances. Therefore, the utilities of both consumers and providers are enhanced, in terms of cost and time. Also by adopting a dynamic pricing approach, it moderates the shortcomings of fixed pricing strategies in cloud computing, such as resource wastage and lack of fairness. Furthermore, contrary to most of the previous approaches, ARAM supports dynamic pricing for all types of services and considers load balancing issues in allocating resources. Today, most of the providers in the industry use fixed pricing models ignoring the market condition. Therefore, it seems necessary to use efficient dynamic pricing and resource allocation models, such as ARAM, in cloud computing.

Keywords: Auctions, Cloud Computing, Dynamic Pricing, Resource Allocation

1. Introduction

Cloud computing is a new emerging technology which has attracted the attention of both the research community and the business industry. The main goal of cloud computing is to provide different computing resources as a variety of services for customers. With the recent advancements in information and communications technology, computing is slowly being recognized as the 5th utility, beside utilities such as, electricity, natural gas, water, and sewage¹.

Today, most of the cloud providers have employed fixed pricing strategies, i.e., they don't change their service prices according to different market conditions. Fixed pricing has drawbacks such as resource wastage, lack of fairness and independency to market conditions. Amazon is one of the rare examples that have used dynamic pricing to increase its revenue and resource utilization in its

*Author for correspondence

spot instances². Hence, the necessity of applying dynamic pricing strategies in resource allocation mechanisms of cloud computing has opened a new field of research.

Resource allocation and reaching Service-Level Agreements (SLAs) in cloud computing has also been one of the interesting research topics. In³ a comprehensive agent based cloud computing model has been proposed which includes a mechanism for allocating resources and reaching SLAs. This model also benefits from dynamic pricing strategies. In a new method for reaching SLA considering market conditions, agents apply time-dependent strategies to make new offers in each round of negotiation. The concession rate in these strategies is set according to the bargaining position of each agent⁴. Time-dependent strategies have also been adopted among consumer, broker, and service provider agents, without considering the competitions existing in the market⁵. In another work, a Price-and-Time-Slot Negotiation (PTN) mechanism is adopted which allows cloud consumers and providers to negotiate on price and time-slot preferences⁶. Furthermore, a new solution is proposed to automate the negation process between cloud providers and consumers, specifically for IaaS category. In this method, providers consider their resource utilization amount in their offers and as a result concede more on the price of less utilized resources⁷.

Auction-based methods have also been widely used in applying dynamic pricing and resource allocation in cloud computing, such as a Multi-Unit Combinatorial Auction (MUCA) method for a single provider⁸. In another work, a resource procurement approach is proposed suggesting three possible mechanisms, which are: Cloud-Dominant Strategy Incentive Compatible (C-DSIC), Cloud-Bayesian Incentive Compatible (C-BIC), and Cloud Optimal (C-OPT). In C-DSIC and C-BIC, the provider with the lowest cost per unit QoS is declared as the winner. While, in C-OPT the provider with the least virtual cost is the winner⁹. An intelligent economic approach for dynamic resource allocation (IEDA) has also been proposed, in which an improved combinatorial double auction protocol is devised for trading various types of resources among consumers and providers, targeting only IaaS¹⁰. Cloud Market Maker (CMM)¹¹, which is an auction-based approach, provides a dynamic unified view of different cloud offerings from different providers for cloud consumers in real-time. This unified view can help the consumers to select an appropriate provider. It also allows providers to apply a dynamic pricing model, which results in provider's profit enhancement.

In the case of reducing energy consumption in cloud data centers, new approaches have been proposed which use proper resource provisioning method in cloud environments¹². It has also been shown that how efficient resource allocation strategies can optimize time, cost, power consumption and load balancing in cloud environments¹³. Meta-heuristic algorithms have been also widely used for resource allocation in IaaS cloud¹⁴.

In this work, we propose a new model for applying dynamic pricing strategies in resource allocation of cloud computing, which we called it: *ARAM* (Auction-based Resource Allocation Model). In this model, the service requests of consumers are submitted to broker agents. Each broker agent runs an independent reverse auction for each request. Providers can choose to participate in auctions based on their own criteria and preferences.

ARAM provides a dynamic pricing scheme for all types of services. The main advantage of ARAM, compared to forward auction based models, is to eliminate the need for configuring service instances (This type of configuration imposes high overheads on both providers and consumers in terms of time and cost). The other advantage of this model is that by considering the amount of resource utilization in providers, the profit of both parties increase. It also enables load balancing management in the data centers of providers.

By this introduction, this paper is structured as follows: Section 2 provides a background on this topic. Then, in Section 3, ARAM, our new auction-based model (proposed in this paper for dynamic pricing) is presented. After that in Section 4, we provide an evaluation and discussion over previous works and our proposed model. Finally, Section 5 presents the conclusion and future work.

2. Background

The complexity of cloud resources allocation and the variable nature of cloud requests, accentuate the need to apply resource management systems that are capable of allocating cloud resources and pricing its services efficiently and automatically. In this section, we explain how using multiagent systems can be helpful in automation of the cloud resources allocation and adopting dynamic pricing techniques.

2.1 Cloud Computing

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources¹⁵, which has so many advantages such as, high computing power, cost effectiveness, scalability, and availability. Cloud computing provides a wide variety of services. In general, these services can be classified into three service models¹⁶:

- Infrastructure as a Service (IaaS) which provides virtualized computing resources (e.g., Amazon's EC2).
- **Platform as a service (PaaS)** which provides a platform for application development (e.g., Google App Engine).
- Software as a service (SaaS) which provides access to software applications (e.g., Salesforce)

These services are accessed via a network, usually the Internet, by a broad group of users or clients. In such a

working environment, most of the times providers and consumers need to reach an agreement on different issues. In order to reach such an agreement, providers and consumers need to consider several parameters in their negotiations, such as, the requirements of consumers, the budget of consumers, the available resources of providers, and the acceptable price ranges of providers. The agreement obtained by providers and consumers is called Service-Level Agreement (SLA)¹. The providers are committed to provide services for clients based on this SLA, and the clients cannot claim something more than what they have previously agreed upon (unless the two parties try to reach a new agreement). If the parties violate the SLA, they receive a penalty.

2.2 Multiagent Systems

Multiagent Systems (MAS) have shown great success in applying to those systems consisting interactions among their components, which are too hard to be manually managed and can be considered as the bottleneck of the system¹⁷. Cloud computing can be one of those examples where the providers and consumers need to interact with each other, in order to reach an agreement, so that the services can be served by the provider to the consumer based on this signed contract. Hence, since in a cloud system the number of providers, consumers and the parameters to reach an agreement upon may be high, using MASs to automate the interactions seems a suitable choice. One of the negotiation protocols used by agents in a multiagent system is the auctions, which is explained in the next section¹⁷.

2.3 Auctions

Auctions are very simple interaction scenarios and a powerful tool for allocating goods, tasks, and resources by automated agents¹⁷.

An auction takes place between an agent known as auctioneer and a set of other agents known as bidders. One type of auctions is forward auction, in which the seller is the auctioneer and the buyers are the bidders. However, sometimes in the auctions, the auctioneer and the seller are two separate entities. In such cases, the auctioneer is responsible from the seller to sell its good in an auction. The bidders bid for the goods in the auction and based on the type of the auction, the winner and the price are determined.

Another type of auctions is reverse auction. In reverse auction, the buyer is the one playing the role of the auc-

tioneer and the sellers are the bidders. Here also the auctioneer and the buyer may be two separate entities. In reverse auctions, the auctioneer is responsible from the buyer to acquire its needed good in an auction. Thus, the bidders bid for the price they are willing to receive to provide the good. Reverse auctions-also known as procurement auctions- are preferred over other auctions for procuring resources because they reduce obtaining costs.

2.4 Auction-based Cloud Computing

A cloud is a distributed computing system which provides services for its clients. Since the cloud's available resources, services, and its user's requirements, are constantly changing, the environment can be considered as a dynamic environment. Also, in a cloud computing system, the cloud providers and users need to interact with each other in a way that the interest of both parties, are considered.

As mentioned in the previous subsection, auctions are simple interaction scenarios which can be easily modeled. Also, regarding the continuous increase in the number of cloud providers and the competition among them, and the shortcomings of fixed-pricing strategies, using dynamicpricing models seems necessary. Thus, auction-based models can be used to implement dynamic pricing strategies in cloud computing. Also, reverse auctions, based on their usage in resource procurement applications, can be considered as a suitable methodology for resource allocation in cloud computing.

3. Our Proposed Model (ARAM)

In a cloud computing system, providers need to allocate their resources as different services to cloud consumers efficiently, so that they can maximize their overall profit. To achieve this, the providers should consider a number of issues such as their available resources, the load balancing condition in their data centers, and the current market condition. Indeed, when a cloud provider receives a new service request from a consumer, it should first consider whether it has enough available resources to fulfill this requirement or not. If there were enough available resources, it should then try to maximize the amount of profit it can acquire by suggesting a reasonable price.

In general, when the demand for a service is high (respectively low) in the market, it is better for the provider to decrease (respectively increase) the service price. This will not only help the provider to stay in the competitive cloud market, it also helps the cloud consumers to acquire their requested services with better prices. Also the provider can consider a higher priority for those requests resulting in better load balancing in its data center.

The manual management of the resource allocation process in cloud computing, considering the issues discussed above, if not impossible, can be really challenging and considered as the bottleneck of the system. Hence, using new techniques to automate this process seems irresistible. Since MASs have been widely and successfully applied in automation of different processes in distributed systems¹⁷, inspired by them, we have applied a reverse auction technique in a multiagent environment to automate the resource allocation process in cloud computing.

If we desire to have a successful MAS, it should be designed properly both in micro (agent) and macro (society) levels. Hence, each agent not only should be capable of autonomous, flexible action in its environment to meet its design objectives, it should be able to interact with other agents in the environment effectively¹⁷.

Our proposed model, which we called, an Auctionbased Resource Allocation Model (ARAM), includes three types of agents: Consumer Agents (CAs), Broker Agents (BAs), and Provider Agents (PAs). In ARAM, which its architecture is shown in Figure 1, CAs and PAs act on behalf of cloud consumers and providers respectively, and the auctions are taken place by BAs.

CAs which are acting on behalf of consumers, try to acquire the requested service of their related consumers, with the most appropriate price. Each cloud consumer has a single CA related to it. PAs are the agents responsible from providers to allocate their resources in terms of different services such that bring more profit for them. Each cloud provider has also a single PA related to it. BAs act as intermediates between these two types of agents. They hold different auctions for each service request, providing the opportunity for CAs and PAs to increase their utility.

Using multiple BAs instead of one BA in the system, leads to a competition among BAs and avoids any collusion between BA and PAs. Because, in the case that we have one broker in the system, the broker and the providers may collude with each other to provide the requested service of the consumer with its initial proposed price. In this way, they guarantee more profit for themselves. Also, one single BA in the system can be considered as a single point of failure. Because if for any reason the BA does not act properly, it will effect on the performance of the overall system.

Using CAs in ARAM saves cloud consumers from the burden of finding a suitable service provider. Also, by using of PAs, the cloud providers do not involve themselves with



Consumer i

Figure 1. Our ARAM model architecture.

the complicated task of resource allocation and selecting appropriate service requests to fulfill. Indeed, in a nonautomated cloud environment, each cloud consumer has to investigate different services provided by different cloud providers manually, so that it can find a cloud service which meets its requirements the most. Of course, considering the large number of cloud providers and the vast variety of services they provide, finding and selecting an appropriate cloud provider is not that easy. Hence, manual finding and selection of a cloud provider usually does not end in a good result. Also, in such environments, providers usually adopt fixed pricing strategies so that they can save themselves from the complexities related to market condition and load balancing issues. However, this way they deprive themselves of all the advantages of dynamic pricing strategies.

After service request submission by any consumer, the interaction protocol between agents of the system has the following steps:

- The CA related to the consumer submits the service request to BAs.
- Each BA runs an independent auction for that specific request.
- The PAs with enough free resources bid on auctions.
- The Bas determine the winners.
- The BAs announce the result to the CA and the PAs.

The detailed description of the interaction protocol is as follow:

Stage 1 (*announcement*): After request submission by any consumer, the related CA is charged to submit this request to a number of BAs. The submitted request *i*, includes three values: the service (s_i) needed by the consumer, the maximum price (p_i) the consumer is willing to pay for this service, the time (t_i) by when the consumer needs the service to be deployed and is considered as the deadline of the auction.

Stage 2 (*running an auction*): When a new request is submitted to a BA, the BA starts to run a separate auction for that request. In this auction, the BA whom plays the role of the auctioneer announces the auction to PAs. The steps followed by a BA to hold auctions, is illustrated in Algorithm 1.

Stage 3 (*bidding*): PAs investigate the announced auction and in the case of having enough free resources, submit a preliminary bid (*pb*). If the preliminary bid (*pb*_i) of PA_i is lower than p_i (*pb*_i < p_i) and the lowest bided price in the auction, PA_i gets the permission to take part in the

auction. After being permitted, the dominant strategy for each PA in the auction, is to bid a price a little lower than the lowest bid (lb), until it reaches its reserve price. Reserve price is the price that the PA suffers a loss, if it wins the auction with a price lower than it. If the lowest price bided in the auction gets lower than the reserve price of PA, that PA leaves the auction. Also, if a PA delivers an auction announcement that it may receive more profit by participating in it, it can leave the current auction and it should pay a penalty in the case that its biding price is at the moment the lowest bided price.

Stage 4 (*winner determination*): By reaching the auction to its deadline, the PA with the lowest bided price is determined as the winner by the broker. Also the consumer should pay a price equal to the winner price to acquire its requested service from the winner PA.

Stage 5 (*results announcement*): If the auction has a winner, i.e., at least one PA is ready to provide the requested service with a price lower than the initial price of consumer, the BA announces the CA that the winner PA provides its requested service. Hence, the consumer should pay a price equal to the winner price to the winner PA. Otherwise, the BA announces the CA that no PA is willing to provide the requested service with a price equal or lower than the its initial price.

In stage 3, while PAs investigate different auctions to take part in, they can consider the utilization amount of their different resources. Therefore, in addition to selecting those auctions resulting in the balance of the load in their data centers, they can bid with lesser prices on their less utilized resources to achieve more profit. Since, it is obvious that, it is more affordable for providers to lease their resources with lesser prices than keeping them unutilized. In this way, the consumers will also get the opportunity to acquire their requested services with cheaper prices.

Algorithm 1. The algorithm of a Broker Agent (BA) in ARAM

inputs: req_i /*The submitted request *i* */

 s_i /*The service type asked in request i */

 $p_{\rm i}~$ /*The maximum price the consumer is willing to pay for request i */

 $t_{\rm i}~/^*$ The time limit determined by the consumer for request $i~^*/$

pb /*preliminary bid submitted by PAs */

for each new service request req(*i*, s_i , p_i , t_i)**do** /* *Auction initialization* */ auction.ID \leftarrow req(*i*)

auction.SERVICE \leftarrow req(s_i)

auction.LB \leftarrow req (p_i) /* lowest bided price in the auction */

auction.DEADLINE $\leftarrow \text{req}(t_i)$

Announce the auction to PAs;

while ¬REACHED?(auction.DEADLINE)

subscribedPAs \leftarrow Subscribe new PAs with pb<auction.LB bids \leftarrow GetBids(subscribedPAs); /*Get all new submitted bids from all PAs in the auction*/

auction.LB \leftarrow MIN(bids)

$if {\it Current Winner Leave then}$

Consider a penalty for the winner PA because of its left

auction.LB \leftarrow 2ndMIN(bids)

end-if

end-while

Announce the end of auction to looser PAs

Ifany Winnerthen

Announce the winner PA it has won the auction

Announce the CA its requested service is provided by the winner PA with the price *LB*

else

```
Announce failure to CA
```

end-if

end-for

4. Evaluation and Discussion

In this section, we discuss about our proposed model and compare it with previous proposed models from different perspectives. We also evaluate our model and highlight its main advantages.

Similar to previous works, our proposed model employs dynamic pricing strategies to overcome fixed-pricing strategies shortcomings. However, the main contribution of this model is to eliminate the need for configuring service instances usually required in auction-based models. It also considers the amount of resource utilization in cloud providers' data centers to manage load balancing issues and enhance the utility of both consumers and providers. The amount of utilized resources has only been considered in⁷, which is only suitable for IaaS category. While, in our proposed model, all types of services provided by cloud vendors are supported.

In our proposed model, the shortcomings of the previous works are targeted and tried to overcome. In order to have better evaluation we present Table 1, which shows the comparison of the existing methods with our proposed model considering different features including: Supporting from multiple cloud providers and types of services, needing to configure service instances, considering resource utilization parameters, and using auction-based approaches.

One of the main features of all these models is that they all follow a dynamic pricing strategy. Indeed, supporting dynamic pricing strategies was the main goal of all these models. Also, most of the investigated models consider multiple cloud providers in their architecture. It means that multiple cloud providers have the opportunity to lease their resources and acquire income. The only model that lacks this feature is the model proposed in⁸, in which only a single provider (e.g. Amazon), presents its service instances in an auction and consumers can then bid for any number of them in a combinatorial auction paradigm. Considering multiple providers can result in a better competition among different providers, since they will all try to provide better services with lesser prices to gain more customers.

Supporting different types of services is the other important feature that provides a dynamic pricing paradigm for all types of services.

One of the main disadvantages of forward auction based models is that they all need to configure service instances. Indeed, when providers want to present their service instances in a forward auction, they have to analyze the market and requests for different services, so that they can present instances more compatible with market demands. Hence, along with all over heads providers will have to tolerate to analyze, at the end the requirements of the customers are not matched exactly with the available offers of the market. Hence, the consumers will also have to tolerate another overhead which is to buy something more than their real needs with higher costs.

Using reverse auction technique in ARAM, the analyzing phase is eliminated which is one of the main advantages of our model. Since, the analyzing phase, in addition to complexity, imposes high costs to cloud providers in terms of time and cost. This is because in order for the cloud providers to configure their service instances according to market demands, they need to investigate the current service requests carefully, which is an extremely costly and time-consuming job. Further, this way the cloud consumers are able to spend exactly equal to the cost of their own requested services. In previous forward

Model	Multiple cloud providers	Types of services	Configuring service instances	Resource utilization parameters	Auction-based approach
Agent-based cloud computing ³	\checkmark	All types	×	×	×
Complex and concurrent negotiations on preferred time-slots and price ⁶	\checkmark	All types	×	×	×
Agent-based Intercloud economic model⁴	\checkmark	All types	×	×	×
An autonomous time- dependent SLA negotiation strategy ⁷	\checkmark	IaaS	×	\checkmark	×
Multi-unit combinatorial auction ⁸	×	All types	\checkmark	×	\checkmark
C-DISC, C-BIC, C-OPT ⁹	\checkmark	All types	\checkmark	×	\checkmark
IEDA ¹⁰	\checkmark	IaaS	✓	×	\checkmark
Cloud Market Maker ¹¹	\checkmark	IaaS	\checkmark	×	\checkmark
ARAM: Our proposed model	\checkmark	All types	×	\checkmark	\checkmark

Table 1. The comparison of existing models and ARAM

auction based methods, the consumers have to choose one of the service instances available in the market, which matches its requirements the most.

The amount of resource utilization in data centers and load balancing issues are one of the main challenges that cloud providers are faced in managing their resources. Therefore, in a cloud computing environment, it may be desirable for cloud providers to offer more attractive prices for their less utilized resources. Since, leasing the resources with cheaper prices can be more economical than keeping them unutilized (something similar to Amazon spot instances²). Also the providers can assign the priority to participate in those auctions that winning in them can result in more load balancing in their data centers. As it can be seen in the table, resource utilization related issues has only been considered in⁷ previously and one of the main drawbacks of this model is that it only supports IaaS.

It can be seen that our proposed model enhances previous works from several aspects. It considers multiple providers in its architecture to increase the competition among providers to provide better services. It also allows consumers to acquire their required services, including all types of services, with lower costs. One of the main advantages of the proposed model is that it eliminates the need to configure the service instances by providers, which can impose high extra costs to both providers and consumers. Resource utilization and load balancing issues are other important features that have been considered in our model increasing the utility of both providers and consumers.

5. Conclusion and Future Work

Today, the large number of cloud providers, the dynamicity of the cloud environment, and the variety of consumer requirements, demands adopting new dynamic pricing strategies to overcome the problems of fixed pricing strategies in existing cloud systems. These problems include, lack of fairness, resource wastage, and ignoring of market conditions. Although, most of the cloud providers in the industry still use fixed pricing strategies, systems are usually biased toward the benefits of providers. To overcome all these drawbacks of fixed pricing strategies, this paper presented a new dynamic pricing model based on a reverse auction approach.

In our proposed model, the service requests of consumers are submitted in Broker Agents (BAs) by their related Consumer Agents (CAs). Then, each BA runs an auction for each request separately and the providers investigate different auctions to choose the most suitable one to participate in. The process of selecting a suitable auction and bidding in it is all done by the Provider Agents (PAs). While selecting an auction, the PAs can take few issues such as the load balancing condition in their data centers and the amount of the utilization of their resources into account. Considering such issues not only results in utility enhancement of providers and consumers, also leads into a more efficient management of resource utilization of providers. By the end of an auction, its result is announced by the BA to the CA and the winner PA. The CA should pay a price equal to the winning price to the winner PA to acquire its requested service.

Our proposed model eliminates the demand for configuring service instances by cloud providers, usually required in auction based methods, resulting in both parties utility enhancement. It supports all types of services and results in better management of the workloads in data centers.

We will implement the proposed model and investigate the enhancement in the utility of cloud providers and consumers in numerical terms, as our future works. Also, we will investigate the effect of considering resource utilization of service requests on load balancing issues in more detail. The QoS issues will also be considered in our future works.

6. References

- 1. Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems. 2009; 25(6):599–16.
- 2. Amazon Web Services. Available from: https://en.wikipedia. org/wiki/Amazon_Web_Services
- Sim KM. Agent-based cloud computing. IEEE Transactions on Services Computing. 2012; 5(4):564–77.
- 4. Sim KM. Agent-based interactions and economic encounters in an intelligent intercloud. IEEE Transactions on Cloud Computing. 2015; 3(3):358–71.
- 5. Siebenhaar M, Nguyen TAB, Lampe U, Schuller D, Steinmetz R. Concurrent negotiations in cloud-based systems. Proceedings of the 8th International Conference on Economics of Grids, Clouds, Systems, and Services; Paphos, Cyprus. 2012. p. 17–31.

- Sim KM. Complex and concurrent negotiations for multiple interrelated e-Markets. IEEE Transactions on Cybernetics. 2013; 43(1):230–45.
- Dastjerdi AV, Buyya R. An autonomous time-dependent SLA negotiation strategy for cloud computing. The Computer Journal. 2015; 58(11):3202–16.
- 8. Wang Q, Ren K, Meng X. When cloud meets eBay: Towards effective pricing for cloud computing. 2012 Proceedings IEEEINFOCOM; Orlando, FL. 2012. p. 936–44.
- 9. Prasad AS, Rao S. A mechanism design approach to resource procurement in cloud computing. IEEE Transactions on Computers. 2014; 63(1):17–30.
- Wang X, Wang X, Che H, Li K. an intelligent economic approach for dynamic resource allocation in cloud services. IEEE Transactions on Cloud Computing. 2015; 3(3):275–89.
- Javed B, Bloodsworth P, Rasool RU, Munir K, Rana O. Cloud market maker: An automated dynamic pricing marketplace for cloud users. Future Generation Computer Systems. 2016; 54:52–67.
- Jayapandian N, Zubair Rahman AMJ, Gayathri J. The online control framework on computational optimization of resource provisioning in cloud environment. Indian Journal of Science and Technology. 2015 Sep; 8(23). DOI: 10.17485/ijst/2015/v8i23/79313.
- Shyamala K, Rani TS. an analysis on efficient resource allocation mechanisms in cloud computing. Indian Journal of Science and Technology. 2015; 8(9):814–21.
- Madni SHH, Abd Latiff MS, Coulibaly Y, Abdulhamid SM. An appraisal of meta-heuristic resource allocation techniques for IaaS cloud. Indian Journal of Science and Technology. 2016 Jan; 9(4). DOI: 10.17485/ijst/2016/ v9i4/80561.
- Mell P, Grance T. The NIST definition of cloud computing. USA: National Institute of Standards and Technology; 2011.
- Hwang K, Dongarra JJ, Fox GC. Distributed and cloud computing: from parallel processing to the Internet of Things. USA: Morgan Kaufmann Publishers Inc; 2011.
- 17. Wooldridge M. An introduction to multi agent systems. USA: John Wiley and Sons Publishing; 2009.