Accepted Manuscript

Cloud Market Maker: An automated dynamic pricing marketplace
for cloud users

Barkha Javed, Peter Bloodsworth, Raihan Ur Rasool, Kamran Munir,
Omer Rana

PII: S0167-739X(15)00205-8
DOI: http://dx.doi.org/10.1016/j.future.2015.06.004
Reference: FUTURE 2782

To appear in: *Future Generation Computer Systems*

Received date: 8 June 2014
Revised date: 5 June 2015
Accepted date: 9 June 2015

Please cite this article as: B. Javed, P. Bloodsworth, R.U. Rasool, K. Munir, O. Rana, Cloud Market Maker: An automated dynamic pricing marketplace for cloud users, *Future Generation Computer Systems* (2015), http://dx.doi.org/10.1016/j.future.2015.06.004

Highlights (for review)

1. This paper presents Cloud Market Maker (CMM); a marketplace for cloud users.
2. The system provides a dynamic pricing marketplace for providers to maximize their revenue.
3. It provides users with decision support when choosing a cloud resource.
4. It employs a multi-agent multi-auction approach for creating an automated marketplace for cloud users which were not possible with other existing systems.

Cloud Market Maker: An automated dynamic pricing marketplace for cloud users

Barkha Javed [a] *, Peter Bloodsworth [a], Raihan Ur Rasool[b], Kamran Munir [c], Omer Rana [d]

[a] *NUST School of Electrical Engineering and Computer Science,, Islamabad, Pakistan*
[b] *King Faisal University, Saudi Arabia*
[c] *University of the West of England, Bristol, UK*
[d] *Cardiff University, Cardiff, Wales, UK*

Abstract

Cloud providers commonly incur heavy upfront set up costs which remain almost constant whether they serve a single or many customers. In order to generate a return on this investment, a suitable pricing strategy is required by providers. Established industries such as the airlines employ dynamic pricing to maximize their revenues. In order to increase their resource utilization rates, cloud providers could also use dynamic pricing for their services. At present however most providers use static schemes for pricing their resources. This work presents a new dynamic pricing mechanism for cloud providers. Furthermore, at present no platform exists that provides a dynamic unified view of the different cloud offerings in real-time. Due to a rapidly changing landscape and a limited knowledge of the cloud marketplace, consumers can often end up choosing a cloud provider that is more expensive or does not give them what they really need. This is because some providers spend significantly on advertising their services online. In order to assist cloud customers in the selection of a suitable resource and cloud providers in implementing dynamic pricing, this work describes an automated dynamic pricing marketplace and a decision support system for cloud users. We present a multi-agent multi-auction based system through which such services are delivered. An evaluation has been carried out to determine how effectively the cloud market maker selects the resource, dynamically adjusts the price for the cloud users and the suitability of dynamic pricing for the cloud environment.

* Corresponding author E-mail Addresses: barkha.javed@seecs.edu.pk (B.Javed), peter.bloodsworth@seecs.edu.pk (P.Bloodsworth), Rrasool@kfu.edu.sa (R.Rasool), Kamran2.Munir@uwe.ac.uk (K.Munir), o.f.rana@cs.cardiff.ac.uk (O.F.Rana)

Keywords:

Cloud Computing; Cloud Market Maker; Dynamic Pricing; Multi-agent; Multi-auction; IaaS Cloud Providers

1. Introduction

In recent years growing companies have needed to invest heavily in their IT infrastructures. This makes it difficult for new startups to become established and for small companies to develop and achieve their business goals. Today a growing number of cloud infrastructures are becoming increasingly important in addressing the needs of business. In many cases using the cloud reduces the financial barriers to enter a particular marketplace. Public cloud commonly follows a pay-as-you-go model. In this model, computing can be used as a utility which saves the user from investing significantly on IT infrastructure. Examples of public Clouds include Amazon Web Services, Google AppEngine, and Microsoft Azure [1]. These have attracted attention from both industry and academia due to their ability to provide cost effective services. They save growing companies from investing in complex IT infrastructure and the associated human resources that are required for managing them.

The adoption of cloud computing has increased to a remarkable level in the last few years. The survey conducted by Cloud Industry Forum (CIF) showed that by the end of 2013 more than 75% of UK businesses would be using at least one type of cloud services [2]. Furthermore, AMI partners predict that by 2014, small and medium enterprises are expected to spend around $100bn on cloud computing [3]. The rapid increase in its usage is generating economic growth in many sectors. According to a report by Gartner, the cloud computing market revenue has increased from $91.4bn in 2011 to $111bn in 2012 and this is expected to reach $206.6bn by 2016 [4].

The general excitement surrounding cloud computing has attracted and increased the number of cloud providers, as a result of which competition among them is increasing. In order to survive and to prove successful, a pricing approach is required that can maximize the revenue of providers. In industries such as the Airlines a dynamic pricing approach is used to increase their profits. Airlines have high levels of operational costs that include establishing and maintaining airline services, labor, airport equipment, airplanes, and other related services. This upfront cost remains reasonably static whether an airplane carries one passenger or if it is full. Dynamic pricing plays a pivotal role in increasing revenues by filling each flight up to the greatest extent possible. Prices begin low but rise dynamically as more bookings are made and spare capacity reduces. American Airlines are estimated to generate an additional $500 million in revenue every year based on its dynamic pricing strategies [5].

In a similar way to the airline industry, cloud providers also need to invest considerably in their infrastructure before they can begin trading. A heavy investment is required in procuring data center space, setting and management of the data center, software licensing, firewalls, physical security, and on human resources. This cost remains almost constant whether a provider serves one customer or a large number of customers. However, most Infrastructure-as-a-Service (IaaS) cloud providers charge a fixed price for their services [6] with the exception of Amazon spot instances that use a dynamic pricing approach [7]. Dynamic pricing has shown to be beneficial in increasing the revenue of airlines. Cloud providers could therefore also potentially benefit by adopting similar strategies to increase their revenue. Thus, the first motivation of this work is to propose a dynamic pricing approach for cloud providers.

The increasing number of cloud providers whilst giving consumers a greater choice also makes it difficult for them to select the most appropriate provider for their needs. Every IaaS cloud provider prices their resources differently and customers therefore often need to get quotes from several potential suppliers. Searching for and comparing each and every cloud provider is a time consuming task. In order to do this, a customer could rely on prior knowledge about cloud providers or may opt for a well-known provider without really considering all the options. This may prove more expensive for the consumer and limits the entry of emerging cloud providers into the marketplace. The many different cloud pricing models increases the need of a system that can support customers in deciding which cloud provider to select. A platform is therefore required which provides a unified view of cloud providers to customers.

Therefore, the second motivation of this work is to assist customers in the selection of most suitable cloud provider for their requirements.

This paper presents Cloud Market Maker (CMM) a system which has been developed to create a dynamic pricing marketplace for providers and to provide users with decision support when choosing a cloud resource. The CMM creates a market wherein cloud providers can list their resources and associated prices; it dynamically adjusts the cloud resource price and assists customers in the selection of the most suitable provider from the market. The contribution of the paper is twofold. Firstly, the CMM dynamically adjusts the prices of cloud resources in real-time using a supply demand auction based model. It assists providers in determining the market price for their resources. This also helps emerging providers to be discovered by potential customers. Secondly, the CMM also works on the behalf of the cloud customer. It supports customers in making the decision regarding the most effective provider for their requirements. Based on the customers' requirements, it selects a suitable cloud provider for them by searching a unified view of suppliers. The CMM not only selects a suitable provider but also helps the user to contact them in order to buy and access the cloud resource. Both of these main features have novelty in themselves but together they produce an approach which has not been tried before according to the understanding of the authors.

The paper is structured as follows: Section 2 covers the work related to dynamic pricing strategies for cloud providers and existing comparison portals for IaaS cloud customers. It also considers the literature on Multi-agent Systems (MAS) and auction mechanisms. This review establishes the suitability of MAS and auction mechanism for constructing a cloud market maker. Based on the literature review, the cloud market maker system is presented in Section 3. An evaluation of the approach is covered in Section 4. Finally, the conclusion and future work are presented in Section 5.

2. Related Work

2.1 Cloud Pricing Models

In a rapidly changing environment such as the cloud, dynamic pricing is required in order to adapt to the constantly changing market conditions. It is an effective strategy to cope with unpredictable demand, unutilized resources, and to generate more revenue [8]. At present however most cloud providers offer resources at a fixed price (with the exception of Amazon's Spot Instances). The resources offered by providers are mostly categorized as On-Demand and Reserved Instances. Amazon is the only major provider at present who is providing spot instances [7], on-demand and reserved instances. In order to acquire a spot instance the customer is required to place a bid. The user secures the instance when their bid exceeds the current spot price. The spot instance can be used for as long as the current spot price remains below the customer's bid. It is not known in advance how long the instance will be available for. The spot price of an instance changes in relation to the level of demand for it. Using this type of resource has its risks as it can be lost at any point in time when the spot price increases. This makes it of limited value in the majority of usage scenarios.

The usage of a Dynamic pricing strategy to maximize revenue for an individual provider was proposed by Xu and Li[8]. In this work a revenue management framework was proposed to tackle the infinite horizon dynamic pricing problem faced by Amazon. In the cloud environment, the demand for resources is non-deterministic therefore a near optimal pricing policy is required to deliver a better return on investment. Further, an auction mechanism has been employed by Wei et al. for dynamic price adjustment in a cloud environment [9]. The proposed approach addresses the shortcoming of Amazon's spot instances. Unlike them, users are not interrupted in the middle of their task i.e. they can use the instance as long as they require. A mechanism has been proposed in the paper that guarantees the services; the instances once auctioned off are not made available to other users until the user terminates it.

An interesting study has been carried out by Arun and Marc [10] which considered the enhancement of the revenue that cloud providers generate. In a similar way to that of the airline industry, they have introduced the concept of different classes of buyer within a cloud environment. In an airline the classes of travel include economy, business and first class. The authors have employed this idea and based on the buyers' classes, current demand, resource reservation, and future prediction, the cloud resources are allocated to users. It was concluded from their study that in comparison with static methods, dynamic pricing generates more revenue and increases the utilization of a provider's resources. The advantages of dynamic pricing over static pricing in cloud environment have also been demonstrated in [11].

Existing work demonstrates the suitability of dynamic pricing for use in the cloud. It has been shown to increase the profit and utilization of resources. Dynamic pricing is presently employed by only one IaaS cloud provider (Amazon) whereas most of the other IaaS cloud providers employ static pricing for their resources. Furthermore, it has been observed from the literature that much of the work has been done to improve the existing dynamic pricing schemes work with an individual provider such as Amazon. At present, no system is available that works for a number of cloud providers. Our system however has the ability to create a market in which several IaaS cloud providers can list their resources and the market maker adjusts their resource price based on the current market situation.

2.2 Decision Portals for Cloud Customers

Our work not only acts on the behalf of IaaS cloud providers but it also assists cloud customers. Each supplier tends to have their own pricing models for their resources. A provider commonly packages their network, storage, and compute services differently which makes it difficult for customers to understand which factors make up their charges. A platform is therefore required that provides decision support and a unified view of IaaS cloud providers for cloud customers.

In order to assist customers in decision-making, the CloudHarmony portal [12] provides benchmarks for public cloud providers. Benchmarking parameters include *network*, *performance*, and *uptime monitoring*. Based on the benchmark results, the selection of cloud provider is often left to the customer. This is a time consuming task because customers need to compare each and every provider's benchmarks. Moreover, the portal does not assist the customers in making comparisons between the different pricing models of cloud providers. The Cloudorado portal [6] also provides the pricing details of IaaS cloud providers to cloud customers. In order to make a comparison the customer is required to visit each and every cloud provider which is a time consuming task. This makes it difficult for the consumer to decide which provider is more appropriate for their requirements.

Assisting customers in deciding which cloud provider is the most suitable one for their needs is the focus of work by Ang Li et.al [13] which proposed a cloud comparator (Cloudcmp). Based on customer's perceived preferences, a comparison of four top cloud providers including Amazon Web Services, Rackspace CloudServers, Microsoft Azure, and Google AppEngine was carried out in this paper. The results given were only relevant in the time period in which they were generated. They cannot be generalized due to the varying resource levels and heterogeneous pricing models of cloud providers. The system does not assist customers in resource selection. Moreover, comparison of only the top four cloud providers has been carried out in this work and many emerging providers have therefore been missed out. Such issues could be resolved with the provision of an automated system that take into account the different pricing models of cloud providers and could assist potential customers in the selection of a suitable provider.

To enable cloud customers to locate a cost effective cloud provider for their requirements, PlanForCloud[14] creates a detailed cloud cost forecast for cloud customers. It produces a three year forecast based on customers' requirements and the list prices from providers for their respective services. This helps cloud customers to select the most cost effective cloud provider for their requirements. Our system is different from PlanForCloud as it dynamically adjusts the price for cloud providers. It also works for on-demand resources i.e. one hour or for a month. The user-based decision regarding the selection of a cloud provider is supported by our system. Customers and providers only need to provide their requirements and resource details.

During our literature review we could not find an automated system that can select a suitable cloud provider for cloud customers. Previous work [6, 12, 13, 14] assist in the comparison of cloud providers but the decision of selection of a cloud provider is left to the customer which may be frustrating for users.

Furthermore, the automated comparison of pricing models is not supported by existing work. A system is therefore required that can take into account the different pricing models and selects suitable provider for the cloud customers. In order to make a selection, an effective way of working out the best resource for user is required. An intelligent entity is therefore necessary that can work autonomously, on behalf of the users, and to assist them in comparison and deciding of a suitable cloud resource. Cloud computing has a dynamic nature i.e. resource that is available at one moment may disappear at the next. This means that an intelligent entity needs to be able to work 24/7 without human intervention. A Multi-agent System (MAS) seems suitable for this task because agents have potential to intelligently take decisions and can manage rapidly changing conditions.

2.3 Multi-Agent Systems for Managing Complex Environments

Over the past few years, MAS have found their application in information management. The use of MAS reduces the human effort and has proved to be useful in the constantly changing environment [15]. The flexible nature of agent based systems being employed in open and dynamic environments such as internet for information gathering, retrieval, and provision [16]. It is clear from the literature that agents can successfully mediate e-commerce transactions. Agents are employed in Business-to-Business (B2B), Business-to-Consumer (B2C), and Consumer-to-Consumer (C2C) environments. Agents have potential to perform tasks of e-commerce such analyzing and information gathering, resource brokering, and carry out the transactions on behalf of users [17,18].

Multi-agent Systems have also demonstrated their ability to solve distributed computing problems and perform distributed information fusion [19]. MAS have been applied in the grid for resource management and scheduling. Chia-Hung Chien et. al. [20] proposed a multi-agent approach for allocating resources to tasks of a workflow in the grid environment. Agent-based technologies have not only demonstrated its advantage for grid resource allocation but also for grid resource management. Economic agents are employed by Li Chunlin and Li Layuan [21] for grid resource management. The work in [22] uses agent teams for grid resource discovery, allocation, and management. The benefits and success of agents in the grid has also attracted their use in the cloud environment because problems in grid and cloud are quite similar. A multi-agent approach can be utilized for resource discovery, brokering, trading, and managing of cloud resources [23]. A MAS is used by Sim [24] for cloud resource discovery, negotiation, and service composition. An agent-based test-bed for cloud resource discovery, management, and Service Level Agreement (SLA) negotiation has been proposed in [25].

The literature shows the potential of MAS to carry out complex tasks and demonstrates their suitability for a distributed environment such as the cloud. In this work we employed a MAS to construct the cloud market maker. Agents work on behalf of customers and providers. It selects a suitable provider for customer and dynamically adjusts the price for providers. Further, dynamic approach has been adopted for pricing of the cloud resources. Besides the many advantages of a dynamic pricing approach, it does however have some drawbacks which need to be addressed.

2.4 Auctions

A dynamic pricing scheme introduces additional complexities that are associated with market pricing and budget planning [26]. Buyers and sellers are required to make up front choices in relation to their budgets and minimum pricing requirements. Constantly fluctuating levels of supply and demand necessitate that a mechanism is put in place which can adjust prices in real-time without user involvement. An auction mechanism appears to be well suited to these challenges because of its abilities in terms of price determination and adjustment. Given the supply and demand at any given point of time the price of the resource is automatically set. Furthermore, it also provides the room for price negotiation between parties [26, 27].

Auction mechanisms are not without their potential drawbacks however. For example, if an agent participates in just a single auction then there is always a possibility that they will lose it or perhaps pay above the market value for the item. Participation in multiple auctions helps agents in buying the good at a suitable price, not exceeding the buyer's valuation; it has attracted attention of many researchers. A multi-agent system called a BiddingBot is proposed in [28]. A heuristic bidding strategy for buying multiple goods in multiple English auctions has been proposed in [29]. The heuristic bidding algorithm that is employed in the paper is the Earliest Closest First (ECF) heuristic algorithm. A coordination algorithm for participation in multiple simultaneous English auctions (for multiple goods) is proposed in [30]. The coordination algorithm ensures that agent has the lowest leading bids possible to purchase the required number of goods. A Neuro-Fuzzy Approach is used in [31] for bidding in the multiple English auctions.

A multi-auction approach has a great deal of promise, especially as it closely follows the ways in which humans create supply demand based marketplaces. This strongly suggests that it is also suitable for use in our Cloud Market Maker. Unlike Amazon spot instances, CMM employs an auction mechanism to determine the dynamic price of cloud resources by creating a market within which a number of different cloud providers participate. The price is adjusted up or down based on the current market conditions and it therefore provides balanced benefits to both consumers and providers. Furthermore, our system employs a multi-auction approach to increase the probability that customers will be able to acquire resources within their time constraints.

2.5 Cloud Brokers

A financial brokerage model for cloud computing has been proposed by Rogers and Cliff (R&C) [32]. The R&C brokerage model extends the WZH reservation model [33]. The broker proposed in [32] acts as an intermediary for cloud providers and customers. The broker translates on-demand behavior to reserve instance requirements which helps providers to forecast market demand for the following 12 to 36 months. This assists them to manage their resources in terms of capacity and to plan for future periods of demand. It also works for cloud customers and assists them to acquire resources at lower prices.

The system proposed in [32] seems quite similar to our work. However, there are several significant differences. Our cloud market maker adjusts resource prices for cloud providers based on the current market conditions. It also provides them a platform through which they can advertise to attract customers. The R&C model does not dynamically adjust the price for cloud providers its aim instead is to forecast future demand. Unlike the R&C model, the cloud market maker doesn't need to buy resources from cloud providers and therefore does not directly provide the resources instead of the provider. It only takes the resource details from providers and makes them available to cloud customers by providing them a unified view.

Both the R&C system and the cloud market maker work for cloud customers. The goal of the R&C model is to provide economical resources to cloud customers, whereas the aim of the cloud market maker is not only to select most economical resource but also to select the most suitable one. The R&C model also works only for reserved instances whereas the cloud market maker works for both on-demand and reserved instances. Furthermore, it is not clear in [32] which types of cloud providers are being considered whereas in our work only IaaS cloud providers are targeted.

2.6 A Marketplace for Cloud Users

In order to allow Amazon cloud customers to sell their unused EC2 reserve instances to other Amazon customers, Amazon recently opened the *Reserved Instance Marketplace* [34]. Their goal is to provide a new facility to existing Amazon customers through which they can sell their unused resources. A paper published by John and Philip [35] argued that the opportunities which exist for cloud brokerage are reducing and that new secondary markets, such as Amazon's Reserved Instance Marketplace are replacing them.

Recently, the IaaS cloud provider Enomaly opened a clearing house and marketplace called *SpotCloud* [3, 36]. Using SpotCloud, cloud service providers can sell their unused resources to buyers and resellers who require resources at the best possible price. SpotCloud provides the opportunity to providers to sell their unused resources which may otherwise go unsold. Unlike SpotCloud, our system helps providers in dynamic price adjustment (both up and down) and assists customers in finding a suitable provider. In our approach, the providers' resources are not sold to the marketplace. Our work presents a new type of market-based decision support system for cloud users. SpotCloud and the Amazon Reserved Instance Marketplace show that such a concept is practical. It also appears that cloud providers are looking to embrace new techniques which allow them to generate more revenue from their resources.

3. The Cloud Market Maker System

3.1 Motivating Example

Before going into greater detail regarding the Cloud Market Maker we will consider a practical example which will clarify the anticipated application of system. Assume a company 'Cloudy Inc.' starts a new project (say Project 1) that requires heavy computing resources. The company buys some new hardware for the completion of the project. While project '1' is in progress, the company receives a new project '2' which also requires a heavy investment. This not only limits the company in terms of starting additional projects but also makes it difficult for it to grow. To meet this need, the company decides to use the capabilities of cloud computing. It searches for the suitable provider for its needs. It must look at each available provider before making a decision and there are many of them (it takes more than a week.) Cloudy Inc. eventually decides to use provider 'C' because it is well-known, it advertises its services well, and its pricing model is easy to understand. Provider 'A' is a new provider and is therefore not selected by the cloud customer because it does not know what reputation it has. Furthermore, it does not select provider 'B' because its pricing model is complicated and is not understood well by the customer. In this case Cloudy Inc. loses out because provider C is actually much more expensive than providers A and B. This reduces its profitability and thus its long-term stability.

Let us now look at the same example from the cloud provider's perspective. An emerging provider such as provider 'A' may likely face problems making customers because it has limited marketing resources and needs to establish a good reputation before the majority of customers will trust it. Due to this, most of its resources are underutilized which makes it difficult for the provider to break even, find its place in the market, and compete with other cloud providers. Provider ‘C’ is a well-known provider and might face the opposite problem by having too many customers and not enough computing resources to serve all of them. This may result in damage to its reputation among its customers and instability in its systems.

The case study discussed above illustrates that cloud customers face problems when selecting the most suitable cloud provider. High prices and complex cost models limit their choices. Cloud customers thus rely on their prior knowledge regarding cloud provider which further restrict their decisions. On the other hand, cloud providers can experience dramatic swings in resource usage. At some less busy times hardly any resources are rented out and other times demand exceeds capacity. This makes it difficult for them to achieve their business goals. Furthermore, new and emerging providers find it difficult to attract customers because consumers trust on their knowledge or experience in deciding the provider for their requirements.

A unified platform is therefore required by both cloud customers and providers. Customers need it for decision making and providers require it to efficiently utilize their resources and to attract customers. However, the literature shows that at present no such system is available. Therefore, we have developed a marketplace called Cloud Market Maker for cloud users. It is a unified platform for cloud customers and providers. It will attract customers because of its ability to work on their behalf by helping them to decide on a suitable provider for their requirements.

Providers also need to be part of the marketplace in order to make their resources available to potential consumers. They can register each of their virtual machines in the marketplace which are given a unique identifier. They are then sold automatically using an agent-based auction mechanism. Cloud market maker dynamically adjusts the price for cloud providers which will help them to efficiently utilize their resources and to manage demand thereby improving their service provision. At the end of a successful auction a customer is paired with a provider at an agreed price. The unique identifier is then used to direct the user and transaction to the pre-existing payment and virtual machine provisioning systems of the selected provider. Our system is designed to co-exist with the existing policies and mechanisms of cloud providers such as Amazon and Rackspace rather than to rely on future developments in InterCloud [37] operability. The only extra requirement on providers is that they will need to carefully manage a registry of which virtual machines are part of the marketplace and which are available to their internal sales platforms. We believe that this makes our system practically applicable for real-world cloud providers.

Cloud Market Maker is intended to be deployed by a third party service provider who supplies interfaces for both cloud providers and customers. Through such web-based interfaces both groups enter their requirements. This simplifies the application set up as it means that the platform (discussed in next section) does not need to be deployed locally on the systems of cloud provider or customers. Cloud providers are given login IDs and submit information regarding their available instances. They can view the details of their previous auctioned instances. The system also shows the status of their current instances. Through the cloud customers' interface, users can provide their requirements, urgency, and budget details. The customers can also view status of their requests. Using our system, the cloud providers will provide their resource and price details only. The aim of our system is not to take over the hosting from cloud providers but to apply dynamic pricing to their existing offerings. The system matches customers with providers by considering the requirements of both parties within the supply and demand context that exists at a given time.

3.2 System Architecture

The cloud market maker is required to perform complex tasks (price regulation and decision regarding best provider), work on behalf of users, work autonomously, and dynamic nature of the cloud requires it to work 24/7. Therefore an intelligent entity has been identified for this purpose. The literature review, which was briefly summarized in section 2, demonstrated the suitability of a multi-agent system for our purpose and it has therefore been employed in CMM. Furthermore, the literature showed the benefits of a multi-auction approach for dynamic price adjustment and it has therefore been applied to create a marketplace. The use of agents in auctions has been shown to outperform humans in several studies [38, 39]. The auction process involves bidding which is time consuming; the use of agents in the bidding process saves time and also provides the flexibility of participating in multiple simultaneous auctions [39]. Based on the analysis made in the previous section, the cloud market maker has been designed and developed accordingly.

The architecture of the cloud market maker has been designed by taking into consideration the multi-auction approach and agents that are required for complex decision making. Therefore, the system follows a multi-agent architecture. The agents work autonomously and create automated marketplace for cloud users. The system is required to incorporate agents that can work for providers and customers, manage and run auctions (for dynamic price adjustment), handle the information regarding the auctions running in the system, and control the overall platform. The architecture of the cloud market maker system has been designed by taking this in consideration and is shown in Figure 1. It shows how the six classes of agents in the system work together.

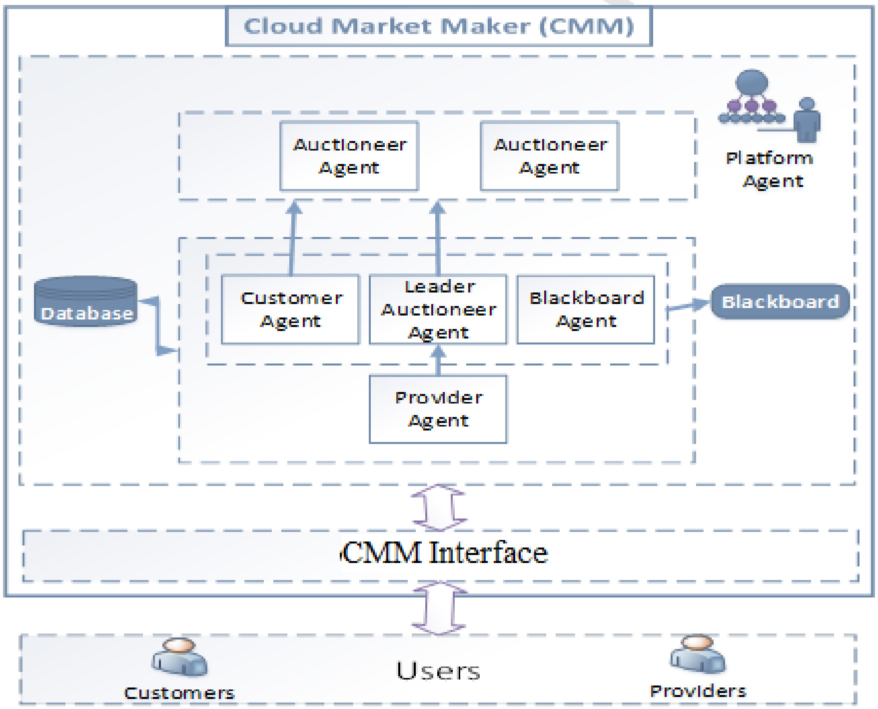

Figure 1: Architecture of the Cloud Market Maker

The Agents in the Cloud Market Maker

Agents: P**latform** A**gent (**PA**),** C**ustomer** A**gent (**CA**),** PR**ovider** A**gent (**PrA**),** B**lackboard** A**gent (**BA**),** L**eader** A**uctioneer** A**gent (**LAA**), and** A**uctioneer** A**gent (**AA**)**

1. The PA is responsible for launching and managing the agents and scaling the system up and down in relation to system load.

2. The customer makes requests and the provider issues offers via the respective CA and PrA.

3. The BA displays the auctions running in the system.

4. The AA are managed, launched, and deleted from the system by LAA.

3.3 The Working of the Cloud Market Maker

The architecture, as shown Figure 1, shows that customer and provider agents work on behalf of the cloud customers and providers. The customer agent is responsible for placing bids in an auction and selection of most suitable resource for the customer. The cloud users (providers and customers) provide the requirements details to the system through the interface of the cloud market maker (as shown in Figure 1). The requirements details include system specification and user details. In addition to this the cloud customers also provide their urgency values to the system; this will be discussed in detail in section 3.4. In IAAS cloud services, resources are commonly delivered in the form of virtual machines (VMs). The Operating System (OS), network bandwidth, and location are not currently taken into account by the system and are therefore not included in requirements by users. These may be considered by some to be important parameters, but they are also prone to dynamism and make accurate comparisons difficult. Bandwidth for example, can be dependent on networking conditions at a given time and may therefore change frequently. Moreover, the instances provided reflects the on-demand or reserved instances. The notion of Amazon's spot instance is not considered because currently only Amazon is providing spot instances. Moreover, users may not often choose spot instance for their requirements because there is always a risk that there tasks will be cancelled when the market price of that spot instance becomes greater than the user offered price.

The platform agent controls the overall system. The leader auctioneer agent handles the auctioneer agents. The auctioneer agent in the system is responsible for managing the auctions that include starting, restarting, and closing of auctions. The auctioneer agent is also responsible for dynamic price adjustment. When cloud provider provides the details of unused/available resources to the system, the system will make the resource available to customers by auctioning it. In a similar way to that of Ebay [40], the cloud market maker also makes resources available whenever a provider needs to sell / rent out a resource i.e. the resources are made available for auction by providers. If an auction is unable to sell the resource then auctions are rescheduled every 30 minutes (in our system each auction lasts 30 minutes). However, auction duration can be configured after further experimentation to match the needs of the system or the load upon it.

The auction selection process is just as important as bidding in an auction. If the auction selection is poor then chance of acquiring the resource is also likely to be very low. The selection algorithm has therefore been carefully designed and is covered later in this section. The selection source is equally important i.e. the mechanism by which the customer agent gets to know about the auctions that are currently running in the system. The architecture of the system shows that a blackboard has been employed to display the auctions. In a blackboard architecture [41] agents do not directly communicate with each other instead the information is made available to all agents in a system through a common information space. This reduces the complexity and overhead of message exchange between the agents. Furthermore, this is well suited for dynamic environments. The customer agent interacts with the respective blackboard to get the required information. It saves the time of customer agent and reduces the message overheads. The working of the system is explained below.

Providers send details regarding their available resources to the PrA. The PrA sends information to the LAA regarding the new entry. On receiving the resource details from the provider, the LAA starts an auction. The customers enter the system via the PA. The PA registers the customers and the customer request is then given to CA. Based on the customer's request; the CA is redirected to the blackboard agent. The BA is responsible for managing the blackboard. The blackboard displays the available resources of providers for auction. The offers are given to the BA by the provider agent who takes the requirements details from providers. The blackboard has seven entries i) Auction ii) Provider ID iii) Item ID iv) Current highest bid in an auction v) Number of bidders in an auction vi) End-Time (each auction will run for 30 minutes) and vii) requirements details. The reason for including the ItemID is that it might be possible that one provider lists multiple resources; the ItemID helps in distinction. The CA reads the blackboard and based on the selection function, given in the subsequent section, the CA selects the auction to participate in. In case if agent finds that winning is not possible in the auction then the CA can switch to some other auction. This increases the probability of getting the resource. Finally, the result of the auction is given to the customer and the provider agents. The customer and provider agents return the result to the customer and the provider. Furthermore, the database as shown in Figure 1 holds the information of customers, providers, feedback, and rating of providers.

3.4 Selection of Resource and Dynamic Price Adjustment

This section covers the resource selection and dynamic price adjustment techniques in detail. In the first step the blackboard is read by the CA, which selects an auction using a Selection Function which is shown in Figure 2. The CA considers seven parameters for selection: current highest bid of an auction (a1), remaining time of auction (a2), current number of bidders in an auction (a3), budget of the customer (a4), urgency of the customer (a5), remaining time to get the resource (a6), and Quality of Service (QoS) (a7). Urgency is an option which gives the customer the ability to indicate how urgently the resource is required. The urgency is categorized into six groups i) Immediate (i.e. within 1 hour) ii) within 2 hours iii) within 3 hours iv) within 6 hours v) within 12 hours vi) and within 16 hours. Urgency plays an important role in price acclimation. The immediate urgency level reflects the customer's willingness to pay the maximum of their budget. The lower urgency levels (such as 12 hours, 16 hours) indicate that a customer prefers to get the most economical resource and it gives agents the chance to search for the most affordable ones. The urgency values can be fined tuned during the experimentation phase. The selected values reflect the capability of system to adjust the price in relation to different urgency levels; this is further explained in bidding algorithm which is given in Section 3.4.3.

3.4.1 Quality Of Service (QoS)

Before going into the detail of how the selection function and bidding algorithm are used in practice, it is important to consider the Quality of Service (QoS) that is provided. In a dynamic environment such as the cloud, it is quite difficult and challenging to analyze QoS levels because users' preferences regarding it may vary. Therefore in order to capture this aspect, our system considers only the rating of cloud providers. Customers' feedback includes availability of instances, performance, response time, fulfillment of Service Level Agreement (SLA) of a provider, and elasticity. The metrics through which customers rate a service is shown in Table 1.

Table 1: Feedback by Cloud Customers

<table>
<tr><th>No.</th><th>Parameters</th><th>Rating</th></tr>
<tr><td>1.</td><td>Availability</td><td rowspan="5">Excellent, Very Good, Good, Average, and Poor</td></tr>
<tr><td>2.</td><td>Performance</td></tr>
<tr><td>3.</td><td>Response Time</td></tr>
<tr><td>4.</td><td>Fulfillment of SLA</td></tr>
<tr><td>5.</td><td>Elasticity</td></tr>
</table>

Customers provide feedback regarding the cloud services they access via the CMM system. They rate the services using a range from Excellent to Poor. Based on the customers' feedback, the system implicitly assigns weights (points) to the providers' services; this helps in quantifying the QoS. For Excellent, Very Good, Good, Average, and Poor rating 10, 9, 8, 5 and 0 points have been set. Using the customers' feedback the system calculates the individual provider ratings; it implicitly assigns an overall score of between 0-10 to each provider. The system calculates a provider's QoS by taking the average of its feedback.

3.4.2 Selection Function

A selection function has been created in order for agents to be able to choose between resources. A multi-attribute utility function (U) is used for this purpose because in order to select an auction, a number of attributes need to be considered. The urgency value guides the selection of auction in the first place and is also useful in determining how bids should be placed during it. The budget is used during selection to check if the highest bid is less than customer budget or not. Moreover, the comparison of budget and current highest bid in an auction also helps in selecting those auctions that CA can bid in. After pruning the number of auctions (L), the utility function is applied on the selected auctions to see which auction is the most suitable for the agent to participate in.

In order to understand how the selection function is used in practice let us consider the following scenario. Suppose a new customer enters the system and provides the requirements to the CA. Based on the given requirements, the CA reads the relevant blackboard. In order to increase the probability of an agent successfully getting a resource, the CA will tend to select an auction that ends within a short period of time. This means that an agent will likely have the chance to participate in multiple auctions (if for example it fails to get the resource from current auction.) The CA needs to select the most appropriate auction to participate in. It firstly considers the customer's urgency and budget. The comparison of highest bid in an auction with the customer's budget is based on the urgency.

The urgency of the customer guides in auction selection and bidding. As the urgency for a resource increases (time reduces) agents are allowed to select and bid up to an increasing percentage of their overall budgets. For example, if urgency is: 16 hours, 12 hours or 6 hours then the customer agent will check if the highest bid in an auction is less than or equal to 20%, 40%, and 70% of the customer's budget before going ahead with a selection and a bid. Within the last three hours of the agent's deadline the CA will increase the threshold to 80%. If the urgency is one hour then the customer agent will check if highest bid in an auction is less than 95% of customer's budget. These thresholds were calculated in relation to the urgency levels. Different thresholds can be chosen to provide greater flexibility to customers. The urgency levels and thresholds will not negatively impact the provider because the auction mechanism ensures that the resource is provided above or at the provider's minimum required price. Moreover, the customer agent will participate in the earliest closing auction. In the case of the immediate urgency level; the number of active bidders in an auction is also considered. Based on these attributes, the points are calculated through the utility function and each auction is assigned the points accordingly. The selection function is given in Figure 2. The variables a1, a2, a3, a4, a5, a6, and a7 in Algorithm 1 were previously described (at start of Section 3.4).

Algorithm 1: Selection Function

Input: a1, a2, a3, a4, a5, a6, a7

Output: Selected Auction

IF a5 NOT EQUAL 0

  Then

    IF a4 Greater Than a1

      Then

        x1 = (1-(a1/a4))

        x2 = 10-a3

        x3= (1- (a2) /(a6))

        x4= a7

      IF u=u1then

      U=∑(x1, x2, x3, x4)

      Else

U=∑(x1, x3, x4)

END IF-ELSE

END IF

END IF

Figure 2. Selection Function

The utility for each auction is calculated. In the case where two or more auctions return the same utility score the CA will choose the first auction with that score. The CA will participate in the auction with the greatest utility score. If urgency is immediate, then only those auctions will be selected that have a1<95% of customer's budget. The auctions in which a1=>95% of customer's budget will not be considered because there is lower chance of getting an item from such auctions. It might also be possible that the highest bid is so close to the customer's budget that the agent cannot make a bid. The 95% threshold is chosen so that CA has some chance of placing bids in an auction and acquiring the resource without exceeding the customer's budget. If a1<95% of customer's budget then the CA will calculate x1, x2, x3, and x4 and based on the value of U, the auction is selected. The CA will participate in the auction which has the greatest value of U. The reason for not considering x2 for low urgency is that there is enough time for CA to buy an item. In this case it is less important to limit the number of bidders. Considering the number of bidders for immediate urgency increases the chance of resource acquisition. This is because generally speaking the fewer bidders there are in an auction the more likely that an agent will win the resource. The limitation of ten bidders is therefore only used for the immediate urgency level which can be further adjusted by experimentation.

3.4.3 Bidding Algorithm

After the selection of the relevant auction, the customer agent bids in the auction using the bidding algorithm given in Figure 3(only first and last cases are shown here). The auction determines the market price of a given resource. In some cases a large number of bidders may act to increase a resource price but the system always ensures that selected resources remain within the customer's budget. In a similar way to that of the selection function, urgency also plays an important role in bid placement during an auction. The reason for using different urgency levels during bidding is to give the customer agent a chance to acquire most economical resource for the customer. In cases with a low urgency level (16 hrs, 12 hrs, 6 hrs) the customer agent attempts to get a resource at the lowest possible cost to the user even if this means participating in several auctions.

The customer agent starts with a low percentage of its overall budget. If it fails to acquire the resource at such a percentage then it gradually increases it. When the urgency level reaches immediate the customer agent bids up to 100% of its budget in order to increase the chance of resource acquisition. If urgency levels are not considered during bidding then the system would always bid up to maximum of the customer's budget. This may not necessarily result in the acquisition of the most economical resource for customers. Therefore, the urgency of the customer is considered during selection and bidding. Furthermore, it is worthwhile to mention that at all levels of urgency the system ensures that resources are only sold at or above the providers' reserve price. This process follows strategies which humans commonly use in auctions to buy items in a cost effective manner. For example, consider how many people use Ebay in a similar way when they are shopping for something.

Algorithm 2: Bidding Algorithm

Input: Bids in Auction

Output: Result (Win or Lose)

Auction A selected

After selection CA switches to the following cases;

Case1: a6 LESS than Equal 1hour

IF a1 LESS than 95% of customer's Budget then

Continue bidding until 100% of

customer's budget is reached.

IF outbid then

Use selection function to select another auction

Case6: a6 EQUALS 16 hours

IF a1 LESS than 20% of customer's budget then

Continue bidding until 20% of customer's budget is reached.

IF outbid **OR** a1 GREATER than 20% of customer's budget then

Use selection function to select another auction

Figure 3. Bidding Algorithm

The selection of suitable resource is considered during auction selection whereas the bidding algorithm dynamically adjust the price. In order to find the system's capability, an evaluation has been carried out and is covered in detail in the next section.

4. Evaluation

In order to evaluate the Cloud Market Maker system, it is important to examine the system's ability to satisfy the cloud customers and providers in terms of resource selection and dynamic price adjustment. This can be carried out by establishing the system's capabilities when it is presented with a range of user preferences and a rapidly changing market environment. A prototype system was therefore fully implemented using JADE [42] which was then used during the evaluation process. The system works for users and supports them in decision making. To evaluate how well it performs these roles we need to examine the system's ability to reach an agreement between customers and providers for a given resource. Therefore, the first criterion employed for evaluation is to calculate the number of successful transactions i.e. the number of winners. It demonstrates the number of customers/ providers satisfied by the system. It also shows how effectively customers have been mapped to providers using the multi-agent approach.

In addition to the selection of the best provider for cloud customers, the cloud market maker also dynamically adjusts the price for cloud providers. To determine the suitability of dynamic pricing for cloud providers, the second criterion for evaluation is the transacting price. It is the price at which the resource is provided to the customer. It indicates the customers' total budget usage and providers' total revenue generated when static and dynamic pricing approach is employed. Furthermore, to discover the aptness of dynamic pricing for a changing environment like the cloud, a comparison of static and dynamic pricing has been carried out.

The evaluation of the system was quite challenging because at present no comparable system could be found. Therefore, the only system that CMM system can be compared to is that of a manual approach. Such an approach involves humans in the selection process. The manual selection of resource illustrates the humans' capability to make the decision regarding the cloud resource selection. Therefore for this evaluation, the mapping of customers to providers was firstly carried out manually to estimate the possible number of transactions and the transacting price then it was done using our prototype to determine the system's performance. A comparison was then carried out between the number of transactions and the transacting prices that were delivered by the manual and automatic methods. The comparison shows the benefits of using the cloud market maker for complex decision making regarding cloud resource selection and dynamic pricing. The experimental setup is explained in the following section; it also covers the manual and automated methods.

4.1 Experimental Setup

The CMM system requires input from both cloud providers and customers. Cloud customers and providers give their requirements and resource details to the system using a standard web-based interface. Input from cloud customers varies in terms of resource details, requirement details, urgency, and pricing. In order to understand what is necessary in terms of creating a realistic simulation of such an environment let us consider an example. Imagine there are five customers (C1, C2, C3, C4, and C5) and each provides their requirements to the system. Eight virtual machine resources (R1, R2, R3, R4, R5, R6, R7, and R8) are available and the customers want to select a suitable resource (provider) from the available options. C1 has enough budget and its requirements match R3. C1 bids for R3 and acquires the resource as it has a sufficient budget to buy it. C2 and C3 fail to get a resource because currently the available resources do not match their requirements. Both C4 and C5 bid for R5. C5 wins the auction for the resource due to it having a higher budget and a close mapping between its requirements and R5. C4 has enough time available to get the resource so it waits for a new resource to become available as remaining resources (R1, R2, R4, R6, and R7) do not match its requirements.

The previous example demonstrates one of the possible scenarios that can occur in a real-time environment. Many such situations can occur in dynamic real environments and therefore a range of simulated inputs from both customers and providers are required in order to carry out a proper evaluation of the cloud market maker system. Furthermore, the decision of the system is influenced by the current market circumstances. In real-time the load on the system is not likely to remain constant, this is due to dynamic nature of the cloud. A situation can occur where large numbers of resources are available for auctions and customers need to consider them all before making a decision. Similarly, the conditions for providers are also dynamic. It is possible that for a particular provider, the numbers of requests surge at one time point. In such a case the price may fluctuate quite wildly In order to test the system against the main real-time scenarios simulated inputs were given to it. A wide range of possible real-time scenarios were carefully considered during the design of the experiments in order to evaluate the system as accurately as possible. This was because we could not ask cloud providers such as Amazon to try out our marketplace on their live systems as this would have been too risky financially for them. The simulation allows us to quantify the performance of the system prior to carrying out live tests.

During experimentation a range of different sets of inputs were given to the system in order to find out its capability of selection and price adjustment based on current market circumstances. A number of test cases were designed. Every test case contained two lists. One of the two lists contained the customers' information and their requirements details. The other list contained the providers' information and their resource details. These data sets were given as input to the system; the output shows the number of successful transactions and the transacting prices. The test cases help to assess the system's capability against a human decision maker given a range of different input sets.

To simulate the dynamic nature of the cloud, variable levels of load were placed on the system. In order to assess the ability of the system to handle dynamic load conditions, data sets of different sizes were given to the system. The data sets were categorized in four major categories, small, medium, large, and extra- large. Each small dataset contained maximum of 10 providers or customers. For medium, large, and extra-large datasets the maximum numbers of providers and customers were 25, 50, and 100. In order to keep the survey within capability of the people who were doing it, a maximum of 100 providers and customers were considered. Further, by list of providers we mean the total number of resources listed by providers. It is likely that each provider will list more than a single free resource at any given time. By considering a range of inputs to the system we can understand how it performs in various situations.

Each category (small, medium, large, and extra-large) of the test cases considers three major scenarios of types of input to the system i) small number of customers and large number of providers ii) large number of customers and small number of providers and iii) equal number of both. For each case the number of possible transactions is either the minimum number of providers or the minimum number of customers. For example, suppose there are 10 providers and 5 customers then the maximum number of transactions that are possible can only be 5. In the case of equal numbers, either the number of customers or the providers can be considered. Taking this into consideration, data sets covering each scenario were used in the experiments. The datasets were first evaluated by manual approach and then by our system. The manual and automated approaches are now explained in greater detail.

4.1.1 Manual Approach

As previously mentioned, the CMM system was compared to a manual system in order to carry out the evaluation. The manual mapping allows us to determine how effectively humans can map the resources when given large number of options. It shows the human's capability to make decision regarding the best cloud resource selection for their requirements. It further helps to clarify the benefits of the cloud market system for cloud users. The manual approach was quite challenging as the manual mapping of customers and providers can be done in a number of ways because every customer may have different preferences or methods. In order to explore this, ten potential customers were interviewed to capture their buying behavior and selection criteria. To ensure fairness, the potential customers that were selected had no knowledge of system selection criteria. Every customer was interviewed individually and buying behavior was recorded.

From the interview it was found that most of the customers preferences were i) Price ii) Brand iii) Quality iv) Demand level/Time and v) Ease of access. The outcome of interviews is shown in Figure 4. During the interview process many customers inquired about the brand of the provider. As this is somewhat subjective it is not presently included in the characteristics of our system, this parameter is not considered in the manual method. The other reason for not including this parameter is to be fair to emerging providers. It is also difficult to consider timing in the manual approach because the time considered in prototype spans to 16 hours and it is difficult to express in manual approach. During interviews it became clear that the customers' priority was to get resources quickly. The aim of this interview was to capture their criteria for selection which was then used as a foundation for the manual process. After the interview, each customer was given the lists of providers and customers. They were asked to map the two lists. After the manual mapping the same lists as provided in manual mapping were executed by the system.

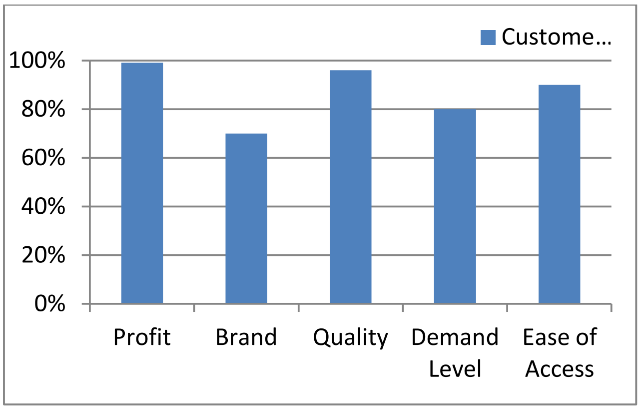

Figure 4: Customers' Preferences

4.1.2 Automated Approach

The automated approach was also evaluated using sets of system inputs. The output was then analyzed in order to determine capabilities of the system. To ensure that the results were fair it was necessary that the experimental setup for both manual and automated approach was the same. However for the automated approach, a number of extra scenarios can occur in a real environment. This is because input from providers regarding new resources can be given to the system anytime during the urgency timeframe of the customer. This is how the system would work in a real-time environment. However, the system cannot be evaluated with this setup because this is not really possible to do this with the manual mapping process. Therefore, it is necessary to exploit the setups are possible for both manual and automated approaches. Different real-time scenarios and the choices that were made during the evaluation are now explained.

It is possible that during the urgency period of the customer i) an auction isn't available at the start i.e. when a customer joins the system ii) auctions start from time to time iii) a large number of auctions are created at the start and no new auctions are added in the middle of the customer's urgency period iv) a large number of auctions are opened at the start and new auctions keep being added in the middle of customer's urgency period v) no auction opens and urgency of customer is reached thus producing zero transactions vi) auctions are opened when the urgency of customers is about to end. In order to fairly evaluate the system using the manual and automated approaches, scenario iii was the only option that was possible with both approaches without producing trivial results. If other scenarios had been considered then the results could have been biased against the manual approach. It is possible, in automated approach, that auctions are added/opened in the middle in order to increase the number of transactions. This is not however possible in a manual approach. Further, the chosen scenario allows us to determine how successfully the system selects the appropriate cloud resource from a large resource pool. The goal of the evaluation is to understand the system's capability to make complex decisions.

For every category (small, medium, large, and extra-large) of the dataset ten experiments were conducted to verify the outcome for the manual and automated processes. Firstly the number of transactions and transacting price for each dataset was calculated manually. Then each test case was executed automatically for six urgency levels which were mentioned previously (Section 3.4). The number of transactions and transacting price (for both manual and automated approach) is the average that is calculated from the experiments.

4.2 Results

The results allow us to compare the manual and automated systems. The comparison allows us to confirm the benefits of the cloud market maker system for cloud users; it shows how the CMM system assists in the selection and dynamic price adjustment without human involvement. The criteria, i.e. the number of successful transactions and the transacting price, were previously explained.

4.2.1 Number of Successful Transactions

The number of transactions delivered by the manual and automated outcomes is shown in Figures 5, 6, 7 and 8. Figure 5 presents the results for small data sets; Figure 6, 7, and 8 shows the output of medium, large, and extra-large data sets.

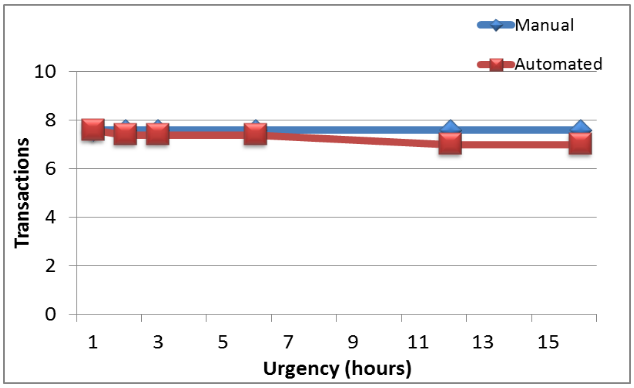

Figure 5: Number of Transactions for Small Data Set

From Figure 5 it is clear that if the dataset is small, then in some experiments (but not all) the manual and automated approaches show similar number of transactions; this is due to the fact that creating mapping manually is not that difficult because of the limited options available. The precise reason for a low number of transactions at low urgency levels (16 hours, 12 hours) for automated approach is explained later on.

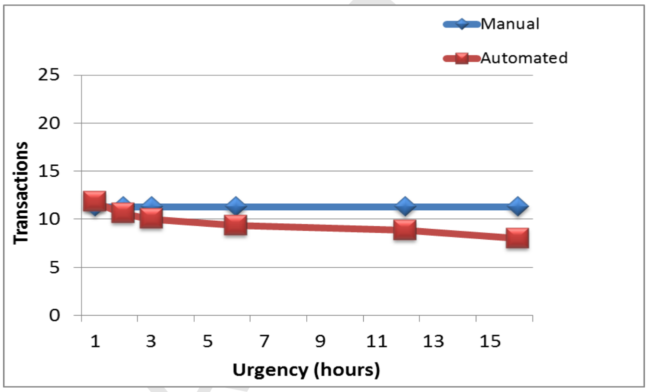

Figure 6: Number of Transactions for Medium Data Set

The results for the medium dataset are shown in Figure 6, almost same trend as for small dataset has been observed. But it has been observed that for the immediate urgency level the automated approach performs a little better than the manual one. The difference is not that large because traversing and manually mapping the list of twenty five customers and providers is not that challenging.

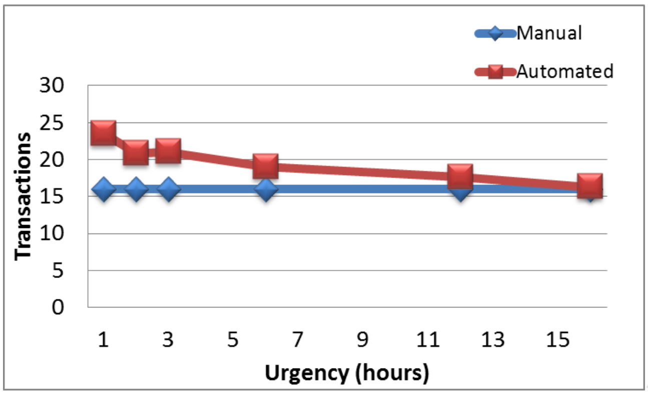

Figure 7: Number of Transactions for large Data Set

When the number of customers and providers are increased to 50 as shown in Figure 7, the automated approach outperforms manual approach. From the above figure it is clear that when number of options increases, manual mapping becomes much more difficult.

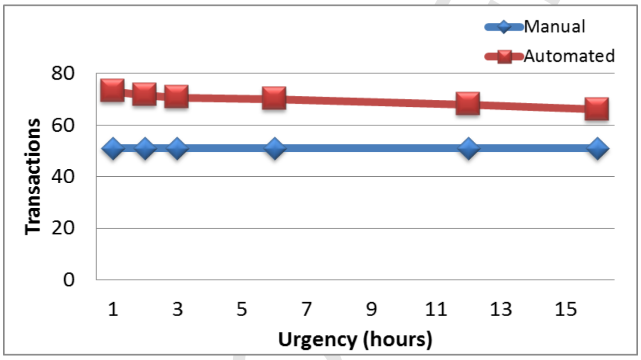

Figure 8: Number of Transactions for Extra-Large Data Set

In the case of the extra-large dataset, as shown in Figure 8, it has been shown that the automated approach greatly outperforms the manual approach. During the manual approach when the dataset is extra-large, the manual mapping was done without really considering and traversing every option. Moreover, it has been observed that when numbers of options increases then only first few items in the lists were traversed in the manual approach thus reducing the number of transactions. The manual mapping of the large and extra-large data sets showed a significant increase in the degree of difficulty. The mapping showed that humans find it difficult to look for each and every option before making the final decision and this may not necessarily help them to select the best option.

The results show that different datasets influence the decisions of the human subjects involved in the manual mapping experiment. This is because the complexity regarding the selection and decision making increases with the availability of large number of options. For example consider a situation when a customer needs to buy an item ‘A’ from the market. The customer will easily make the decision if there are limited options available. In a case with a large number of possible options, the customer will find it quite difficult to select and decide and it may not necessarily results in selection of most suitable one. The same observation was made during manual mapping. The mapping was easily carried out for small number of possibilities; however the difficulty level increased with large number of options. Furthermore, humans also lose interest when they need to compare each and every option which may result in selection of the resource that is not the most suitable one. Therefore, the number of transactions produced by manual approach is lower for large datasets. This demonstrates the suitability of automated systems for such complex decisions.

The results further show fewer transactions for the automated approach when the urgency level is low. This is because of the experimental setup and design of the system. The system design shows that during selection, customer agents firstly compare the highest bid in an auction with customer’s budget. For example if the urgency or remaining time is 16 hours then customer agent select only that resource that is less than 20% of customer’s budget. The customer agent does not select that resource which costs above 20% of budget at this urgency level. The customer agent then gradually increases its threshold in relation to customer’s remaining time to get the resource. At low urgency levels the goal of customer agent is to get the most economical resource for customer. Therefore, it does not tend to select the resource above the specified thresholds (in selection and bidding algorithm). This naturally reduces the number of transactions at these urgency levels. When the remaining time (to acquire the resource) reduces, the customer agent increases its threshold. This often results in a large number of transactions at the higher (immediate, 2, and 3 hours) urgency levels.

The other factor which has influenced the number of transactions at low urgency levels is the experimental setup. This setup was chosen to enable a fair comparison between manual and automated approaches. If other setups were used, as mentioned previously, then the outcome could be a different pattern depending on the time (during customers' urgency) of inputs to the system. This is especially the case if new auctions are added periodically during the customers' urgency period. The comparison of automated and manual with such setups was not possible and this is why only this setup was considered.

4.2.2 Transacting Price

The literature shows that most IaaS providers charge fixed prices for their instances. In order to find the benefits of dynamic pricing for the cloud providers, it is important to make comparison between fixed and dynamic pricing. The manual mapping of customers to providers examines static pricing approaches because the mapping shows the willingness of customers to pay the listed price. The price produced by the system is the dynamic price. Figure 9 shows the transacting price delivered the manual (Static Pricing) and automated approach (Dynamic Pricing).

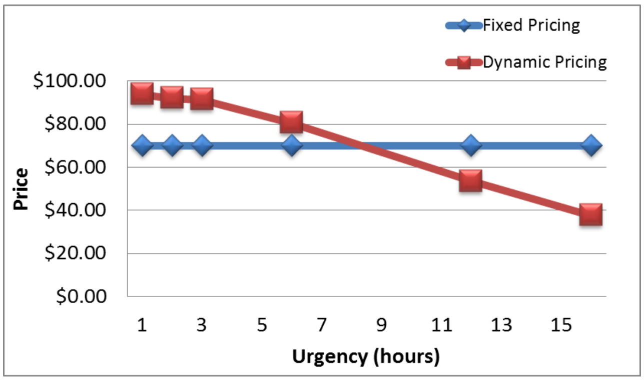

Figure 9: Transacting price

It can be seen from Figure 9 that our automated approach produces a low transacting price at low urgency levels. This is because the system tends to select the most economical resource at these urgency levels. Moreover, the customer agent does not tend to select that resource which costs above a specific threshold. For example if urgency level is low then customer agent will bid to a low threshold. For example, if customer's remaining time or urgency is 12 hours then our system bids up to 40% of customer's budget. This results in low transacting price at low urgency levels. Further the low number of transactions, in automated approach, at low urgency level is another factor for low transacting price. The reason of low number of transaction at these urgency levels has been explained previously (Section 4.2.1).

As customers' urgency increases, the customer agent bids to its maximum budget which increases the transacting price. This shows the ability of auction mechanism to adjust the price in relation to the current demand. The price fluctuates but the auction mechanism ensures that resource is delivered at a price which is greater than or equal to provider's reserve price. This ensures that the resource is within customer's budget. The results also show that average transacting price for manual approach is same for all urgency levels. This is because the manual approach does not adjust the price in relation to current demand level. It does not matter in the manual approach which urgency level is selected as it transacts at a fixed price. Further, the results demonstrate that the dynamic approach can augment the revenue of cloud providers in some cases. The results also demonstrate the ability of our system to dynamically adjust the price in relation to current market situation. It is worthwhile to mention that in automated approach low transacting price at low urgency level does not indicate that static pricing outperform dynamic pricing at these urgency levels. The low transacting price at these urgency levels is due to low number of transactions and the experimental setup that was explained in section 4.1.

4.3 Analysis

The evaluation shows the system's capability to dynamically adjust the cloud resource price and discover the most suitable resource for cloud customers. The number of successful transactions shows the system's ability to select the most suitable resource by traversing all the possibilities. The results demonstrates that a manual approach for large numbers of options is not suitable because humans find it difficult and time consuming to compare each and every option for selection of best choice for their requirements. The results thus clearly demonstrate the benefits of this system for the cloud customers. The transacting price was the evaluation criterion that was used to find the suitability of dynamic pricing for cloud providers. It is clear from the result that dynamic and static pricing produce the same transacting price when the demand does not fluctuate. The transacting price produced by the dynamic approach varies with the fluctuating market conditions. When the demand for resource grows the resource price increases and helps to generate additional revenue for providers. The results clearly show the suitability of dynamic pricing for cloud providers.

Our system is more efficient than a manual system. It traverses every option. It not only selects the best choice but also saves time for users. The average time taken by the system in resource selection is shown in Figure 10. If the resource is available for customer then the system takes much less time in selection than a manual approach. The time increases if no suitable resource is available for respective customer. In such case, the system continuously checks the blackboard for availability of new resource. This naturally increases the time in selection. The graph below shows the system's efficiency in terms of complex decision taking and suitable resource selection from large number of available options.

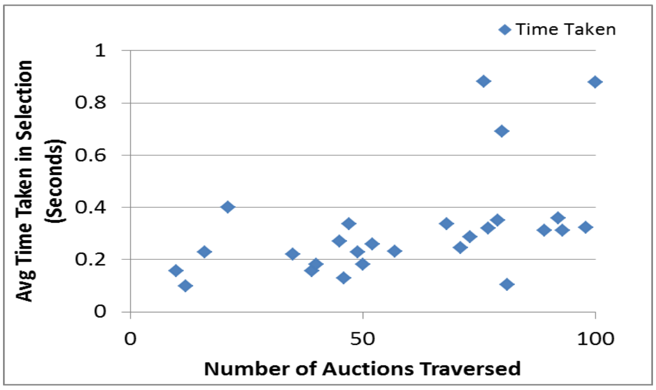

Figure 10: Average Time Taken in Resource Selection

5. Conclusion and Future Work

The large upfront cost and changing environment of the cloud requires a dynamic pricing to maximize the revenue of the cloud providers. Providers also price their resources differently which makes it difficult for consumers to identify or decide which cloud provider is most suitable for their workload. Users may have to rely on what knowledge they have about the cloud providers which may drastically limit their choices. To assist customers in decision making and providers in dynamic price adjustment, this paper presented the Cloud Market Maker System (CMM). It employs a multi-agent multi-auction approach for creating an automated marketplace for cloud users which were not possible with other existing systems. In order to assess the implemented approach, it is important to analyze the suitability of dynamic pricing for the cloud environment and the number of customers and providers it satisfies in terms of resource discovery. To do this, several parameters such as the price delivered by the system and numbers of successful transactions were analyzed. These parameters were first calculated manually then were also carried out using our system. The results demonstrate the benefits of dynamic pricing for the cloud environment. The results also demonstrate the system's ability to discover suitable resource for cloud customers. Moreover, in the interviews with customers it was observed that most of them prefer a known and trusted provider. This system will therefore help emerging cloud providers to gain recognition.

From the experimental results, it is clear that the cloud market maker effectively provides decision support to users in the selection of a suitable provider. After selection and price determination, a Service Level Agreement (SLA) is agreed between the customer and the selected provider using standard methods. Future work will consider how additional parameters and an SLA can be included within the model. In the current implementation of the system cloud providers and customers are not exposed to the internal workings of the system. These design features remove the some risks that are associated with the auction process. However, future work will also consider the issues related to security and privacy of providers and customers.

Contributions

Barkha Javed conducted the research and proposed the idea of Cloud Market Maker. She worked under the supervision of Dr. Peter Bloodsworth. He worked closely with her and helped her throughout her research work. Dr. Raihan and Dr. Kamran helped technically and in writing the paper. Prof. Omer Rana provided general guidance and helped in refining the work. All authors read and approved the final manuscript.

References

[1] Armbrust. M., et al, “Above the Clouds: A Berkeley View of Cloud Computing”, Technical Report No. UCB/EECS-2009-28, 2009.

[2] “UK Cloud adoption and trends for 2013”, Paper 8 by Cloud Industry Forum. http://www.fasthosts.co.uk/downloads/white-papers/cif-whitepaper-8-uk-cloud-adoption.pdf [Accessed August 20, 2013].

[3] Marston. S., et al, “Cloud computing-The business persective”, Decision Support Systems 51 (2011) 176–189.

[4] http://betanews.com/2012/09/18/public-cloud-spending-reaches-109b-tops-206b-by-2016/ [Accessed 28 Feb 2014 05:10:13 GMT]

[5] McAfee, R. P and V. te Velde, "Dynamic pricing in the airline industry", T. J. Hendershott, ed. Handbook on Economics and Information Systems. Elsevier, 2007.

[6] http://www.cloudorado.com/, [Accessed 8 May 2013, 10:17:23 GMT].

[7] http://aws.amazon.com/ec2/spot-instances/ [Accessed July 24, 2013].

[8] Xu. H and Li. B, "Maximizing Revenue with Dynamic Cloud Pricing: The Infinite Horizon Case", IEEE International Conference on Communications(ICC), 2012.

[9] Wang. W., et al, "Revenue Maximization with Dynamic Auctions in IaaS Cloud Markets", IEEE/ACM 21st International Symposium on Quality of Service(IWQoS), 2013.

[10] Anandasivam. A and Premm.M, "Bid price control and dynamic pricing in clouds", 17th European Conference on Information Systems, 2009.

[11] Kamra. V., et al, "Cloud Computing and its Pricing Schemes", International Journal on Computer Science & Engineering, Vol. 4 Issue 4, p577-581, Apr 2012.

[12] http://www.cloudharmony.com/clouds [Accessed July 19,2013]

[13] Ang Li, et.al.,"CloudCmp: Comparing Public Cloud Providers", IMC '10 Proceedings of the 10th ACM SIGCOMM conference on Internet measurement , pp.1-14, 2010, Melbourne Australia.

[14] http://www.planforcloud.com/ [Accessed 16 Apr 2014 00:44:07 GMT]

[15] Sycara. K, "Multiagent Systems", AI magazine, Vol. 19, No.2, Intelligent Agents Summer 1998.

[16] Shakshuki. E, et al, "A Multi-Agent System Architecture for Information Gathering", 11th International Workshop on Database and Expert Systems Applications, 2000.

[17] Maes. P,et al, "Agents that Buy and Sell", Communications of the ACM,Vol. 42 Issue 3, March 1999.

[18] He.M and Leung.H, "Agents in E-Commerce: State of the Art", Knowledge and Information Systems, 2002.

[19] Zhang,Z.,et.al, "Multiagent System Solutions for Distributed Computing, Communications, and Data Integration Needs in the Power Industry", In Power Engineering Society General Meeting, 2004.

[20] Chien, C,H., et.al, “Market-Oriented Multiple Resource Scheduling in Grid Computing Environments”, vol.1, 2005.

[21] Chunlin,L, and Layuan,L, “The use of economic agents underprice driven mechanism in Grid resource brokering”, Journal of System Architecture, vol.50, pp.521-535, 2004.

[22] Dominiak.M,et al,“Utilizing Agent Teams in Grid Resource Brokering”,In Multiagent and Grid Systems, 2007.

[23] Talia. D, “Cloud Computing and Software Agents Towards Cloud Intelligent Services”, WOA, pp.2-6, 2011.

[24] Sim. K, “Agent-based Cloud Computing, Services Computing”,*IEEE Transactions on Services Computing, 2011.*

[25] Sim. K,, “Agent-Based Cloud Commerce”, Industrial Engineering and Engineering Management (IEEM), 2009.

[26] Lee. J, “ Recurrent Auctions in E-Commerce”, A Thesis Submitted to the Graduate Faculty of Rensselaer Polytechnic Institute USA, 2007.

[27] He. M, “On Agent-Mediated Electronic Commerce”, IEEE Transactions On Knowledge And Data Engineering, vol. 15, July/August 2003.

[28] Ito. et al., “BiddingBot, “A Multiagent Support System for Cooperative Bidding in Multiple Auctions”, Proceedings of the Fourth International Conference on MultiAgent Systems, pp. 399 - 400, July 2000.

[29] He, M. et al, “A Heuristic Bidding Strategy for Buying Multiple Goods in Multiple English Auctions”, *ACM Trans on Internet Technology,* pp. 465-496, 2006.

[30] Preist. C, “Economic Dynamics of Agents in Multiple Auctions”, In Proc. 5th International Conference on Autonomous Agents, 2001.

[31] He. M, et al, “An adaptive bidding agent for multiple English auctions: a neuro-fuzzy approach”, In *IEEE Conference on Fuzzy Systems,* pp.1519-1524, 2004.

[32] Rogers O and Cliff D, “A financial brokerage model for cloud computing”, Journal of Cloud Computing, Advance, Systems and Applications 1(2):1–12, 2012.

[33] Wu F, Zhang L, Huberman B, “Truth-telling reservations”, Lecture Notes in Computer Science (LNCS) 3828:80–91. doi:10.1007/11600930_9, 2005.

[34] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ri-market-general.html [Accessed 28 Apr 2014]

[35] Cartlidge. J and Clamp. P, “Correcting a financial brokerage model for cloud computing: Closing the window of opportunity for commercialisation”, Journal of Cloud Computing: Advances, Systems and Applications, ISSN 2192-113X, 2014.

[36] http://www.spotcloud.com/ [Accessed 18 Nov 2013 00:42:09 GMT].

[37] Buyya. R, et.al., “InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services”, ICA3PP'10 Proceedings of the 10th international conference on Algorithms and Architectures for Parallel Processing, pp. 13-31, doi:10.1007/978-3-642-13119-6_2, 2010.

[38] Luca. M.D and Cliff. D, “Human-Agent Auction Interactions: Adaptive-Aggressive Agents Dominate”, Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence,2011.

[39] Das.R., et al, “Agent-Human interaction in the continuous double auction”, In Proceedings of the 17th International Joint Conference on Artificial Intelligence, 2001.

[40] http://www.ebay.com/[Accessed 5 Nov 2013]

[41] D. Rudenko and A. Borisov, “An Overview of Blackboard Architecture Application for Real Tasks”, Scientific Proceedings of Riga Technical University, 2007.

[42] http://jade.tilab.com/ [Accessed 5 June 2013]

Ms. Barkha Javed

Barkha Javed completed her M.Sc. degree in Information Technology from NUST School of Electrical Engineering and Computer Science (NUST-SEECS), Pakistan, in 2013. Her main research interests are Distributed Computing and Artificial Intelligence.

Dr. Peter Bloodsworth

Dr. Peter Bloodsworth joined NUST-SEECS (DoC) as a Foreign Professor in autumn 2011. He has a PhD in Multi-agent Systems from Oxford Brookes University in the UK. Dr. Bloodsworth has around 10 years of experience in academia with special interest in multi-agent systems, semantic technologies and distributed computing. Before joining SEECS, he worked as a Research Fellow in the Centre for Complex Cooperative Systems at UWE, Bristol. He has participated in several European projects and was a work package leader in the FP7 funded neuGRID project. He remains active in PhD supervision and is fully qualified to act as director of studies for research degrees in UK universities. His research work is published in various international journals and conferences with over twenty publications thus far. He is a full member of IEEE and served in the program committees of IEEE conferences. He is also a Chartered Member of the British Computing Society (BCS) and continues to act as a reviewer for several IEEE and other international journals.

Dr. Raihan Ur Rasool

Dr. Raihan ur Rasool has been a Fulbright fellow at The University of Chicago, USA where he worked with his fellow colleagues of Argonne National Lab (ANL) on Large Scale Systems. He received his PhD from Wuhan University, China in Computer Engineering in 2007. Raihan holds several distinctions including a 'Best Foreign PhD Student' certificate and Fulbright postdoctoral grant from Department of State, USA. He has worked on various projects related to Grid and Cloud computing, and heterogeneous distributed systems. Raihan worked as Associate Dean at NUST School of Electrical Engineering and Computer Science, Pakistan before moving to King Faisal University (KFU). He is currently serving at KFU College of Computer Science and Information Technology (CCSIT), Saudi Arabia.

Dr. Kamran Munir

Dr Kamran Munir has a PhD (2010) and MSc (2001) in Computer Science. He has over 12 years of experience in academic research and teaching, and his primary research interests include big data, distributed data and knowledge management, provenance management and information retrieval. His research work, comprising over 30 articles, is published in various international conferences and journals and is a reviewer for various IEEE, ACM, Elsevierand other international journals. Dr. Munir has contributed to the various European Commission projects:

Asia Link STAFF, Health-e-Child, and neuGRID/N4U in which he leads the Joint Research Area. He is a member of the Centre for Complex Cooperative Systems at UWE and is co-leader of its Large-Scale Data Management unit.

Prof. Omer F. Rana
Omer Rana is Professor of Performance Engineering at Cardiff School of Computer Science & Informatics. He was formerly the deputy director of the Welsh eScience Centre at Cardiff University where he had an opportunity to collaborate with a number of scientists working in computational science and engineering. He holds a PhD in "Neural Computing and Parallel Architectures" from Imperial College, London University in the UK. His research interests are in the areas of high performance distributed computing, data mining and analysis and multi-agent systems. He is an associate editor of the ACM Transactions on Autonomous and Adaptive Systems, IEEE Transactions on Cloud Computing, series co-editor of the book series on "Autonomic Systems" by Birkhauser publishers, and on the editorial boards of "Concurrency and Computation: Practice and Experience" (John Wiley) & Journal of Computational Science (Elsevier).

*Biographies (Photograph)

Click here to download high resolution image

***Biographies (Photograph)**
Click here to download high resolution image

*Biographies (Photograph)
Click here to download high resolution image

*Biographies (Photograph)
Click here to download high resolution image

*Biographies (Photograph)
Click here to download high resolution image