ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (354-361), Month: April - June 2015, Available at: www.researchpublish.com

Design of Price and Time Slot Negotiation Mechanism for Cloud Service Reservation Based On the Quality of Service

Priyanka G

Department Of Computer Science And Engineering, Visvesvaraya Technological University,Belagavi, Belagavi,India

Abstract: **Cloud service providers currently charge consumers for metered usage based on appropriate pricing model (fixed and spot pricing). When making reservations for cloud services in automatic pricing mechanism, consumers and providers need to establish service-level agreements (SLA) through Price and Time-slot Negotiations (PTN).Whereas it is essential for both a consumer and a provider to reach an agreement on the price of a service and when to use the service (i.e. time slot). Even though these issues are essential, mechanisms to automate the negotiation of price and time slot for cloud services using Quality of Service (QoS) have not been devised. Also existing PTN mechanisms have dealt with advance reservations considering bandwidth or time constraints and SLA negotiation, to date; there is no service reservation system that considers both price and time slot negotiation with QoS. QoS parameters such as service initiation time, penalty rate ratio are included in the PTN mechanism. The consumer can make multi issue negotiation and the provider offers concession for the requested PTNs to improve the resource utilization and acceptance rate. Another novelty of this work is formulating a novel time-slot utility function that characterizes preferences for different time slots. These ideas are implemented in an agent-based Cloud testbed. Experimental results show that PTN agents reach faster agreements and achieve higher utilities than other related approaches.**

Keywords: **Quality Of Service (Qos), PTN Negotiation, Negotiation Strategies, Utility Function, Resource Allocation.**

1. INTRODUCTION

A **Cloud** is a parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resource(s) based on service-level agreements (SLAs) established through negotiation between service providers and consumers. Hence, a Cloud service provision is commonly governed by an SLA .An SLA is a service guarantee that defines a set of quality of service (QoS) constraints such as price or time constraints and specifies how the service is offered. some of the important issues to be considered to establish an agreement between a consumer and service provider for utilization of cloud services are as follows: 1) determining when to use a service (i.e., time slot) and 2) determining the price of the service. Even though these issues are essential, mechanisms to automate the negotiation of price and time slot for Cloud services have not been devised.

Whereas previous works[10] have dealt with advance reservations considering bandwidth or time constraints and considered SLA Negotiation. Till today, there is no service reservation system that considers both price and time-slot negotiations (PTNs).price and time slot have to be negotiated simultaneously because there is an inverse relationship between price and time-slot utilities [i.e. a consumer needs to pay a higher price (obtaining a lower price utility) to use a service at a more desirable time slot (obtaining a higher time-slot utility)]. This proposed work considers a multi-issue negotiation mechanism [1] for PTNs for Cloud service reservations. Whereas Yan et al. and Lang [2] designed multi-issue SLA negotiations for Web services and Grid resource negotiations[4], respectively, their mechanisms are not specifically designed to negotiate price and time slot for Cloud service reservations.

2. OBJECTIVE AND MOTIVATION

The objectives of proposed work are as follows:

1) To devise a QoS based PTN mechanism that includes the design of a novel utility function for characterizing time-slot preferences.

2) To design tradeoff and concession algorithms for the negotiation strategy of consumers and providers.

3) To implement an agent-based Cloud testbed.

4) To evaluate the QoS based PTN mechanism by conducting experiments using the agent-based testbed.

5) To conduct a case study on applying the QoS based PTN mechanism to the pricing of Cloud resources.

One of the challenging issues in Cloud service reservations is devising an appropriate pricing model.Amazon elastic Cloud computing (EC2) provides consumers with both fixed pricing and flexible pricing for leasing virtual machine instances. On-demand instances allow consumers to pay a fixed price by the hour without a long-term commitment and to start the instances immediately. With reserved instances, consumers need to pay a one-time fee for a one- or three-year term but benefit from paying a discounted hourly usage fee within the term. Spot instances enable consumers to bid for unused computing capacity. Instances are charged at the spot price set by Amazon. The spot price changes periodically, depending on the supply and demand for spot instances. Consumers' requests can be fulfilled if their maximum bid prices are above the spot price and they can run their applications on the spot instances for as long as their maximum bid prices exceed the current spot price. All consumers will pay the same spot price for that period even if their maximum bid prices are above the spot price. But consumers generally cannot plan when to start and terminate their applications. Consumers' applications running on spot instances can potentially be interrupted. When a consumer is running spot instances and if the current spot price exceeds the consumer's maximum bid price, the instances will be terminated without warning. This provides the motivation for designing a mechanism that will allow a consumer to reserve a preferred time slot by negotiating the price for reserving the time slot with the service provider. By having such a negotiation mechanism for flexible pricing of cloud resources, providers can benefit from more efficient utilization of their resources, and consumers can benefit from cost saving in some situations and having more flexibility in planning the start and termination times for running their applications. In general, the demand for resources tends to fluctuate with time [5]. Whereas Cloud resources are more likely to be idle at nonpeak time (e.g., at night or during weekends), providers may find it difficult to accept additional requests at peak time. Hence, providers may prefer to schedule resources to meet consumers' requests at nonpeak time to reduce the time of having idle resources. This increases their chance of generating more revenue.

3. RELATED WORK

Since this work explores the issue of designing a multi-issue Cloud negotiation mechanism, areas related to this work include the following: 1) Grid resource negotiation; 2) concurrent negotiation for resource co allocation;

Grid Resource Negotiation: Sim [4] reviewed and compared state-of-the-art approaches for Grid resource negotiation mechanisms in terms of strategies and protocols. Some of the works surveyed in [4] are as follows. Lang [2] adopted a two-phase bargaining protocol for Grid resource negotiation. In [2], the negotiation protocol consists of the following: 1) a distributive negotiation phase, in which self-interested agents adopt heuristic strategies to iteratively exchange bids, and 2) an integrative negotiation phase, in which agents attempt to find joint gains while attempting to maintain the utility distribution outcomes in the distributive negotiation phase. In the Policy-driven Automated Negotiation Decision-making Approach (PANDA), Gimpel et al[7] adopted a rule-based framework for negotiation of service contracts.

Concurrent Negotiation for Resource Co allocation: venugopal et al. [8] adopted a protocol for negotiating SLAs based on Rubinstein's alternating offers protocol [9] for the advance reservation of grid resources. The protocol in [8] considered negotiation for time slots and the number of resources (i.e., grid nodes) between a consumer and a provider in a grid environment. In [8], the consumer side of the protocol was implemented through a grid bus broker, and the provider side of the protocol was implemented within a .net-based enterprise grid system called aneka. Aneka includes a reservation manager that allocates reservations and enforces them on allocation managers of each grid node.

4. SYSTEM OVERVIEW

As a result **QoS Based Price and Timeslot Negotiation Mechanism** is designed. The consumer and service provider are participating in cloud negotiation through the cloud registry. The registry is an information repository. Provider acts as a service advertiser while consumer discovers the services from environment. Tradeoff and concession making algorithms are implemented for PTN. Cloud reservation is done in memory array. Single issue and multi issue negotiation is considered for QoS like deadline, budget, penalty rate ratio, service initiation time, acceptance rate, resource utilization etc., A coordinator is introduced which distribute the applications across different data centers which enabling SLA's for improving application's performance, reliability and scalability. PTN mechanism follows the negotiation protocol in which the agent makes negotiation in alternate rounds. It will accept when both the consumer agent and provider agent reached in an agreement for price and time. The negotiation fails when one of agent's deadlines expires before reach the agreement.

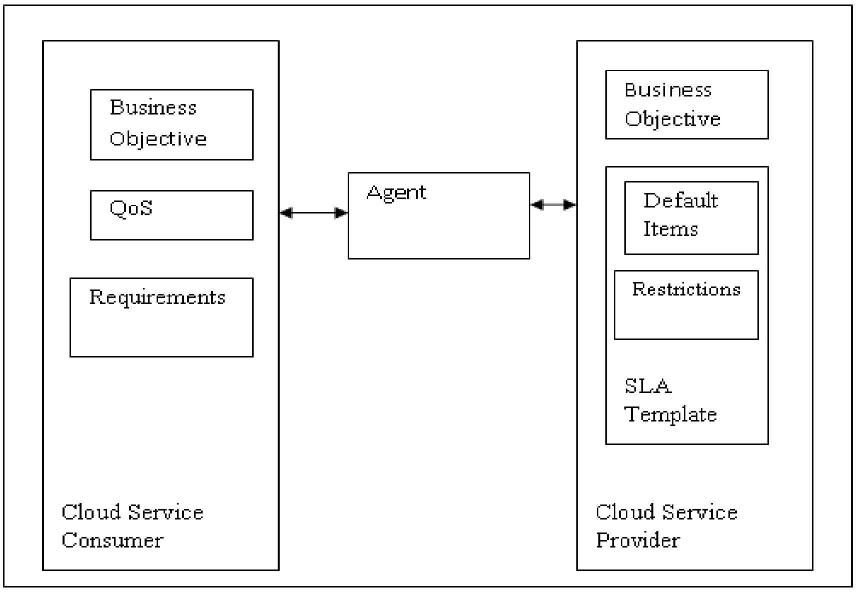

Figure. 4. 1 Schematic diagram of the QoS based PTN

Figure 4.1 represents the agent acts as a broker between the consumer and provider. Consumer is defined with business objective, requirements and QoS. Provider also has a business objective, SLA terms and restrictions. The agent reserves the resources based on the requirements of the consumer

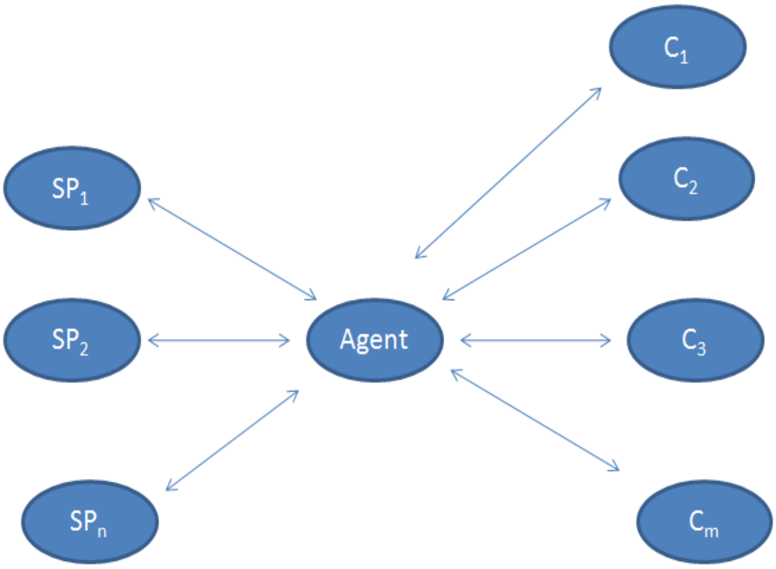

Figure. 4. 2 Experimental setup

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (354-361), Month: April - June 2015, Available at: www.researchpublish.com

Figure 4.2 shows the experimental setup of QoS based PTN mechanism. There may be 'n' number of service providers and 'm' number of consumers. All the consumers and providers must register their details in the agent repository. Then the agent allocates the resources to the consumers when the negotiation succeeds.

QoS parameters [6] considered in the proposed work are mentioned below

• Service Initiation Time: Maximum time user would like to wait for the initiation of the request.

•Penalty Rate Ratio: A ratio for consumers' compensation if the provider misses the deadline.

The PTN mechanism consists of the following:

1) An Aggregated Utility Function: utility function U(x) represents an agent's level of satisfaction for a negotiation outcome *x*. Since each Cloud participant has different preferences for different prices and time slots, a price utility function, a time-slot utility function, and an aggregated utility function are used to model the preference ordering of each proposal and each negotiation outcome.

Price utility function: Whereas consumers prefer the cheapest price for leasing a service, providers want to sell their services at the highest prices. Let IP_C and RP_C (respectively, IP_P and RP_P) be the most preferred (initial) price and the least preferred (reserve) price of a consumer (respectively, provider) agent. Let P be a price that both agents reach an agreement.

The Equation (4.1)depicts the price utility function for consumer.

$$U_P^C(P) = \begin{cases} u_{min}^P + (1 - u_{min}^P) \cdot \left|\frac{RP_C - P}{RP_C - IP_C}\right|, & IP_C \leq P \leq RP_C \\ 0, & \text{otherwise} \end{cases} \quad (4.1)$$

Where,

IPc = Initial price of customer

RPc = Least preferred price of customer

U^P_{min} = Minimum utility (eg. 0.01)

$$U_P^P(P) = \begin{cases} u_{min}^P + (1 - u_{min}^P) \cdot \left|\frac{P - RP_p}{IP_p - RP_p}\right| & , RP_p \leq P \leq IP_p \\ 0 & , \text{otherwise} \end{cases}$$

The Equation depicts the price utility function for provider. (4.2)

Where,

IPp = Initial price of customer

RPp = Least preferred price of provider

P =price that both agents reach an agreement

U^P_{min} = Minimum utility (eg. 0.01)

Time-slot utility function: Time slot is also another important consideration for Cloud service reservations.A consumer can have multiple sets of acceptable time-slot preferences. A provider's time-slot preferences are based on the service demand, temporal ordering and fitting job size.

In Equation (4.1) the time slot utility function considering service initiation time for PTN mechanism is devised.

$$U_T^C(T)^x = \begin{cases} u_{min}^T, & R_{st} \leq R_{at} \leq R_{mt} \\ 0, & \text{otherwise} \end{cases} \quad (4.3)$$

where,

Rst = Request submission time

Rat = Request actual start time

Rmt = Request maximum start time

Rmt = Rst + Swt

Swt = Service waiting time

$U_T^C(T)^x$ = Consumer's time slot utility function

U_{min}^T = Minimum utility

This time slot utility function helps to derive the utility value of each service provider based on the service initiation time given by the customer. The broker compares the utility valued from the entire service provider. Then it chooses a provider for completing the request.

In Equation (4.2) the time slot utility function considering penalty rate ratio for PTN mechanism is devised.

$$U_{PR}^C\,(PR)^x = \begin{cases} U_{min}^T, & M_c < P_R \\ 1, & M_c = P_R \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

where,

$U_{PR}^C\,(PR)^x$ = Consumer's utility function for penalty rate ratio

M_C= Missing Count

P_R= Penalty rate

U_{min}^T = Minimum utility (e.g 0.01)

This time slot utility function helps to derive the utility value of each service provider based on the penalty rate ratio given by the customer. The broker compares the utility valued from all the service provider. Then it chooses a provider for completing the request.

2) Negotiation Strategies: In the QoS Based Mechanism bilateral negotiations [3] between a consumer and a provider is considered, where both agents are sensitive to time and adopt a time-dependent concession-making strategy for PTNs. Since both agents negotiate on both price and time slot, generating a counter proposal can be making either a concession or a tradeoff between price and time slot. Hence, an agent's strategy for multi-issue negotiation is implemented using both tradeoff algorithm and concession making algorithm.

Tradeoff algorithm: The novelty of this work is adopting a new tradeoff algorithm with QoS parameters, called a "burst mode" proposal, which is designed to enhance both the negotiation speed and the aggregated utility. In the burst mode, agents are allowed to concurrently make multiple proposals, with each proposal consisting of a different pair of price and time slot that generates the same aggregated utility. These concurrent proposals differ from each other only in terms of the individual price and time-slot utilities.

Algorithm for Burst Mode Proposal

Input: Request list from consumers, resource list from providers

Output: Allocated VM list

1. Get the price and time of all Providers registered in the PTN
2. For each consumer get the multiple proposal which consist of price, time and QoS
3. Agent checks for a flexible resource for a given request
4. If match found based on service initiation time and penalty rate ratio then

5. Allocate VM

6. Else

7. Call negotiation protocol

Negotiation protocol: The negotiation protocol of the PTN mechanism follows Rubinstein's alternating offers protocol in which agents make counteroffers to their opponents in alternate rounds. Both agents generate counteroffers and evaluate their opponent's offers until either an agreement is made or one of the agents' deadline is reached. Counterproposals are generated according to the negotiation strategy. If a counterproposal is accepted, both agents found a mutually acceptable price and time slot with the QoS given by the user. If one of the agents' deadline expires before they reach an agreement, the negotiation fails.

Concession making: The concession-making algorithm determines the amount of concession for each negotiation round, which corresponds to the reduction in an agent's expected total utility based on the QoS Parameters like service initiation time and penalty rate ratio. Agents in this work adopt the time-dependent strategies in to determine the amount of concession required for the next proposal.

Negotiation protocol

Input: Request list from consumers, resource list from providers

Output: Allocated VM list

1. Agent contacts the provider

2. If specification matches then

3. Agent asks how much concession Provider can make

4. If the concession made is accepted by the consumer then

5. Agreement made and protocol stops

6. Else

7. Next round of negotiation is called

5. RESULT ANALYSIS

To evaluate the performance of the QoS based PTN mechanism, Resource Utilization and Acceptance rate are used.

Resource utilization is the total amount of resources actually consumed, compared against the amount of resources planned for a specific process.

Acceptance rate is the ratio of the number of request accepted to the total number of request submitted by the consumer

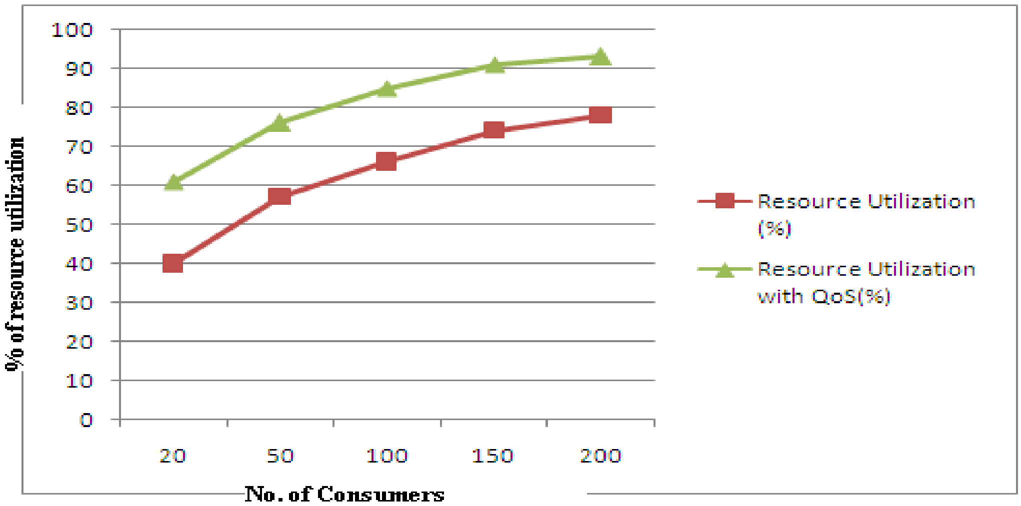

Figure.5. 1 Resource Utilization

From the Figure 5.1, it is proved that Resource utilization is poor when QoS is not considered. By implementing the QoS based PTN mechanisms, the resource utilization is improved.

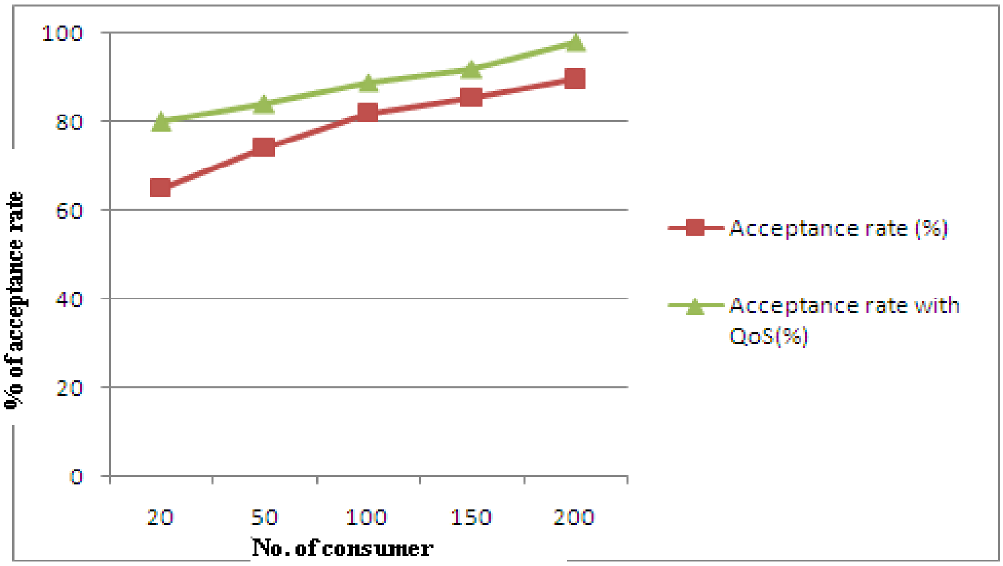

Figure 5.2 Acceptance rate

Figure5.2 shows that acceptance rate increases in QoS based PTN mechanism as compare to existing PTN mechanisms**.**

The proposed QoS based PTN yields higher results in the percentage of resource utilization which influences the following,

a) Reaching quick agreements between the customer and the provider.

b) Acquiring the desired cloud service by the customer.

c) Achieving higher utility by provider.

d) Improving customer satisfaction.

6. CONCLUSION

The reservation manager in cloud environment identifies a common time slot that is acceptable to both consumer and provider agents, but did not consider agents' preferences for different time slots. There are single-issue negotiation mechanisms and multi issue negotiation mechanisms for cloud resource negotiation. To improve the system the PTN mechanism is proposed for both price and time-slot negotiations with QoS parameters.

An enhanced tradeoff algorithm, known as the "burst mode" proposal, is used to increase both the negotiation speed and the aggregated utility. In existing agent can only make one proposal at a time, PTN agents can concurrently make multiple proposals. Thus the effectiveness of PTN mechanism with QoS is improved. Using the PTN mechanism, not only consumers can benefit by paying a lower price but also providers can have more flexibility in allocating consumer's applications to other available time slots, hence achieving more efficient resource utilization.

REFERENCES

[1] S. Son and K. M. Sim, "A multi-issue negotiation mechanism for Cloud service reservation," in Proc. Annu. Int. Conf. CCV, May2010, pp. 123–130.

[2] F. Lang, "Developing dynamic strategies for multi-issue automated contracting in the agent based commercial grid," in Proc. IEEE Int. Symp. CCGrid, Workshop Agent-Based Grid Econ., Cardiff, U.K., May 2005, pp. 342–349.

[3] S. Son and K. M. Sim, "An adaptive tradeoff algorithm for multi-issue SLA negotiation," in Proc. Conf. Grid Distrib.Comput., vol. 121, CCIS, Dec. 2010, pp. 32–41

[4] K. M. Sim, "Grid resource negotiation: Survey and new directions," IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 40, no. 3, pp. 245–257,May 2010.

[5] G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of Cloud computing," Commun. ACM, vol. 53, no. 4, pp. 50–58, Apr. 2010.

[6] B.Krishnakumar, K.Kousalaya, R.S.Mohana, N.Sasipriyaa ,"Design Of QOS Based Price And Time Slot Negotiation Mechanism For Cloud Service Reservations", Volume- 2, Issue- 1, May-2014.

[7] H. Gimpel, H. Ludwig, A. Dan, and R. Kearney, "PANDA: Specifying policies for automated negotiations of service contracts," in Proc. ICSOC, vol. 2910, LNCS, New York, 2003,pp. 287–302.

[8] S. Venugopal, X. Chu, and R. Buyya, "A negotiation mechanism for advance resource reservation using the alternate offers protocol," in Proc. 16th IWQoS, Twente, The Netherlands, June 2008, pp. 40–49.

[9] A. Rubinstein, "Perfect equilibrium in a bargaining model," Econometrica, vol. 50, no. 1, pp. 97–109, Jan. 1982.

[10] Seokho Son and Kwang Mong Sim, Senior Member, IEEE,"A Price- and-Time-Slot-Negotiation Mechanism for Cloud Service Reservations", IEEE Trans. Syst., Man, And Cybernetics—Part B: Cybernetics, Vol. 42, No. 3, June 2012 713