A Novel Model for Competition and Cooperation among Cloud Providers

[1]Mr.Karthikeyan, [2]Amalan V R, [3]Ranjith R, [4]Goutham Raj M, [5]Yoganandam M,
[1]Assitant Professor, [2345]Final Year of Information Technology,
Panimalar Engineering College, Chennai, India.

***Abstract* - Cloud Computing is a novel paradigm for the provision of computing infrastructure, which aims to shift the location of the computing infrastructure to the network in order to reduce the costs of management and maintenance of hardware and software resources. Cloud computing has a service oriented architecture in which services are broadly divided into three categories: Infrastructure as a Service (IaaS), which includes equipment such as hardware, Storage, servers, and networking components are made accessible over the Internet; Platform as a Service (PaaS), which includes hardware and software computing platforms such as virtualized servers, operating systems, and the like; and Software as a Service (SaaS), which includes software applications and other hosted services.**
To obtain accurate estimation of the complete probability distribution of the request response time and other important performance indicators, this model allows cloud operators to determine the relationship between the number of servers and input buffer size, on one side, and the performance indicators such as mean number of tasks in the system, blocking probability, and probability that a task will obtain immediate service, on the other.

***Index Terms* - Cloud computing, Cooperation, Markov Decision Process.**

I. INTRODUCTION

In this project the proposed system, the task is sent to the cloud center is serviced within a suitable facility node; upon finishing the service, the task leaves the center. A facility node may contain different computing resources such as web servers, database servers, directory servers, and others. A service level agreement, SLA, outlines all aspects of cloud service usage and the obligations of both service providers and clients, including various descriptors collectively referred to as Quality of Service (QOS). QOS includes availability, throughput, reliability, security, and many other parameters, but also performance indicators such as response time, task blocking probability, probability of immediate service, and mean number of tasks in the system, all of which may be determined using the tools of queuing theory. That task will be processed in corresponding cloud server based on user category where scaling depend on it.

The competition among providers leads to the dynamics of cloud resource pricing. Modelling this competition involves the description of the user's choice behavior and the formulation of the dynamic pricing strategies of providers to adapt to the market state. To describe the user's choice behavior, we employ a widely used discrete choice model, the multinomial logit model which is defined as a utility function whose value is obtained by using resources requested from providers. From the utility function, we derive the probability of a user choosing to be served by a certain provider. The choice probability is then used by providers to determine the optimal price policy. The fundamental question is how to determine the optimal price policy. When a provider joins the market, it implicitly participates in a competitive game established by existing providers. Thus, optimally playing this game helps providers to not only survive in the market, but also improve their revenues. To give providers a means to solve this problem, we formulate the competition as a non-cooperative stochastic game. The game is modelled as a Markov Decision Process (MDP) whose state space is finite and computed by the distribution of users among providers.

On the second problem, the cooperation based on a financial option allows providers to enhance revenue and acquire the needed resources at any given time. The revenue depends on the total operation cost which includes a cost to satisfy users' resource requests (i.e., cost for active resources) and another cost for maintaining data center services (i.e., cost for idle resources). We address the problem of cooperation among providers by first employing the learning curve to model the operation cost of providers and then introducing a novel algorithm that determines the cooperation structure. The cooperation decision algorithm uses the operation cost computed based on the learning curve model and price policies obtained from the competition part as parameters to calculate the final revenue when outsourcing or locally satisfying users' resource requests. The cooperation among providers makes the cloud market become a united cloud environment, called Cloud of Clouds environment as illustrated in *Fig. 1*. In this architecture, the Cloud of Clouds Broker is responsible for coordinating the cooperation among providers, receiving users' resource requests and also doing accounting management.

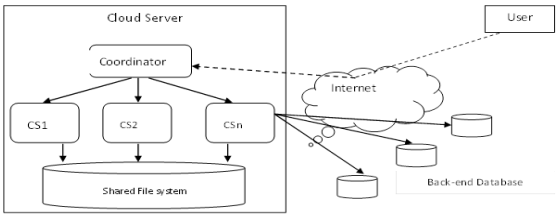

Fig. 1: Overall architecture of a Cloud of Clouds system.

The rest of this paper is organized as follows. After discussing related work in our assumption and the relevant models for users' resource requests, providers' operation costs and revenues. The game formulation for competition among providers. In this method for solving the stochastic game which results in the MPE presents the model and algorithm for cooperation among providers. This presents the numerical simulations and analysis of results that assess the validity of the proposed model.

II. RELATED WORK

Dynamic pricing and competition

Dynamic pricing in the cloud has gained considerable attention from both industry and academia. Amazon EC2 has introduced a "spot pricing" feature for its resource instances where the spot price is dynamically adjusted to reflect the equilibrium prices that arises from resource demand and supply. Analogously, a statistical model of spot instance prices in public cloud environments has been presented in, which fits Amazons spot instances prices well with a good degree of accuracy. To capture the realistic value of the cloud resources, the authors of employ a financial option theory and treat the cloud resources as real assets. The cloud resources are then priced by solving the finance model. Also based on financial option theory, in, a cloud resource pricing model has been proposed to address the resource trading among members of a federated cloud environment. The model allows providers to avoid the resource overprovisioning and under provisioning problems. But there is an underlying assumption is that there are always providers willing to sell call options.

In and, the authors presented their research results on dynamic pricing for cloud resources. While studied the case of a single provider operating an IaaS cloud with a fixed capacity, focused on the case of an oligopoly market with multiple providers. However, both and make the assumption that the user's resource request is a concave function with respect to resource prices. The amount of resources requested will decrease when prices increase. This assumption is not practical when users have a processing deadline or architectural requirements for their execution platform. Users therefore have to request the required amount of resources no matter what the prices are. In and, the authors presented auction based mechanisms to determine optimal resource prices, taking into account the users budgetary and deadline constraints. However, they considered the pricing model of only one provider. In contrast, we consider the realistic case of the current cloud market with multiple providers. In addition, users may have their preferences in choosing to be served by particular providers.

Game theory in utility computing

Game theory has been widely applied in economic studies for dynamic pricing competition. In utility computing, game theory has been applied to study different issues: scheduling and resource allocation dynamic pricing and revenue optimization. In a game theoretic resource allocation algorithm has been proposed to minimize the energy consumption while guaranteeing the processing deadline and architectural requirement. In a user oriented job allocation scheme has been formulated as a non-cooperative game to minimize the expected cost of executing users' tasks. The solution is a Nash equilibrium which is obtained using a distributed algorithm. However, none of these works considered the user's choice behavior, although some of them assume that resources are owned by different resource owners.

Cooperation among providers

Cooperation among providers in cloud computing has been extensively studied with two research approaches: cloud federation and coalitional formation based on coalitional game theory.

The idea of federating systems was originally presented for grid computing. For instance, in the authors used the federation approach to get more computing resources to execute large scale applications in a distributed grid environment. The application of the federation approach in the cloud was initially proposed within the RESERVOIR project. Nevertheless, the aforementioned works focused only on aggregating as much resources as possible to satisfy users' resource requests. They did not consider the economic issue is one of the intrinsic characteristics of cloud computing. In the authors presented an economic model along with a federated scheduler which allow a provider, operating in a federated cloud, to increase the final revenue by saving capital and operation costs.

Based on coalitional game theory, studied the problem of motivating self-interested providers to join a determined horizontal dynamic cloud federation platform and the problem of deciding the amount of resources to be allocated to the federation. The authors of used the coalitional game approach to form cloud federations and share the obtained revenue among coalition members fairly. However, this work did not consider the operation cost of providers which is an important factor in the economic model. In this paper, we consider the realistic case of the current cloud market where providers may have different operation costs. Cooperation among providers may reduce the operation cost and therefore improve the final revenue.

Clustered Classification

The server calculates which cloud doing which job. That is Monitoring cloud access, cost calculation and equal sharing of jobs in cloud. Here multiple servers formed into cluster formation. Group applications into service Performance of classes which are then mapped onto server clusters which parses application level information in Web requests and forwards them to the servers with the corresponding applications running. The switch sometimes runs in a redundant pair for fault tolerance. Each application can run on multiple server machines and the set of their running instances are often managed by some clustering software.

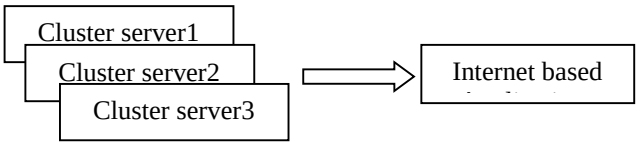

Fig. 2: Clustered Classification

Differentiated Services

After the servers may be clustered then allocate the task which can be assigned to each server and calculate the performance and priority. Each server machine can host multiple applications. The applications store their state information in the backend storage servers. It is important that the applications themselves are stateless so that they can be replicated safely.

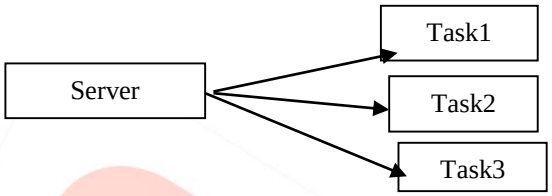

Fig.3: Differentiated Services

Distributional Classes

The system under consideration contains m servers which render service in order of task request arrivals (FCFS).The capacity of system is m þ r which means the buffer size for incoming request is equal to r. As the population size of a typical cloud center is relatively high while the probability that a given user will request service is relatively small, the arrival process can be modelled as a efficient process.

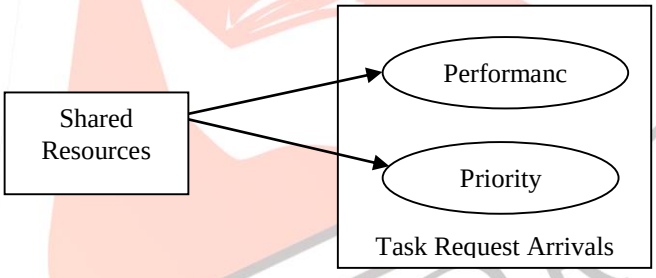

Fig. 4: Distributional Class

Load Shifting

The load of data center applications can change continuously. We only need to invoke our algorithm periodically or when the load changes cross certain thresholds. Hence, if a flash crowd requires an application to add a large number of servers, all the servers are started in parallel. Our algorithm is highly efficient and can scale to tens of thousands of servers and applications. The amount of load change during a decision interval may correspond to the arrivals or departures of several items in a row. A large load unit reduces the overhead of our algorithm because the same amount of load change can be represented by fewer items. It also increases the stability of our algorithm against small oscillation in load. On the other hand, it can lead to inefficient use of server resources and decrease the satisfaction ratio of application demands.

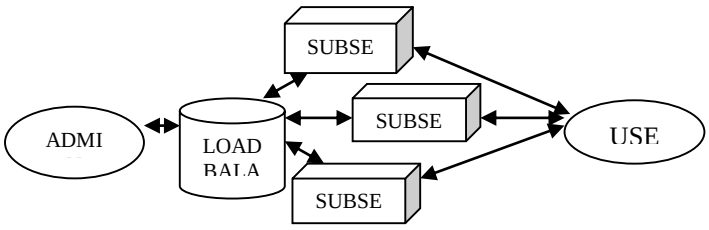

Fig. 5: Load Shifting

Auto Scaling

The space timing calculates by the reference of cloud usage. That is, the cost also calculates based on cloud space utilization and cloud usage. The server calculates which cloud doing which job. That is monitoring cloud access, cost calculation and equal sharing of jobs in cloud. We analyze and compare the performance offered by different configurations of the computing cluster, focused in the execution of loosely coupled applications. Different cluster configurations with different number of worker nodes from the three clouds Providers and different number of Jobs (depending on the cluster size), as shown in the definition of the

different cluster configurations, we use the following acronyms. We want to enable the use of largescale distributed systems for task parallel applications, which are linked into useful workflows through the looser task coupling model of passing data via files between dependent tasks and potentially larger class of task parallel Feature Extraction.

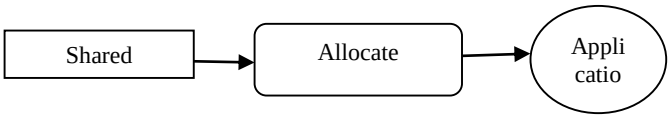

Fig. 6: Auto Scaling

Resource Utilization

There are also some cloud vendors providing auto scaling solutions for cloud users .Users are allowed to define a set of rules to control the scaling actions.

However, the rules and the load balancing strategies they used are very simple. They perform the scaling actions simply when some conditions are met and balance the load evenly across all instances. Since they do not take the state of the whole system into consideration, they cannot reach a globally optimal decision. They allocate resources on shared cluster serves periodically.

Overview of Cloud of Clouds system

Increasing resource demands with different requirements from users raise new challenges which a single provider may not be able to satisfy, given that the resilience of cloud services and the availability of data stored in the cloud are the most important issues. Scaling up the infrastructure might be a solution for each provider, but it costs a lot to do so, and the infrastructure may be underutilized when demand is low. A multiple cloud approach, which is referred to as Cloud of Clouds, is a promising solution in which several providers cooperate to build up a Cloud of Clouds system for allocating resources to users. The Cloud of Clouds system can facilitate expense reduction (i.e., savings on the operation cost), avoiding adverse business impacts and offering cooperative or portable cloud services to users. The architecture of a Cloud of Clouds system is depicted in *Fig. 1* in which a dedicated broker is responsible for coordinating the cooperation among providers. The broker has all information about the resource capacities and price policies of all providers. Based on the users' resource requests, the broker will run a cooperation decision algorithm to decide with whom a particular provider should cooperate. The broker can be cloned on each provider's infrastructure and the cooperation decision algorithm will be executed when required by its owner. However, since price policies and resource capacities of providers change over time keeping the consistency of this information for each version of the broker may not be easy. Therefore, we consider the case of a centralized algorithm run on the centralized Cloud of Clouds Broker to yield a global optimal solution. The focused cooperation problem is the agreement among providers for outsourcing users' resource requests. Although providers are competing to attract users and improve their revenues, between any two providers, an outsourcing agreement may be established such that one provider can outsource its users' resource requests to its cooperator (or helper), i.e., satisfying users' resource requests by using the cooperator's infrastructure. However, how is the outsourcing cost calculated? Since providers are rational, the cooperation should result in a win-win situation where the provider who outsources its users' resource requests may pay a lower cost than satisfying them locally, and the provider who hosts outsourcing requests will receive the final revenue at least as much as that without cooperation.

It is to be noted that under the Cloud of Clouds model, many intertwined issues need to be considered before the system can operate efficiently. First, interoperability is one of the major issues. Every provider has its own way on how users or applications interact with the cloud infrastructure, leading to cloud API propagation. This prevents the growth of the cloud ecosystem and limits cloud choice because of provider lock in, lack of portability and the inability to use the services offered by multiple providers. An interoperability standard is therefore needed to enable users' applications on the Cloud of Clouds to be interoperable. Second, cooperation among providers requires a common Service Level Agreement (SLA) governing expected quality of service, resource usage and operation cost. Defining a "good faith" SLA allows the Cloud of Clouds to minimize conflicts which may occur during the negotiation among providers. One reason for the occurrence of these conflicts is that each provider must agree with the resources contributed by other providers against a set of its own policies. Another reason is the incurrence of high cooperation costs (e.g., network establishment, information transmission, capital flow) by the providers as they do not know with whom they should cooperate. Last but not least, the network latency among providers' infrastructures also needs to be taken into account. It can be added as a constraint in the cooperation model to guarantee the service availability and satisfy the special requirements of users, and thus, may affect the optimal cooperation structure. In this paper, we only focus on the cooperation agreement among providers, and leave the study of the other issues mentioned above for future work. Hereafter, the term "hosting" provider is used to refer to the provider who satisfies its own users' requests (i.e., "stays local") and accepts outsourcing requests from other providers, and the term "outsourcing" provider is used to refer to the provider who outsources its users' resource requests to another provider and becomes idle.

Several assumptions are needed to make our model simple but still reflect the real behaviors of providers. First, we assume that a provider can outsource its users' resource requests to only one provider at one time. All its users' resource requests will be satisfied by the selected provider and its local resources become idle. Satisfying a partial number of users' resource requests at the local site or sending to multiple providers may not be the best choice since the overhead of the operation cost at the local site plus the cooperation costs at the partners' sites may increase.

Second, a provider can accept as many as outsourcing requests without facing the limitation of resource capacities. This reflects

our aforementioned assumption that a provider has enough resource capacity to satisfy all users' resource requests since supply is much higher than demand in the current cloud market. Additionally, with the quaint limited capacities, the order of accepting outsourcing requests becomes unimportant. This assumption reflects the case of Amazon EC2 which has quaint limited capacities and wants to have as many resource requests as possible to gain higher revenue.

Finally, a provider can refuse outsourcing requests if its total final revenue when hosting resource requests from others is less than that when outsourcing its own users' resource requests to another provider. In this case, outsourcing providers will choose the next highest provider if it exists. If not, they have to satisfy their users' resource requests locally.

III. CONCLUSION

We have presented the design, implementation, and evaluation of a resource management system for cloud computing services our system multiplexes virtual to physical resources adaptively based on the changing demand. We present a system that uses virtualization technology to allocate data center resources dynamically based on application demands and support green computing by optimizing the number of servers in use. We use the skewers metric to combine VMs with different resource characteristics appropriately so that the capacities of servers are well utilized. Our algorithm achieves both overload avoidance and green computing for systems with multi resource constraints. We have proposed a new strategy that can be included in the Cloud Analyst to have cost effective results and development and we can conclude from the results that this strategy is able to do so. From the work done, we can conclude that the simulation process can be improved by modifying or adding new strategies for traffic routing, load balancing etc. to make researchers and developers able to do prediction of real implementation of cloud, easily. We develop a set of heuristics that prevent overload in the system effectively while saving energy used. Trace driven simulation and experiment results demonstrate that our algorithm achieves good performance. In the cloud model is expected to make such practice unnecessary by offering automatic scale up and down in response to load variation. It also saves on electricity which contributes to a significant portion of the operational expenses in large data centers.

IV. REFERENCES

[1] H. Xu and B. Li, "Dynamic Cloud Pricing for Revenue Maximization," IEEE Trans. Cloud Computing, vol. 1, no. 2, pp. 158–171, 2013. K. E. Train, "Discrete Choice Methods with Simulation," Identity, vol. 18, no. 3, pp. 273–383, 2003.
[2] M. J. Osborne, An Introduction to Game Theory. Oxford University Press, 2004.
[3] R. Bellman, "A Markovian Decision Process," Indiana Univ. Math. J., vol. 6, no. 4, pp. 679–684, 1957.
[4] A. N. Toosi, R. K. Thulasiram, and R. Buyya, "Financial Option Market Model for Federated Cloud Environments," in UCC 2012, Chicago, Illinois, USA, Dec. 2012, pp. 3–12.
[5] A. Gera and C. H. Xia, "Learning Curves and Stochastic Models for Pricing and Provisioning Cloud Computing Services," Service Science, vol. 3, no. 1, pp. 99–109, Mar. 2011.
[6] B. Javadi, R. K. Thulasiram, and R. Buyya, "Characterizing spot price dynamics in public cloud environments," Future Generation Computer Systems, vol. 29, no. 4, pp. 988–999, Jun. 2013.
[7] B. Sharma, R. K. Thulasiram, P. Thulasiraman, S. K. Garg, and R. Buyya, "Pricing Cloud Compute Commodities: A Novel Financial Economic Model," in CC Grid 2012, Ottawa, Canada, May 2012, pp. 451–457.
[8] D. Niu, C. Feng, and B. Li, "Pricing Cloud Bandwidth Reservations under Demand Uncertainty," in SIGMETRICS'12, London, UK, June 2012, pp. 151–162.
[9] H. Xu and B. Li, "Maximizing Revenue with Dynamic Cloud Pricing: The Infinite Horizon Case," in IEEE ICC 2012, Ottawa, Canada, June 2012, pp. 2929–2933.
[10] F. Teng and F. Magoules, "Resource Pricing and Equilibrium Allocation Policy in Cloud Computing," in CIT 2010, Bradford, UK, June 2010, pp. 195–202.
[11] M. Mihailescu and Y. M. Teo, "On Economic and Computational Efficient Resource Pricing in Large Distributed Systems," in CCGrid 2010, Melbourne, Australia, May 2010, pp. 838–843.
[12] G. Allon and I. Gurvich, "Pricing and Dimensioning Competing Large Scale Service Providers," Manufacturing Service Operations Management, vol. 12, no. 3, pp. 449–469, 2010.
[13] D. Bergemann and J. Valim¨aki," "Dynamic price competition," Journal of Economic Theory, vol. 127, no. 1, pp. 232–263, 2006.
[14] K. Y. Lin and S. Y. Sibdari, "Dynamic price competition with discrete customer choices," European Journal Of Operational Research, vol. 197, no. 3, pp. 969–980, 2009.
[15] S. U. Khan and I. Ahmad, "A Cooperative Game Theoretical Technique for Joint Optimization of Energy Consumption and Response Time in Computational Grids," IEEE Transactions on Parallel and Distributed Systems, vol. 20, no. 3, pp. 346–360, 2009.
[16] S. Penmatsa and A. T. Chronopoulos, "Price based User optimal Job Allocation Scheme for Grid Systems," in IEEE IPDPS 2006, Rhodes Island, Greece, April 2006, pp. 336–343.
[17] Z. Kong, B. Tuffin, Y.K. Kwok, and J. Wang, "Analysis of Duopoly Price Competition Between WLAN Providers," in IEEE ICC 2009, Dresden, Germany, June 2009, pp. 1–5.
[18] D. Niyato, A. V. Vasilakos, and Z. Kun, "Resource and Revenue Sharing with Coalition Formation of Cloud Providers: Game Theoretic Approach," in CCGrid 2011, Newport Beach, USA, May 2011, pp. 215–224.
[19] B. Boghosian, P. Coveney, S. Dong, L. Finn, S. Jha, G. Karniadakis, and N. Karonis, "NEKTAR, SPICE and Vortonics: using federated grids for large scale scientific applications," Cluster Computing, vol. 10, no. 3, pp. 351–364, 2007.
[20] M. Sobolewski and R. M. Kolonay, "Federated grid computing with interactive serviceoriented programing," Concurrent Engineering, vol. 14, no. 1, pp. 55–66, 2006.
[22] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. M. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres,

M. BenYehuda, W. Emmerich, and F. Galan, "The RESERVOIR Model and Architecture for Open Federated Cloud Computing," IBM J Res Dev, vol. 53, no. 4, pp. 535–545, 2009.
[23] I. Goiri, J. Guitart, and J. Torres, "Economic model of a Cloud provider operating in a federated Cloud," Information Systems Frontiers, vol. 14, no. 4, pp. 827–843, Sept. 2011.
[24] M. M. Hassan, M. S. Houssan, A. M. J. Sarkar, and E.n. Huh, "Cooperative gamebased distributed resource allocation in horizontal dynamic cloud federation platform," Information Systems Frontiers, pp. 1–20, June 2012.
[25] P. Dubey, "Inefficiency of Nash Equilibria," Mathematics of Operations Research, vol. 11, no. 1, pp. 1–8, 1986.
[26] G. Mart´ınHerran´ and J. Rincon´Zapatero, "Efficient Markov perfect Nash equilibria: theory and application to dynamic fishery games," J. Econ. Dynam. Control, vol. 29, no. 6, 2005.
[27] E. J. Gumbel, "Multivariate extremal distributions," Bull. Inst. Internat. Statist., vol. 39, no. 2, pp. 471–475, 1962.
[28] U. Doraszelski and K. L. Judd, "Avoiding the curse of dimensionality in dynamic stochastic games," Quantitative Economics, vol. 3, no. 1, pp. 53–93, 2012.
[29] U. Doraszelski and A. Pakes, "A Framework for Applied Dynamic Analysis in IO," in Handbook of Industrial Organization, ser. Handbooks in Economics, M. Armstrong and R. Porter, Eds. Elsevier, 2007, ch. 30, pp. 1887–1966.
[30] K. L. Judd, Numerical Methods in Economics. The MIT Press, 1998.