Profit-Aware Policy Scheduler (PAPS) for Resource Allocation in IaaS Clouds

Rajalakshmi Shenbaga Moorthy, Thamarai Selvi Somasundaram, Kannan Govindarajan

Abstract — **Infrastructure as a Service (IaaS) is a type of Cloud Computing service delivery model that provides compute, storage, and network resources to the consumers in an on demand manner. In IaaS cloud environment, resource allocation is one of the complex tasks due to the heterogeneous nature of cloud resources and dynamic job requirements to run the jobs. However, the IaaS cloud resource allocation mechanism should consider both the customer's profit as well as provider's profit, while allocating resources to the jobs. Henceforward, in this research work we proposed a Profit-Aware Policy Scheduler (PAPS) incorporated with two scheduling policies namely Provider Profit-Aware Scheduling Policy (P-PASP) and User Profit-Aware Scheduling Policy (U-PASP). Finally, we have integrated the PAPS with Cloud Scheduler to efficiently handle the user's job requests and allocate the Cloud resources to the job requests in an effective way. It is simulated to prove the effectiveness of the proposed research work by calculating the user's profit, provider's profit, and customer's satisfaction.**

Index Terms — Cloud Computing, Infrastructure as a Service, Resource Allocation, Profit Maximization, Customer Satisfaction.

I. INTRODUCTION

Cloud Computing is the recently emerging technique from the family of the distributed computing paradigm. The cloud service models in the cloud computing is categorized into Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) [1]. Nevertheless, IaaS plays a major role in providing compute, storage and network instances to the customers in an on demand manner based on their application requirements. However, the allocation of resources to the job requests is the biggest challenging issue. The resource allocation mechanism should complete the job requests within the deadline and budget specified by the user as well as it should increase the revenue of the cloud providers. This leads to increase the customers profit and satisfaction as well as providers profit. However, some of the existing research works [2], [3] and [4] aims to maximize the either the provider's profit or to maximize the customer's profit this results in user satisfaction as well as it reduces the revenue of the cloud providers. Hence, it is essential to develop a resource allocation mechanism that should consider both the customer's profit and provider's profit.

Hence, in this research work we proposed a Profit-Aware Policy Scheduler (PAPS), which is integrated with two scheduling policies namely (i) Provider Profit-Aware Scheduling Policy (P-PASP) and (ii) User Profit-Aware Scheduling Policy (U-PASP). P-PASP mainly aims to maximize the provider's profit as an objective, whereas U-PASP mainly aimed to maximize the customer's satisfaction by satisfying the user specified budget and deadline. The proposed resource scheduling policies improve the efficiency of the Cloud Scheduling mechanism by efficiently allocate the job requests to the cloud resources in an effective manner. In brief, the contribution of the research work is summarized below:

1. Designed a Profit-Aware Policy Scheduler (PAPS) to effectively allocate the cloud resources for the job requests
2. Developed a Provider Profit-Aware Scheduling Policy (P-PASP) and User Profit-Aware Scheduling Policy (U-PASP) to maximize the customers and providers profit.
3. Integrate the PAPS Scheduler with Cloud Scheduler, which is integrated with P-PASP and U-PASP to improve the efficiency of the Cloud Scheduling mechanism.
4. Simulation is carried out to prove the effectiveness of the proposed work.

The rest of the paper is organized as follows: Section 1 introduces the proposed work. Section 2 describes the related works, which are closely related to our research work. Section 3 discusses the problem statement and proposed scheduling policies in detail. Section 4 elaborates the system architecture to implement the proposed research work. Section 5 discusses the simulation setup and achieved simulation results for the proposed work. Section 6 concludes the proposed research work and explores the feasibility of future work.

II. RELATED WORKS

Chaisiri et al. [6] proposed the resource-provisioning algorithm based on the stochastic programming model to minimize the cost. The proposed algorithm allocates the cloud resources by reserving the less cost resources for reserving instances and more cost for on-demand

Manuscript received April, 2014.
Rajalakshmi Shenbaga Moorthy*, Department of Computer Science and Engineering, St. Joseph's Institute of Technology. Chennai, India.*
Thamarai Selvi Somasundaram, *Department of Computer Technology, Madras Institute of Technology, Chennai, India.,*
Kannan Govindarajan*, Department of Computer Technology, Madras Institute of Technology, Chennai, India.*

instances. Hong Xu and Baochun Li [7] claimed that there is always a trade-off between the available cloud resources and demand during the pricing of cloud resources. They have formulated the revenue maximization problem based on the stochastic program to maximize the revenue of the cloud providers. In addition, the efficiency of the proposed work validated based on their achieved numerical results. Thamarai Selvi Somasundaram and Kannan Govindarajan [8] proposed CLOUD Resource Broker (CLOUDRB) architecture for effectively managing the cloud resources and completing the scientific applications within the user-specified deadline. Their proposed solves the trade-off issue between the execution time and cost. Kuan Lu et al. [9] proposed a QoS-based resource management framework for solving the trade-off issues between customer QoS application requirements and resource provider's objectives. Hadi Goudarz and Massoud Pedram proposed a resource allocation mechanism in the distributed cloud systems to increase the total profit by meeting the Service Level Agreements. Their proposed mechanism considers processing, storage, and network resources in the resource allocation process.

III. PROBLEM STATEMENT AND SCHEDULING POLICIES

A. Problem Statement

The resource allocation problem in IaaS Cloud resources is stated as "User job requests should be completed within the specified budget and time and it should not decrease the revenue of the cloud providers". The outcome of the proposed resource allocation mechanism should increase the customers satisfaction and profit of the cloud provider's. The objective function is formulated as

$$\text{Max} \sum_{j \in M} \text{Profit}_{r_j}$$

Subject to the constraints

$$\text{Min} \sum_{\substack{i \in N \\ j \in M}} \text{Expenses}_{t_i r_j}$$

$$\text{ActualExecutionTime}_{t_i r_j} \leq \text{Deadline}_{t_i}$$

$$1 \leq i \leq N; 1 \leq j \leq M$$

B. Profit-Aware Scheduling Policies

The Algorithm for User Profit Aware Scheduling Policy is given in Algorithm 1. It first finds out the suitable resources for each task. It computes the difference between the expected execution time, which is based on the difference between finish time and start time of the task on the suitable resources. Next, it computes the expense for the task, which is based on the actual running time, and the cost incurred per hour for the resources. If the actual execution time of the task is less than the deadline specified by the user and also expense is less than the actual budget proposed then satisfaction will be computed from execution time and cost. The resource, which brings maximum satisfaction to the task, will be assigned to process the task.

Algorithm 1: User Profit Aware scheduling Policy

Input: $T \leftarrow \{t_1, t_2, \ldots t_N\}$, $R \leftarrow \{r_1, r_2, \ldots r_M\}$

Output: SR_{t_i} $1 \leq i \leq N$

For $(i=1; i \leq N; i{+}{+})$

{

$SR_{t_i} = \{\ \}$

$\text{Satisfaction}_{t_i} = 0$

For $(j=1; j \leq M; j{+}{+})$

{

Compute the Expected Execution Time of the task as

$\text{ExpectedExecutionTime}_{t_i r_j} = \text{FinishTime}_{t_i} - \text{StartTime}_{t_i}$

Compute the Expenses as

$\text{Expenses}_{t_i r_j} = \text{Cost}_{t_i r_j} * \text{ActualExecutionTime}_{t_i r_j}$

If $\left(\begin{array}{l}\text{ActualExecutionTime}_{t_i r_j} \leq \text{Deadline}_{t_i} \\ \&\&\ \text{Expenses}_{t_i r_j} \leq \text{Budget}_{t_i}\end{array}\right)$

{

Compute the Saved Time as

$\text{SavedTime}_{t_i} = \text{ExpectedExecutionTime}_{t_i r_j} - \text{ActualExecutionTime}_{t_i r_j}$

Compute the Saved Cost as

$\text{SavedCost}_{t_i} = \text{SavedTime}_{t_i} * \text{Cost}_{t_i r_j}$

Compute the Satisfaction as

$\text{MaxSatisfaction}_{t_i r_j} = \alpha * \text{SavedTime}_{t_i} + \beta * \text{SavedCost}_{t_i}$

Include the resource r_j in the set SR_{t_i}

If $\left(\text{Satisfaction}_{t_i} < \text{MaxSatisfaction}_{t_i r_j}\right)$

{

$\text{MaxSatisfaction}_{t_i r_j} = \text{Satisfaction}_{t_i}$

Mark the resource r_j

}

}

}

Assign the resource r_j for the task t_i

}

The algorithm for Provider Profit Aware Scheduling Policy is given in Algorithm 2. It computes the revenue and profit for each resource that satisfies the user task's QoS requirements. The resource that earns highest profit will be assigned to the resource.

Algorithm 2: Provider Profit Aware Scheduling Policy

Input: $T \leftarrow \{t_1, t_2, \ldots t_N\}$, SR_{t_i}

Output: A resource r_j for t_i

For $(i=1; i \leq N; i{+}{+})$

{

For each resource in SR_{t_i}

{

Compute the revenue for the provider as

$Revenue_{t_i r_j} = Cost_{t_i r_j} * ActualExecutionTime_{t_i r_j}$

Compute the profit for the provider

$Profit_{t_i r_j} = Revenue_{t_i r_j} - Cost_{t_i r_j}$

If $\left(Profit_{t_i} < Profit_{t_i r_j} \right)$

{

$Profit_{t_i} = Profit_{t_i r_j}$

Mark the resource r_j

}

}

Assign the resource r_j for the task t_i

}

IV. PROPOSED ARCHITECTURE

The system architecture to implement our proposed research work is shown in Figure 1.

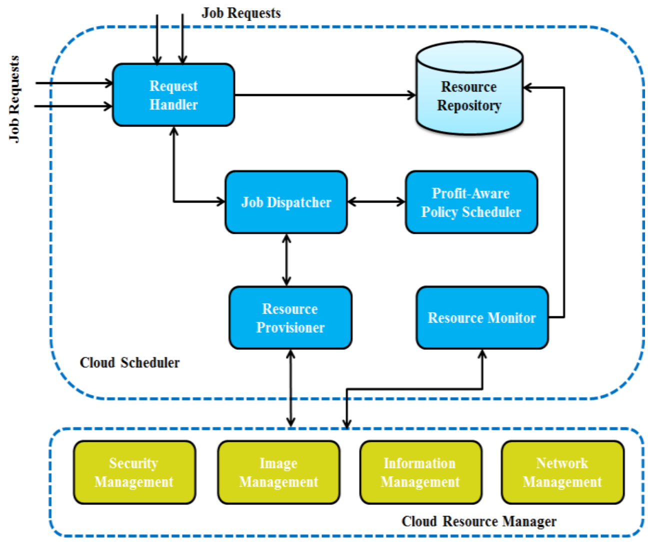

Figure 1: Proposed Architecture

The Request Handler component handles the user application requirements, which consist of software, hardware, budget, deadline, etc. It matches the user hardware and software requirements with available Cloud resources information, and it sends the matched resource list to the Job Dispatcher. It is the component, which first sends the job requests to Profit-Aware Policy Scheduler (PAPS) to allocate the resources for the job requests. It is integrated with two scheduling policies namely (i) User Profit-Aware Scheduling Policy (UPASP) (ii) Provider Profit-Aware Scheduling Policy (PPASP). The scheduling policies, selects the resources based on the user and provider specified constraints, then, it sends the selected resources to the Job Dispatcher. Job Dispatcher dispatches the job requests to the Resource Provisioner component. The Resource Provisioner interfaces the Cloud Resource Manager to provision the virtual machine instances for running the user applications. The Cloud Resource Manager has four major components, namely (i) Security Management (ii) Image Management (iii) Information Management and (iv) Network Management. The security management, handling the authentication and authorization related issues between the users and the cloud providers. The image management is responsible for managing the operating system images in the cloud providers. The information management handles the physical and virtual resource related information. The network managements handle the IP related and connectivity issues between the user's and the virtual machines resides in the cloud providers.

V. SIMULATION SETUP AND RESULTS

The simulation setup is discussed in section 5.1 and simulation results are discussed in section 5.2.

A. Simulation Setup Details

We have simulated the proposed PAPS mechanism in this paper which considers both U-PASP and P-PASP. The simulation results are compared with PAPS considers either User Profit-Aware Scheduling Policy (U-PASP) or Provider Profit-Aware Scheduling Policy (P-PASP). The simulation results are evident that our proposed approach achieves better results compared to other mechanism. The experiments are carried out by generating the job requests from 1 to 500. Similarly, we have generated the cloud resources from 1 to 1000.

B. Simulation Results

The simulation experiments are carried out to calculate the user's profit, provider's profit and customer's satisfaction. The first simulation experiment is carried out to analyze the customers' profit. Our proposed PAPS approach is compared with other techniques of U-PASP and P-PASP. The customers profit results are represented as graphical charts which are shown in Figure 2. From the results as shown in Figure 2, it is evident that our proposed scheduling approach yields better customers profit.

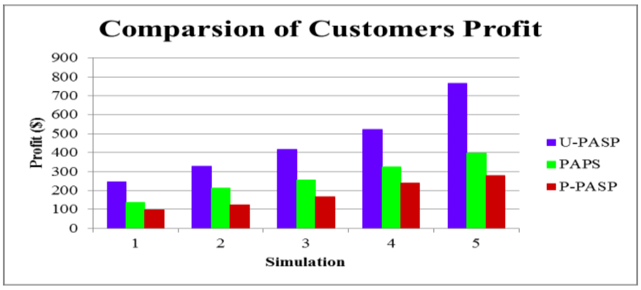

Figure 2: Comparison of Customers Profit

The second simulation experiment is carried out to analyze the provider's profit. Our proposed PAPS approach is compared with other techniques of U-PASP and P-PASP. The results are represented as graphical charts which are shown in Figure 3. From the results as shown in Figure 3, it is evident that our proposed scheduling approach yields better provider's profit, which travels between P-PASP and U-PASP. The customers profit is very high in UPASP, since it is greedy for profit of customers alone and does not take into account of resources, which decreases the profit for the resource provider. The proposed PAPS achieve a balanced customer and provider profit because the proposed algorithm first finds the resources that can process the task in minimal time and also with minimal expenses. However, the customers profit in PPASP is very minimal, since their main motivation is to maximize the profit of the resources, they do not consider the profit of the customers, which leads to increase more expenses for the customers.

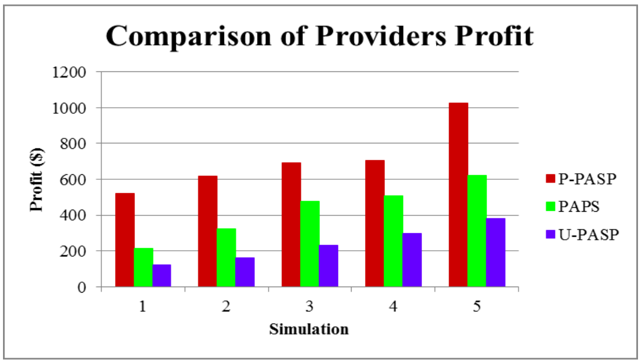

Figure 3: Comparison of Providers Profit

This simulation experiment is carried out to analyze the customer's satisfaction. Our proposed PAPS approach is compared with popular techniques of Round Robin. The results are represented as graphical charts which are shown in Figure 4. From the results as shown in Figure 4, it is evident that our proposed scheduling approach maximizes the customer's satisfaction. The proposed PAPS achieve maximum customer satisfaction than Round Robin approach. The reason behinds that, PAPS allocate the resource to the task by considering deadline and the budget specified for the task. The round robin allocates the jobs without considering deadline and budget, i.e. allocates the resource which is available at that time. It leads to the non-completion of task within the deadline that increases the execution time of the task; obviously it maximizes the cost that reduces the customer's satisfaction.

Figure 4: Comparison of Customer's Satisfaction

VI. Conclusion and Future Work

In this paper, we have proposed Profit-Aware Policy Scheduler (PAPS) architecture to allocate the job requests resources in an efficient manner from the IaaS Cloud resources. We have proposed two scheduling policies namely Provider Profit Aware Scheduling Policy (P-PASP) and User Profit Aware Scheduling Policy (U-PASP) and it is integrated with Cloud Scheduler. The integrated scheduling policies in the Cloud Scheduler enhance the efficiency Cloud Scheduling mechanism. We have also illustrated that the proposed research work effectively schedules the user job requests and allocates the job requests to the cloud resources in such a way that the proposed mechanism maximizes the customer's profit and provider's profit as well as the customer's satisfaction. The future work will be the integration of the resource prediction mechanism to further improve the resource allocation process.

REFERENCES

[1] NIST, National Institute of Standards and Technology (2011), http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloudddefinition.pdf.

[2] Xi Chen; Haopeng Chen; Qing Zheng; Wenting Wang; Guodong Liu, "Characterizing web application performance for maximizing service providers' profits in clouds," Cloud and Service Computing (CSC), 2011 International Conference on, vol., no., pp.191,198, 12-14 Dec. 2011. doi: 10.1109/CSC.2011.6138519.

[3] Lampe, U.; Siebenhaar, M.; Papageorgiou, A.; Schuller, D.; Steinmetz, R., "Maximizing Cloud Provider Profit from Equilibrium Price Auctions," Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on, vol., no., pp.83,90, 24-29 June 2012, doi: 10.1109/CLOUD.2012.19.

[4] J. Chen, C. Wang, B.B. Zhou, L. Sun, Y.C. Lee, A.Y. Zomaya, "Tradeoffs between profit and customer satisfaction for service provisioning in the cloud," 20th ACM HPDC, San Jose, June 8-11, 2011.

[5] Chaisiri, S., Lee, B.S., Niyato, D.: Optimization of resource provisioning cost in cloud computing. IEEE Transactions on Services Computing 5(2), 164{177 (2012). DOI:10.1109/TSC.2011.

[6] Hong Xu, Baochun Li, "Dynamic Cloud Pricing for Revenue Maximization," IEEE Transactions on Cloud Computing, vol. 1, no. 2, pp. 158-171, July-Dec. 2013, doi:10.1109/TCC.2013.15.

[7] Thamarai Selvi Somasundaram, Kannan Govindarajan, "CLOUDRB: A framework for scheduling and managing High-Performance Computing (HPC) applications in science cloud", Future Generation Computer Systems, Volume 34, Pages 47-65, 2014.

[8] Kuan Lu, Ramin Yahyapour, Philipp Wieder, Constantinos Kotsokalis, Edwin Yaqub, Ali Imran Jehangiri, "QOS-BASED RESOURCE ALLOCATION FRAMEWORK FOR MULTIDOMAIN SLA MANAGEMENT IN CLOUDS", International Journal of Cloud Computing (ISSN 2326-7550) Vol. 1, No. 1, January-March 2013.

[9] Goudarzi, H.; Pedram, M., "Maximizing Profit in Cloud Computing System via Resource Allocation," Distributed Computing Systems Workshops (ICDCSW), 2011 31st International Conference on, vol., no., pp.1,6, 20-24 June 2011, doi: 10.1109/ICDCSW.2011.52

Rajalakshmi Shenbaga Moorthy received the B.Tech under the stream of Information Technology from Mookambigai College of Engineering in 2010. She completed her Masters in Engineering (M.E) in Computer Science from Madras Institute of Technology, Anna University in 2013. She is the Gold Medalist in her B.Tech and M.E programme. She is currently working as an Assistant Professor in St. Joseph's Institute of Technology, Chennai. Her research area interest includes Cloud Computing, System Software and Analysis of Algorithms. She has published three International conference papers, which is indexed in IEEE explorer and Elsevier publications.

Thamarai Selvi Somasundaram is working as Professor & Dean in MIT Campus, Anna University, Chennai, India. She received her Ph.D. in Computer Science and Engineering in Manonmaniam Sundaranar University, Tirunelveli. She has 28 years of teaching and research experience. Her area of interest includes Artificial Intelligence, Neural Networks, Grid Computing, Grid Scheduling, Semantic Technology, Virtualization and Cloud Computing. She has authored/co-authored five books. She has received the best IBM faculty award in the year of 2009. She received Active Researcher Award in 2011 by Anna University, Chennai. She has authored/co-authored more than 100 research publications to her credit. She established Grid Computing Research Laboratory in the Department of Information Technology at MIT Campus, Anna University. She is the reviewer for Future Generation Computing Systems (FGCS), Journal of Grid Computing, Journal of Parallel and Distributed Computing, Transactions on Mobile Computing and IEEE Transactions on Data and Knowledge Engineering. She has published nearly 50 reputed journal papers and more than 100 National/International conference papers.

Kannan Govindarajan received the B.E. degree in Information Technology from Bharathidasan University. He completed his M.S (By Research) degree in Computer Science and Engineering from Anna University, Chennai in the area of Grid Computing. He is currently doing his Ph.D in the area of Cloud Computing in Computer Science and Engineering at Anna University. His area of interest includes Grid Computing, Grid Scheduling, Semantic Technology, Virtualization and Cloud Computing. He has been invited to visit Athabasca University, Canada for research interaction and development under the Pro Development Initiative Grant (PDIG) funded by Indo-Canadian Shastri Institute. Furthermore, he has received the BGI Scholarship in 2012 to visit Kuwait University. He is one of the inventors for patent received and also he is one of the inventors for the two patents which have been filed and it is under consideration. He has published four reputed journal papers and more than twenty National/International conferences. He is the reviewer for Journal of Super Computing, Journal of Parallel and Distributed Computing and International Journal of Electrical Power and Energy.

