SEOUL NATIONAL UNIVERSITY
College of Engineering
Technology Management, Economics, and Policy Program

Technology Management, Economics, and Policy Discussion Paper Series

Enabling Business-Preference-Based Scheduling of Cloud Computing Resources

Azamat Uzbekov, Jörn Altmann

TEMEP Discussion Paper No. 2016:134

서 울 대 학 교

공 과 대 학, 기술경영경제정책 대학원과정
08826 서울시 관악구 관악로 1

Technology Management, Economics, and Policy Program
College of Engineering, Seoul National University
599 Gwanak-Ro, Gwanak-Gu, Seoul 151-742, South-Korea
Phone: ++82-2-880-9140, Fax: ++82-2-880-8389

The *Technology Management, Economics, and Policy Program (TEMEP)* is a graduate program at Seoul National University. It includes both M.Sc. and Ph.D. programs, which are integrated into the graduate programs at the College of Engineering.

The *TEMEP Discussion Paper Series* is intended to serve as an outlet for publishing research about theoretical, methodological, and applied aspects of industrial economics, especially those related to the institute's areas of specialization, namely management of technology, information and communication technology, telecommunication, services, health industries, energy industries, as well as infrastructures for development. In particular, paper submissions are welcome, which analyze current technology and industry related issues, discuss their implications, and suggest possible alternative policies. The objective is to gain insights into important policy issues and to acquire a balanced viewpoint of policy making, technology management, and economics. It will enable us to identify the problems in industries accurately and to come up with optimal and effective guidelines. In addition, another important aim of this series is to facilitate communication with external research institutes, individual researchers, and policy makers worldwide.

Research results disseminated by TEMEP may include views on policies for information and communication technologies, technology management, telecommunication, energy, health, and development economics, but the institute itself takes no institutional policy position. If appropriate, the policy views of the institute are disseminated in separate policy briefs. Thus, any opinion expressed in the TEMEP Discussion Paper Series is those of the author(s) and not necessarily the opinion of the institute.

Finally, the current editor of this series would like to thank Prof. Dr. Almas Heshmati for establishing this working paper series in January 2009 and for guiding its development during the course of the first 55 issues. In particular, Prof. Heshmati provided invaluable contributions for setting up the local IT infrastructure, registering the series with RePEc, disseminating the working papers, and setting the quality requirements.

Jörn Altmann, Editor

Office: 37-305
Technology Management, Economics, and Policy Program
College of Engineering
Seoul National University
1 Gwanak-Ro, Gwanak-Gu
Seoul 08826
South-Korea

Phone: +82-70-7678-6676
Fax: +1-501-641-5384
E-mail: jorn.altmann@acm.org

Enabling Business-Preference-Based Scheduling of Cloud Computing Resources

Azamat Uzbekov, Jörn Altmann

Department of Industrial Engineering
College of Engineering
Seoul National University
Seoul, South-Korea
batukasss@gmail.com, jorn.altmann@acm.org

April 2017

Abstract. Although cloud computing technology gets increasingly sophisticated, a resource allocation method still has to be proposed that allows providers to take into consideration the preferences of their customers. The existing engineering-based and economics-based resource allocation methods do not take into account jointly the different objectives that engineers and marketing employees of a cloud provider company follow. This article addresses this issue by presenting the system architecture and, in particular, the business-preference-based scheduling algorithm that integrates the engineering aspects of resource allocation with the economics aspects of resource allocation. To show the workings of the new business-preference-based scheduling algorithm, which integrates a yield management method and a priority-based scheduling method, a simulation has been performed. The results obtained are compared with results from the First-Come-First-Serve scheduling algorithm. The comparison shows that the proposed scheduling algorithm achieves higher revenue than the engineering-based scheduling algorithm.

Keywords: Cloud Computing, Resource Allocation, FCFS, Yield Management, Scheduling, Pricing, Economics-Based Resource Allocation, System Architecture.

JEL Classification Numbers: C61, C63, D24, D81, L86, M15.

1. Introduction

Cloud computing, which has become the infrastructure for ICT services, is even on its way to become a household utility, for which the processes of using cloud services and making payments are similar to other household utilities (e.g., water, electricity) [1] [2]. As for any household utilities, cost is most important. This fact makes an efficient resource allocation vital for cloud service providers (CSP).

Although many aspects of resource allocation have already been discussed in the literature [3], resource allocation for cloud computing still allows new possibilities [1] [4] [5] [6], such as cloud service pricing strategies and scheduling algorithms [7] [8].

In detail, looking at existing research on resource allocation, a partition into two types of methods can be observed: (a) engineering-based resource allocation methods, which consider utilization, response time, and throughput for allocating CPU, memory, and storage [9] [10] [11]; and (b) economics-based resource allocation methods, which consider profit, revenue, and cost for allocating resources [12] [13] [14]. Up to now, these two types of methods are not combined or integrated yet. This is the case despite the fact that CSPs aim at maximizing their profit, which can be achieved by considering the market demand and the cost of the engineering system used. Therefore, CSPs need to consider simultaneously the engineering aspects and the economics aspects of resource allocation [15].

Therefore, the objective of this article is to outline a system architecture that allows a CSP to integrate economics aspects of resource allocation with engineering aspects of resource allocation. The research questions, which can be derived from this objective, are: How does the system architecture look that can integrate economics aspects of resource allocation with engineering aspects of resource allocation? How does an integrating resource allocation method operate, combining an economics-based resource allocation method and an engineering-based resource allocation method? What is the performance of this integrated scheduling algorithm (i.e., business-preference-based scheduling algorithm)?

To answer these research questions, we conduct the following steps: First, based on a solid literature research on system architectures and resource allocation methods, we propose a cloud computing resource allocation architecture that integrates economics-based resource allocation and engineering-based resource allocation. Second, using this architecture, the integrating scheduling algorithm, which is called business-preference-based scheduling, is designed. Third, for showing the workings of the proposed architecture, one scenario with two demand-depending cases are simulated and analyzed. The first case represents normal demand for computing resources and the second case represents high demand for computing resources. The simulation results show that the performance of the business-preference-based scheduling algorithm is better than a FCFS scheduling algorithm with respect to the CSP revenue generated.

The contribution of this article is a new resource allocation architecture, which enables the integration of economics-based resource allocation and engineering-based resource allocation. This integration of these two types of resource allocation methods is missing in existing system architectures [15]. The core of the architecture is the business-preference-based scheduling algorithm, which allows expressing CSP business strategy parameters by adjusting the ready queue of the tasks. It combines yield management and priority-based scheduling.

The remainder of the paper is organized as follows: In the next section, an overview of related work on resource allocation is given. Section 3 introduces the proposed system architecture with a focus on the business-preference-based scheduling algorithm. The application of the architecture is evaluated through simulations in Section 4. Section 5 concludes the paper with a discussion.

2. Background

2.1. Cloud Computing Resource Allocation

Resource allocation methods can be divided into two types: economics-based methods and engineering-based methods [15]. To determine the intersection of both resource allocation methods (Figure 1), relevant articles have been reviewed in the following paragraphs.

Engineering-based methods allocate resources based on task parameters (e.g., time of arrival, length of task, importance of task) and system objectives (e.g., maximization of throughput). The classical algorithms of this type are: First-Come-First-Serve (FCFS), Shortest-Job-First (SJF), Priority Scheduling (PS), and Round-Robin (RR) [16] [17]. In the last decade, each of these methods has been modified and enhanced. For example, RR was improved to use the optimal time quantum [9] [18] [19]. PS and SJF Scheduling have been adapted for cloud-based software systems [20]. CloudSim-based simulations of a generalized priority scheduler, which sets priorities not only for tasks but also VMs, showed good results with respect to execution times [21]. These algorithms, which focus only on engineering aspects, require technical skills but do not require an understanding of the interaction of the provider with the customer [22].

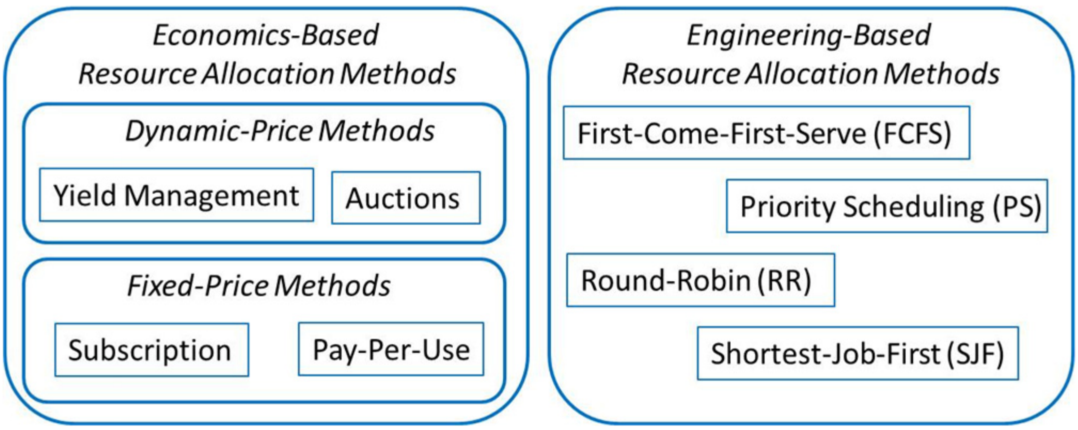

Figure 1. Classification of resource allocation methods into engineering-based methods and economics-based methods.

Economics-based resource allocation methods distribute cloud resources to users according to their preferences by using pricing [16] [17] [18] [19] [23]. These methods can further be subdivided into fixed-price methods and dynamic-price methods. The fixed-price methods are implemented by major CSPs [24] [25] [26]. An example of the fixed-price method is the pay-per-use method, which assigns a fixed price to each resource. In [27], the authors consider the pay-per-use method, to propose a set of policies that allocate VMs according to the QoS purchased by the user. The article discusses cloud resources as a virtual pool of the physical infrastructure without giving detailed resource specifications. Another example of fixed-price methods is the subscription method, under which any number of cloud resources can be used for a fixed price for a certain period of time. With respect to the dynamic-price methods, Al-Roomi et al. used dynamic pricing to allow CSPs or users to change the price depending on pertinent factors [26]. Auctions and yield management are dynamic-price methods [28]. An example of a sealed-bid uniform price auction is Amazon EC2's spot instance. However, these economics-based methods do not consider any engineering aspects.

2.2. Yield Management

Yield management, as an example of a dynamic-price method, takes into account demand, prices, and resource availability [29]. It implements basic principles of supply and demand economics in a way to generate incremental revenues [15]. Yield management, which has been applied in the airline industry, helps to sell resources to the consumer at a specific time at the highest possible price [30]. The possibility of using yield management in computing grids has been studied in detail [30] [31] [32], outlining the requirements for applying yield management to computing grids and showed how the tools based on yield management could be executed. In our previous work [15], yield management has been applied as one of the resource allocation algorithms within a CSP business support framework. However, all yield management applications did not specify the engineering details for mapping yield management allocations efficiently onto ICT resources.

2.3. Demand Estimation

An issue in resource allocation is demand forecasting. Studies of the Internet and media workloads indicate that customer demand is highly variable (i.e., the peak-to-mean ratio is an order of magnitude or more), and it is not economical to overprovision the system using peak demands [33] [34]. Gmach, who has illustrated the peak-to-mean behavior for 139 enterprise application workloads [35], showed that an understanding of enterprise workloads burstiness could help choosing the right tradeoff between quality of service and the resource pool capacity requirements. The ability to plan and operate in the most cost-efficient way is a critical competitive advantage [35]. For this article, we consider historical data to estimate the demand.

3. Techno-Economic System Architecture

3.1. Proposed System Architecture

The system architecture (Figure 2), which we propose for integrating an engineering-based resource allocation method and an economic-based resource allocation method (Figure 1), comprises the following two stakeholders and three modules:

User: The user (customer), who is one of the two stakeholders, negotiates with the cloud service provider about the service level agreement (SLA) on cloud resources needed to execute an application.

Cloud Service Provider (CSP): This stakeholder interacts with the user (customer), in order to negotiate a service level agreement (SLA) that meets the users' needs and the provider's economic objective (e.g., profit maximization, revenue maximization, or social welfare maximization). The SLA determines the resources (e.g., VM) that the user can access and the pricing plan associated with the resources.

Pricing Module: The pricing module allows specifying and applying different pricing plans that the CSP might want to offer in the market to attract customers. Yield-management-based pricing plan is an example of such a pricing plan.

Resource Allocation Integration Module: This module admits tasks and sets the order of tasks (i.e., VMs) in the ready queue, using information about VM prices paid by users and about the engineering-based resource allocation method used by the scheduler.

Scheduler: With the help of this module, tasks from the ready queue are allocated to the hardware (e.g., CPU). The scheduler can be implemented with any engineering-based resource allocation method. The ready queue is an ordered list of tasks (i.e., VMs) that are ready to be executed through the hardware.

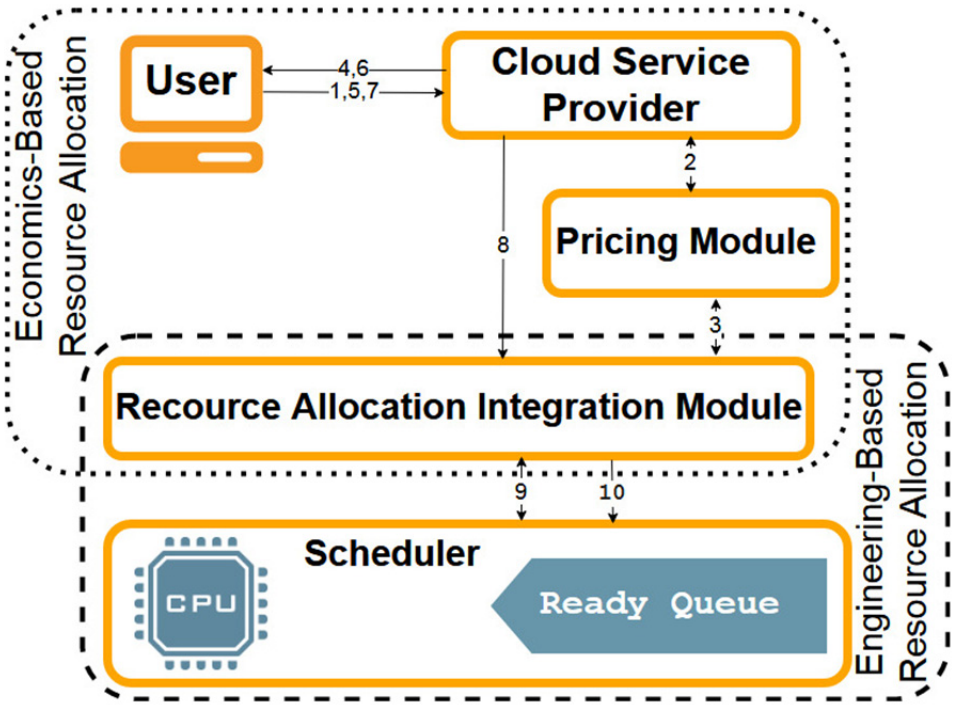

Figure 2. Proposed techno-economic system architecture for integrating engineering-based and economics-based resource allocation methods.

3.2. Resource Allocation Process between User and Cloud Service Provider

The resource allocation process from a user's first request about cloud resource availability to the execution of the VM on a hardware within the proposed system architecture can be described in ten steps (Figure 2): (1) The user sends a request for VM prices to the CSP; (2) The CSP calculates the VM price with the pricing module, which also considers the status of the ready queue. The pricing module can run different economics-based resource allocation methods, depending on CSP's business strategy; (3) The ready queue status is obtained from the resource allocation integration module; (4) The CSP communicates the VM price (as part of a SLA) to the user; (5) If the user agrees to the price (SLA), the user makes a purchase request to the CSP; (6) If the CSP acknowledges the request, the SLA is established [36] [37]; (7) The user submits a task (VM) to the CSP; (8) The CSP informs the resource allocation integration module about the established SLA, the user profile information (e.g., customer status, demand history), and the task; (9) The information from the scheduler about the engineering-based resource allocation method used and the status of the queue is requested from the scheduler by the resource allocation integration module. (10) Based on the information from the scheduler and the CSP, the resource allocation integration module calculates the position of the task in the ready queue. Then, it enlists the VM in the ready queue. Continuously, the scheduler picks the VM from the top of the queue and allocates the VM to the hardware.

3.3. Comparison with Existing System Architectures

The novelty of the proposed system architecture is the resource allocation integration module, which is situated between the engineering-based resource allocation and the economics-based resource allocation of the CSP. This module takes into consideration business aspects of the CSP (e.g., pricing and user profile) as well as the engineering requirements (e.g., engineering-based resource allocation) coming from managing the cloud infrastructure. It also forwards information about the ready queue to the pricing module, such that the pricing module can calculate the optimal pricing plan.

To show the novelty of the proposed system architecture, it is compared with four existing system architectures that have been identified in the literature. All system architectures have been classified according to four criteria, namely type of system architecture, resource allocation method used, stakeholder interactions, and objective. The results of the classification are shown in Table 1.

Table 1 depicts that the resource allocation methods used and the objectives for the design of the system architectures vary widely.

Table 1. Comparison of existing system architectures with the proposed system architecture.

Type of System Architecture	Resource Allocation Method	Literature Reference	Stakeholder Interactions	Objective of System Architecture
Marketplace	Different Types of Auctions	Wang et al. [33]	User to Provider-Business	Identification of Auction Implementation Issues
Marketplace	Combinatorial Double Auction	Samimi et al. [25]	User to Provider-Business	Feasibility Study
Middleware	Business Analytics Algorithms	Altmann et al. [23]	Provider-Scheduling to Provider-Business	Price Setting Considering Scheduling
Middleware	QoS Negotiation Platform	Buyya et al. [13]	User to Provider-Business	Combining Market-Based Objectives and Computing
Middleware	Business-Preference-Based Scheduling	This Article	User to Provider-Business to Provider-Scheduling	Scheduling that Considers Business Preferences

With respect to the type of system architectures, however, two types can be distinguished. The first type designs marketplaces for users and CSPs, to negotiate the price of services [38] [28]. The second type provides middleware, in which the economics-engineering functions are integrated [14] [22].

With respect to stakeholder interactions, all system architectures address the interaction between users and provider businesses [14] [22] [38] [28], focusing on assisting CSPs in setting prices. One middleware architecture also focuses on information forwarding from engineering-based scheduling to provider business [22]. Only the proposed system architecture also considers stakeholder interaction to provider scheduling.

Based on this comparison between the existing architecture and the proposed system architecture, it can be stated that the proposed system architecture represents the first step towards the development of a system architecture for optimizing scheduling based on business preferences. It interconnects the business aspects with engineering aspects.

4. Architecture Validation

4.1. Simulation Scenario

In order to demonstrate through simulations the workings of the proposed system architecture and, in particular, the business-preference-based scheduling, a scenario is assumed that considers two types of users with different preferences. Having different preferences makes the scenario applicable for economics-based resource allocation. In detail, the scenario considers a CSP, who offers two classes of services, namely, a premium service and a standard service. Users, who accept some delays in their task executions, can buy standard services (low-priced services), while users, who expect to experience no delay in any of their task executions, purchase premium services (high-priced services). If the aggregated demand of both types of users for resources is low (i.e., below the capacity of the executing hardware), both types of users get the same amount of resources for their tasks. If the aggregated demand for resources increases beyond the capacity, premium users get priority for obtaining resources, and standard users have to wait for tasks of premium users to finish. Therefore, standard users might experience delays or rejections of their task submissions.

To simplify the scenario, a few assumptions are made: Each task (i.e., VM) requires the same amount of resources; The hardware capacity handles up to 100 VMs/hour without quality degradation (i.e., the ready queue length is limited to 100 VMs/hour). If the number of VMs per hour exceeds 100VM/hour, quality degradation is experienced by users; The actual demand per hour is generated through two normal distributions.

To understand how the proposed system architecture operates at different levels of demand, two cases are distinguished: (a) the normal demand case, in which the number of VMs per hour is the sum of the draw from a normal distribution with a mean of 70 VM/hour (premium users) and from a normal distribution with a mean of 30 VM/hour (standard users); (b) the high demand case, in which the number of VMs per hour is the sum of the draw from a normal distribution with a mean of 80 VM/hour (premium users) and the draw from a normal distribution with a mean of 40 VM/hour (standard users). Figure 3 shows the VM demand in both cases for 60 time periods. For the normal demand case, the demand is above the hardware capacity (i.e., 100 VM/hour) in 27 time periods only. For the high demand case, all time periods but one are above the hardware capacity.

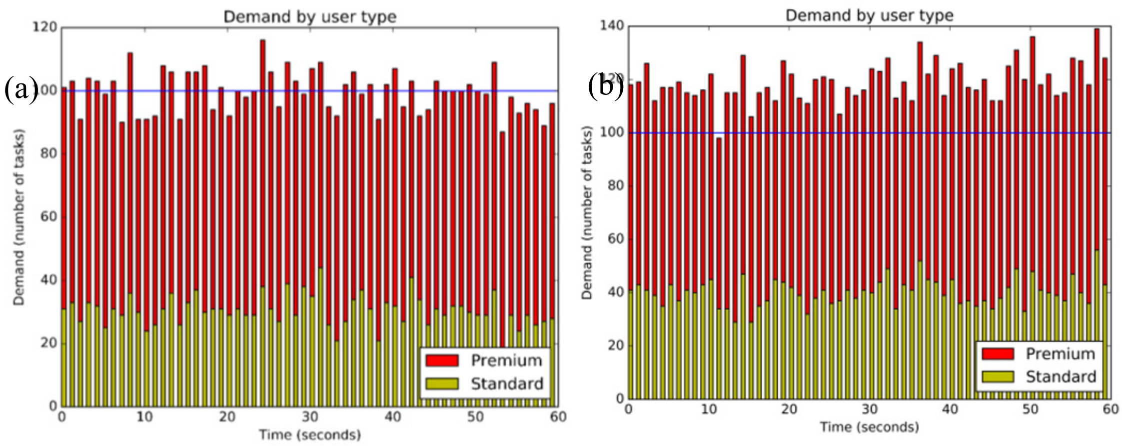

Figure 3. Generated VM demand of premium users and standard users for the two cases: (a) the normal demand case; (b) the high demand case.

The VM allocation integration technique, which is applied to the ready queue, is called expected marginal seat revenue (EMSR) and has been developed by Belobaba [39]. The probability density function $F_i()$ for the number of service requests di in each service class i (i.e., $i = 1$ (premium service) or $i = 2$ (standard service)) is determined by using historical data. The EMSR takes the form of Equation 1, where $F_i(\,)$ is the cumulative distribution function for the total number of VM requests C_i in service class i. C_1 indicates the number of VM requests for service class 1 that will be accepted (i.e., C_1 ready queue slots are allocated to service class 1). The number of ready queue slots C_2 allocated to service class 2 is the maximum ready queue length C minus C_1.

$$F_i(C_i) = \int_0^{C_i} f_i(d_i)\, \partial d_i \qquad (1)$$

Using Equation 1, the aim is to determine the optimal number of ready queue slots that should be allocated to the two service classes, such that the total revenue is maximized. In other words, the protection level (PL) for the premium services must be determined, as the booking limit (BL) for the standard services is calculated as BL = C - PL. The protection level for the premium services is the maximum value of C1 that satisfies the condition:

$$A * (1 - F_1(C_1)) \geq B \qquad (2)$$

The variables A and B represent the full price and the discounted price, respectively. In this scenario, the CSP charges a full price of A = 0.05$/h and a discounted price of B = 0.01$/h. Based on these prices, the generated demand data (Figure 3), and Equation 2, PL and BL are calculated. For allocating the resources, the new business-preference-based scheduling algorithm is used:

```
Step1:  Based on price A, price B, and the cumulative
        probability functions for both VM classes, the
        protection level PL and the booking limit BL are
        calculated;
Step2:  d1(t) = 0; d2(t) = 0; C(t) = 0;
Step3:  WAIT for VM service request from user;
Step4:  IF (C(t) ≤ d1(t) + d2(t)) THEN {
            reject request /* ready queue is full;
            GOTO Step3}
Step5:  IF (VM request is from the standard service
        class && d2(t) < BL) THEN {
            add VM request to ready queue at time t;
            d2(t) = d2(t) + 1;
            GOTO Step3}
Step6:  If (VM request is from the premium service class
        && d1(t) < PL) THEN {
             add VM request to ready queue at time t;
             d1(t) = d1(t) + 1}
Step7:  GOTO Step3;
```

4.2. Simulation Results

The total simulation time covers 5 days, which is split up into 60 time periods, representing one hour time slots. For demonstrating the effectiveness of the business-preference-based scheduling algorithm, the revenue generated through the algorithm during the 60 time periods was calculated and compared with the revenue of a FCFS scheduling algorithm (Figure 4). Based on the results of the comparison, it can be stated that the proposed algorithm generates more revenue than the FCFS algorithm. With respect to the normal demand case, the proposed algorithm generates revenue equal to \$497, while the FCFS algorithm achieves revenue of \$494. The difference of \$3 in revenue represents only a 0.74% increase in revenue. With respect to the high demand case, however, a more significant improvement in revenue is achieved. The proposed business-preference-based scheduling achieves revenue of \$534, compared to revenue of \$502 for the FCFS algorithm. This corresponds to an improvement of 6.4%.

Consequently, it can be stated that the proposed business-preference-based algorithm can optimize resource allocation based on the business strategy of a CSP. Moreover, it can be stated that the idea of interconnecting engineering-based resource allocation with the economics-based resource allocation is viable.

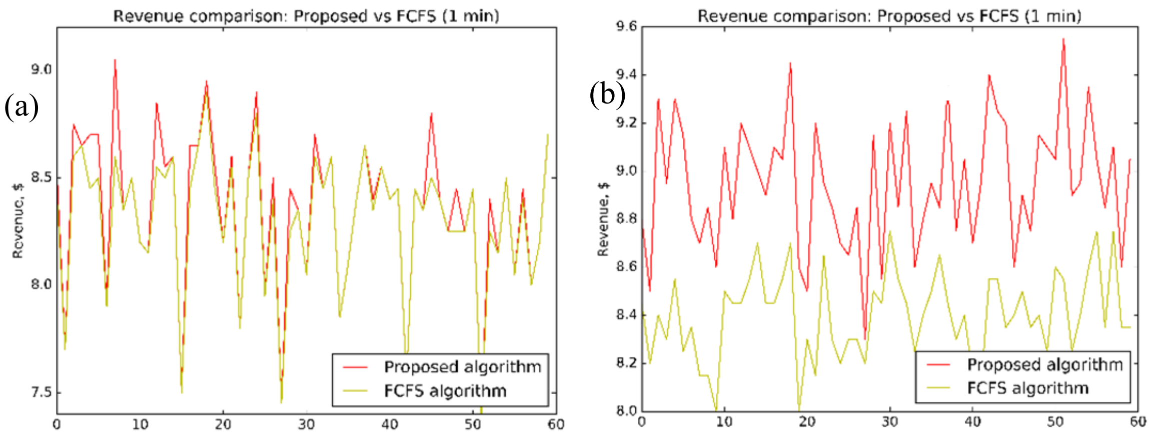

Figure 4. Revenue of the business-preference-based scheduling algorithm compared with the revenue of the FCFS scheduling algorithm for (a) the normal demand case and (b) the high demand case.

5. Conclusion

Although cloud computing has established itself as a beneficial technology, economics-based resource allocation, and engineering-based resource allocation are still separated. To address this source of inefficiency, this article proposes a system architecture for cloud computing that allows integrating economics aspects and engineering aspects in resource allocation. The integration of economics-based resource allocation and engineering-based resource allocation allows optimization at different levels.

The central module of the system architecture is the resource allocation integration module, which translates different pricing schemes (i.e., economics-based resource allocation method) into engineering-based scheduling by manipulating the ready queue. The module, which uses yield management as the economics-based resource allocation method, considers historical data to calculate the optimal protection level for deciding on the admittance of tasks to the ready queue. We name this scheduling algorithm as business-preference-based scheduling algorithm.

Our simulation results of the business-preference-based scheduling show that the proposed architecture generates higher revenue compared with a FCFS scheduling algorithm. This is very positive, as the current simulation considers only static pricing strategy with two price classes.

Our future work aims at extending this research, so that more complicated cases with a greater applicability in the real world can be modeled and assessed.

Acknowledgements

This research was conducted within the project BASMATI (Cloud Brokerage Across Borders for Mobile Users and Applications), which has received funding from the ICT R&D program of the Korean MSIP/IITP (R0115-16-0001) and from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 723131.

References

[1] Agarwal, D., Jain, S., 2014. Efficient optimal algorithm of task scheduling in cloud computing environment. arXiv Preprint arXiv:1404.2076.

[2] Al-Roomi, M., Al-Ebrahim, S., Buqrais, S., Ahmad, I., 2013. Cloud computing pricing models: a survey.

[3] Alam, B., 2013. Fuzzy round robin cpu scheduling algorithm. Journal of Computer Science, 9(8), 1079–1085.

[4] Altmann, J., Courcoubetis, C., Risch, M., 2010. A marketplace and its market mechanism for trading commoditized computing resources. Annals of Telecommunications-Annales Des Télécommunications, 65(11–12), 653–667.

[5] Altmann, J., Hovestadt, M., Kao, O., 2011. Business support service platform for providers in open cloud computing markets. In Networked Computing (INC), 2011 The 7th International Conference on pp. 149–154. IEEE.

[6] Altmann, J., Kashef, M. M., 2014. Cost model based service placement in federated hybrid clouds. Future Generation Computer Systems, 41, 79–90.

[7] Anandasivam, A., Neumann, D., 2009. Managing Revenue in Grids. 2009 42nd Hawaii International Conference on System Sciences. Springer-Verlag GmbH.

[8] Arlitt, M. F., Williamson, C. L., 1996. Web server workload characterization: The search for invariants. In ACM SIGMETRICS Performance Evaluation Review, Vol. 24, pp. 126–137. ACM.

[9] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Zaharia, M., 2010. A View of Cloud Computing. Commun. ACM, 53(4), 50–58.

[10] Belobaba, P. P., 1987. Survey paper-airline yield management an overview of seat inventory control. Transportation Science, 21(2), 63–73.

[11] Breskovic, I., Altmann, J., Brandic, I., 2013. Creating standardized products for electronic markets. Future Generation Computer Systems, 29(4), 1000–1011.

[12] Breskovic, I., Maurer, M., Emeakaroha, V. C., Brandic, I., Altmann, J., 2011. Towards Autonomic Market Management in Cloud Computing Infrastructures. In CLOSER, pp. 24–34.

[13] Buyya, R., Murshed, M., 2002. Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. Concurrency and Computation: Practice and Experience, 14(13□15), 1175–1220.

[14] Buyya, R., Yeo, C. S., Venugopal, S., 2008. Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on, pp. 5–13. Ieee.

[15] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., Brandic, I., 2009. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems, 25(6), 599–616.

[16] Cherkasova, L., Gupta, M., 2004. Analysis of enterprise media server workloads: access patterns, locality, content evolution, and rates of change. Networking, IEEE/ACM Transactions on, 12(5), 781–794.

[17] Dong, F., Akl, S. G., 2006. Scheduling algorithms for grid computing: State of the art and open problems. Technical report.

[18] Foster, I., Zhao, Y., Raicu, I., Lu, S., 2008. Cloud Computing and Grid Computing 360-Degree Compared. 2008 Grid Computing Environments Workshop, abs/0901.0(5), 1–10.

[19] Gmach, D., Rolia, J., Cherkasova, L., Kemper, A., 2007. Workload analysis and demand prediction of enterprise data center applications. In Workload Characterization, 2007. IISWC 2007. IEEE 10th International Symposium on (pp. 171–180). IEEE.

[20] Hamsanandhini, S., Mohana, R. S., 2015. Maximizing the revenue with client classification in Cloud Computing market. In Computer Communication and Informatics (ICCCI), 2015 International Conference on pp. 1–7. IEEE.

[21] Jallat, F., Ancarani, F., 2008. Yield management, dynamic pricing and CRM in telecommunications. Journal of Services Marketing, 22(6), 465–478.

[22] Jeferry, K., Kousiouris, G., Kyriazis, D., Altmann, J., Ciuffoletti, A., Maglogiannis, I., Zhao, Z., 2015. Challenges emerging from future cloud application scenarios. Procedia Computer Science, 68, 227–237.

[23] Kashef, M. M., Uzbekov, A., Altmann, J., Hovestadt, M., 2013. Comparison of two yield management strategies for cloud service providers. In International Conference on Grid and Pervasive Computing pp. 170–180. Springer.

[24] Khankasikam, K., 2013. An Adaptive Round Robin Scheduling Algorithm: A Dynamic Time Quantum Approach. International Journal of Advancements in Computing Technology, 5(1).

[25] Kimes, S. E., 1989. The Basics of Yield Management. Cornell Hotel and Restaurant Administration Quarterly, 30(3), 14–19.

[26] Mell, P., Grance, T., 2011. The NIST definition of cloud computing.

[27] Mishra, M. K., Rashid, F., 2014. An Improved Round Robin CPU Scheduling Algorithm With Varying Time Quantum. International Journal of Computer Science, Engineering and Applications, 4(4), 1.

[28] Netessine, S., Shumsky, R., 2002. Introduction to the theory and practice of yield management. INFORMS Transactions on Education, 3(1), 34–44.

[29] Osterwalder, A., 2004. The business model ontology: A proposition in a design science approach.

[30] Rimal, B. P., Choi, E., Lumb, I., 2009. A taxonomy and survey of cloud computing systems. INC, IMS and IDC, 44–51.

[31] Risch, M., Altmann, J., Guo, L., Fleming, A., Courcoubetis, C., 2009. The gridecon platform: A business scenario testbed for commercial cloud services. In International Workshop on Grid Economics and Business Models pp. 46–59. Springer.

[32] Ru, J., Keung, J., 2013. An Empirical investigation on the simulation of priority and shortest-job-first scheduling for cloud-based software systems. In Software Engineering Conference (ASWEC), 2013 22nd Australian pp. 78–87. IEEE.

[33] Samimi, P., Teimouri, Y., Mukhtar, M., 2014. A combinatorial double auction resource allocation model in cloud computing. Information Sciences.

[34] Sirohi, A., Pratap, A., Aggarwal, M., 2014. Improvised Round Robin (CPU) Scheduling Algorithm. International Journal of Computer Applications, 99(18), 40–43.

[35] Srinivasan, S., Kettimuthu, R., Subramani, V., Sadayappan, P., 2002. Characterization of backfilling strategies for parallel job scheduling. In Parallel Processing Workshops, 2002. Proceedings. International Conference on pp. 514–519. IEEE.

[36] Teng, F., Magoules, F.б 2010. Resource pricing and equilibrium allocation policy in cloud computing. In Computer and information technology (CIT), 2010 IEEE 10th international conference on pp. 195–202. IEEE.

[37] Wang, H., Tianfield, H., Mair, Q., 2014. Auction based resource allocation in cloud computing. Multiagent and Grid Systems, 10(1), 51–66.

[38] Weinhardt, C., Anandasivam, A., Blau, B., Borissov, N., Meinl, T., Michalk, W., Stößer, J., 2009. Cloud Computing -- A Classification, Business Models, and Research Directions. Business {&} Information Systems Engineering, 1(5), 391–399.

[39] Zhang, Q., Cheng, L., Boutaba, R., 2010. Cloud computing: state-of-the-art and research challenges. Journal of Internet Services and Applications, 1(1), 7–18.

TEMEP Discussion Papers

2011-85: Romain Lestage and David Flacher, “Access Regulation and Welfare”

2012-86: Juthasit Rohitratana and Jörn Altmann, “Impact of Pricing Schemes on a Market for Software-as-a-Service and Perpetual Software”

2012-87: Bory Seng, “The Introduction of ICT for Sustainable Development of the Tourism Industry in Cambodia”

2012-88: Jörn Altmann, Martina Meschke and Ashraf Bany Mohammed, “A Classification Scheme for Characterizing Service Networks”

2012-89: Nabaz T. Khayyat and Almas Heshmati, “Determinants of Mobile Telecommunication Adoption in the Kurdistan Region of Iraq”

2012-90: Nabaz T. Khayyat and Almas Heshmati, “Determinants of Mobile Phone Customer Satisfaction in the Kurdistan Region of Iraq”

2012-91: Nabaz T. Khayyat and Jeong-Dong Lee, “A New Index Measure of Technological Capabilities for Developing Countries”

2012-92: Juseuk Kim, Jörn Altmann and Lynn Ilon, “Using Smartphone Apps for Learning in a Major Korean University”

2012-93: Lynn Ilon and Jörn Altmann, “Using Collective Adaptive Networks to Solve Education Problems in Poor Countries”

2012-94: Ahmad Mohammad Hassan, Jörn Altmann and Victor Lopez, “Control Plane Framework Emergence and its Deployment Cost Estimation”

2012-95: Selam Abrham Gebregiorgis and Jörn Altmann, “IT Service Platforms: Their Value Creation Model and the Impact of their Level of Openness on their Adoption”

2012-96: Ivan Breskovic, Jörn Altmann and Ivona Brandic, “Creating Standardized Products for Electronic Markets”

2012-97: Netsanet Haile and Jörn Altmann, “Value Creation in IT Service Platforms through Two-Sided Network Effects”

2012-98: Soyoung Kim, Junseok Hwang and Jörn Altmann, “Dynamic Scenarios of Trust Establishment in the Public Cloud Service Market”

2012-99: Lynn Ilon, “How Collective Intelligence Redefines Education”

2013-100: Ivan Breskovic, Ivona Brandic and Jörn Altmann, “Maximizing Liquidity in Cloud Markets through Standardization of Computational Resources”

2013-101: Juseuk Kim, Lynn Ilon and Jörn Altmann, “Adapting Smartphones as Learning Technology in a Korean University”

2013-102: Baseem Al-Athwari, Jörn Altmann and Almas Heshmati, “A Conceptual Model and Methodology for Evaluating E-Infrastructure Deployment and Its Application to OECD and MENA Countries”

2013-103: Mohammad Mahdi Kashef, Azamat Uzbekov, Jörn Altmann and Matthias Hovestadt, "Comparison of Two Yield Management Strategies for Cloud Service Providers"

2013-104: Kibae Kim, Wool-Rim Lee and Jörn Altmann, "Patterns of Innovation in SaaS Networks: Trend Analysis of Node Centralities"

2013-105: Netsanet Haile and Jörn Altmann, "Estimating the Value Obtained from Using a Software Service Platform"

2013-106: A. Mohammad Hassan and Jörn Altmann, "Business Impact Analysis of a Mediator between the Network Management Systems of the IP/MPLS Network and the Transport Network"

2013-107: A. Mohammad Hassan and Jörn Altmann, "Disjoint Paths Pair Computation Procedure for SDH/SONET Networks"

2013-108: Kibae Kim and Jörn Altmann, "Evolution of the Software-as-a-Service Innovation System through Collective Intelligence"

2014-109: Shahrouz Abolhosseini, Almas Heshmati and Jörn Altmann, "The Effect of Renewable Energy Development on Carbon Emission Reduction: An Empirical Analysis for the EU-15 Countries"

2014-110: Somayeh Koohborfardhaghighi and Jörn Altmann, "How Placing Limitations on the Size of Personal Networks Changes the Structural Properties of Complex Networks"

2014-111: Somayeh Koohborfardhaghighi and Jörn Altmann, "How Structural Changes in Complex Networks Impact Organizatioanl Learning Performance"

2014-112: Sodam Baek, Kibae Kim and Jörn Altmann, "Role of Platform Providers in Service Networks: The Case of Salesforce.com App Exchange"

2014-113: Somayeh Koohborfardhaghighi and Jörn Altmann, "A Network Formation Model for Social Object Networks"

2014-114: Somayeh Koohborfardhaghighi and Jörn Altmann, "How Variability in Individual Patterns of Behavior Changes the Structural Properties of Networks"

2014-115: Kibae Kim, Wool-Rim Lee and Jörn Altmann, "SNA-Based Innovation Trend Analysis in Software Service Networks"

2014-116: Jörn Altmann and Mohammad Mahdi Kashef, "Cost Model Based Service Placement in Federated Hybrid Clouds"

2014-117: Kibae Kim, Songhee Kang and Jörn Altmann, "Cloud Goliath Versus a Federation of Cloud Davids: Survey of Economic Theories on Cloud Federation"

2014-118: Stefan Niederhafner, "The Korean Energy and GHG Target Management System: An Alternative to Kyoto-Protocol Emissions Trading Systems?"

2014-119: Gunno Park and Jina Kang, "Effect of Alliance Experience on New Alliance Formations and Internal R&D Capabilities"

2015-120: Kibae Kim, Jörn Altmann and Sodam Baek, "Role of Platform Providers in Software Ecosystems"

2015-121: Kibae Kim and Jörn Altmann, "Effect of Homophily on Network Evolution"

2015-122: Netsanet Haile and Jörn Altmann, "Risk-Benefit-Mediated Impact of Determinants on the Adoption of Cloud Federation"

2015-123: Netsanet Haile and Jörn Altmann, "Value Creation in Software Service Platforms"

2015-124: Kibae Kim, "Evolution of the Global Knowledge Network: Network Analysis of Information and Communication Technologies' Patents"

2015-125: Baseem Al-athwari and Jörn Altmann, "Utility-Based Smartphone Energy Consumption Optimization for Cloud-Based and On-Device Application Uses"

2015-126: Keith Jeferry, George Kousiouris, Dimosthenis Kyriazis, Jörn Altmann, Augusto Ciuffoletti, Ilias Maglogiannis, Paolo Nesi, Bojan Suzic and Zhiming Zhao, "Challenges Emerging from Future Cloud Application Scenarios"

2015-127: Netsanet Haile and Jörn Altmann, "Structural Analysis of Value Creation in Software Service Platforms"

2015-128: Yeongjun Yeo Dongnyok Shim, Jeong-Dong Lee and Jörn Altmann, "Driving Forces of CO2 Emissions in Emerging Countries: LMDI Decomposition Analysis on China and India's Residential Sector"

2016-129: Dongnyok Shim, Jin Gyo Kim and Jörn Altmann, "Identifying Key Drivers and Bottlenecks in the Adoption of E-book Readers in Korea"

2016-130: Somayeh Koohborfardhaghighi and Jörn Altmann, "How Network Visibility and Strategic Networking Leads to the Emergence of Certain Network Characteristics: A Complex Adaptive System Approach"

2016-131: Somayeh Koohborfardhaghighi and Jörn Altmann, "How Strategic Networking Impacts the Networking Outcome: A Complex Adaptive System Approach"

2016-132: Jörn Altmann, Emanuele Carlini, Massimo Coppola, Patrizio Dazzi, Ana Juan Ferrer, Netsanet Haile, Young-Woo Jung, Dong-Jae Kang, Iain-James Marshall, Konstantinos Tserpes and Theodora Varvarigou, "BASMATI - A Brokerage Architecture on Federated Clouds for Mobile Applications"

2016-133 Dongnyok Shim and Jörn Altmann, "How Marginally Does Impulse Buying Intention Change in Social Commerce? Nonparametric Regression Approach"

2016-134: Azamat Uzbekov, Jörn Altmann, "Enabling Business-Preference-Based Scheduling of Cloud Computing Resources"