



Customer-Satisfaction-Aware Optimal Multi-server Configuration for Profit Maximization in Cloud Computing

Duggirala Siva Sai | P. Venkata Siva

Department of Computer Science and Engineering, Chalapathi Institute of Engineering and Technology, Lam, Guntur

*Corresponding Author Email ID: saisiva164@gmail.com, siva11992@gmail.com

To Cite this Article

Duggirala Siva Sai and P. Venkata Siva. Customer-Satisfaction-Aware Optimal Multi-server Configuration for Profit Maximization in Cloud Computing. International Journal for Modern Trends in Science and Technology 2022, 8(05), pp. 259-264. <https://doi.org/10.46501/IJMTST0805037>

Article Info

Received: 02 April 2022; Accepted: 06 May 2022; Published: 09 May 2022.

ABSTRACT

Along with the development of cloud computing, an increasing number of enterprises start to adopt cloud service, which promotes the emergence of many cloud service providers. For cloud service providers, how to configure their cloud service platforms to obtain the maximum profit becomes increasingly the focus that they pay attention to. In this paper, we take customer satisfaction into consideration to address this problem. Customer satisfaction affects the profit of cloud service providers in two ways. On one hand, the cloud configuration affects the quality of service which is an important factor affecting customer satisfaction. On the other hand, the customer satisfaction affects the request arrival rate of a cloud service provider. However, few existing works take customer satisfaction into consideration in solving profit maximization problem, or the existing works considering customer satisfaction do not give a proper formalized definition for it. Hence, we firstly refer to the definition of customer satisfaction in economics and develop a formula for measuring customer satisfaction in cloud computing. And then, an analysis is given in detail on how the customer satisfaction affects the profit. Lastly, taking into consideration customer satisfaction, service-level agreement, renting price, energy consumption and so forth, a profit maximization problem is formulated and solved to get the optimal configuration such that the profit is maximized. (Abstract)

KEYWORDS—Admin, Customer, Accept Request, Cloud Request, Cloud Admin, Cloud Space.

1. INTRODUCTION

Secure Computing

Computer security (Also known as cyber security or IT Security) is information security as applied to computers and networks. The field covers all the processes and mechanisms by which computer-based equipment, information and services are protected from unintended or unauthorized access, change or destruction. Computer security also includes protection from unplanned events and natural disasters. Otherwise, in the computer

industry, the term security -- or the phrase computer security -- refers to techniques for ensuring that data stored in a computer cannot be read or compromised by any individuals without authorization. Most computer security measures involve data encryption and passwords. Data encryption is the translation of data into a form that is unintelligible without a deciphering mechanism. A password is a secret word or phrase that gives a user access to a particular program or system.

Working conditions and basic needs in the secure computing:

If you don't take basic steps to protect your work computer, you put it and all the information on it at risk. You can potentially compromise the operation of other computers on your organization's network, or even the functioning of the network as a whole.

2. SYSTEM ANALYSIS

2.1 Existing

- Chen *et al.* adopted utility theory leveraged from economics and developed an utility model for measuring customer satisfaction in cloud. In the utility model, consumer satisfaction is relevant to two factors: service price and response time. They assumed that consumer satisfaction is decreased with higher service price and longer response time.
- In other work, the user satisfaction is calculated as the ratio of the actual QoS level and the expected QoS level.
- Wu *et al.* proposed an admission control and scheduling algorithms for SaaS providers to maximize profit by minimizing cost and improve customer satisfaction level. However, they did not give a specific formula to measure customer satisfaction level.
- Chao *et al.* proposed a customer satisfaction aware algorithm based on the Ant-Colony Optimization (AMP) for geo-distributed datacenters.

2.2 Disadvantages of Existing system

- The request arrival rate of a service provider is affected by many factors in actual, and customer satisfaction is the most important factor
- Few existing works take customer satisfaction into consideration in solving profit maximization problem, or the existing works considering customer satisfaction do not give a proper formalized definition for it.
- The existing formulas measuring customer satisfaction of cloud computing cannot properly reflect the definition of customer satisfaction, and they did not take into account user's psychological differences

2.3 Proposed System

- This paper adopts the thought in Business Administration, and firstly defines the *customer satisfaction level* of cloud computing.

- Based on the definition of customer satisfaction, we build a profit maximization model in which the effect of customer satisfaction on quality of service (QoS) and price of service (PoS) is considered.
- In this paper, we build a customer satisfaction-aware profit optimization model and propose a discrete hill climbing algorithm to find the numeric optimal cloud configuration for cloud service providers.

3. THE COST MODEL

CLOUD SERVICE MODEL

The profit optimization of cloud service providers, we should study how to measure the customer satisfaction of a cloud service provider and how the customer satisfaction affects the its profit, which are analyzed in the following. In Cloud Server the M/M/m model, m is the number of servers, and all servers run at an identical speed s (measured by the number of instructions that can be executed in one unit of time). Assume that the interarrival times of service requests are independent and identically distributed exponential random variables, in other words, the arrival requests follow a Poisson process with arrival rate λ . The execution requirements of the tasks (measured by the number of instructions to be executed) are exponential random variables r with mean \bar{r} . Since the server execution speed is s, the service times of the requests are also exponential random variables $x = r/s$ with mean $\bar{x} = \bar{r}/s$. Hence, the average service rate, i.e., the average number of service requests that can be completed by a server with speed s in one unit of time, is $\mu = 1/\bar{x} = s/\bar{r}$.

SERVICE-LEVEL AGREEMENT

In general, the QoS is affected by many factors such as the service time, the failure rate and so forth. However, in this paper, we measure the QoS of a request by its response time for two reasons. First, the service time is easily measured. Second, it gives customers an intuitive feeling of QoS. For customers, they do not care how failures are managed when failures occur. They only care whether the task can be completed successfully and how long it takes.

SYSTEM STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to

be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

- **Economical Feasibility**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

- **Technical Feasibility**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

- **Social Feasibility**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

5. DESIGN ANALYSIS

5.1 UML Diagrams

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the

field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

Definition

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group

UML is a language

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

UML Constructing

UML models can be directly connected to a variety of programming languages and it is sufficiently expressive and free from any ambiguity to permit the direct execution of models.

5.2 UML Documenting

UML provides variety of documents in addition raw executable codes. The use case view of a system encompasses the use cases that describe the behavior of the system as seen by its end users, analysts, and testers.

The design view of a system encompasses the classes, interfaces, and collaborations that form the vocabulary of the problem and its solution.

The process view of a system encompasses the threads and processes that form the system's concurrency and synchronization mechanisms.

The implementation view of a system encompasses the components and files that are used to assemble and release the physical system. The development view of a system encompasses the nodes that from the system's hardware topology on which the system executes.

5.3 Uses of UML

The UML is intended primarily for software intensive systems. It has been used effectively for such domain as Enterprise Information System, Banking and Financial Services, Telecommunications

5.4 Building blocks of UML

The vocabulary of the UML Encompasses 3 kinds of building blocks

- Things
- Relationships
- Diagrams

6. SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

- **Types of tests**
- **Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Unit testing in the protection system

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

TEST RESULT

POSITIVE TEST CASES

CASE 1: LOGIN FORM:

SL.No	Test Case	Expected Result	Test Result
1	Enter valid name and password & click on login button	Software should display main window	Successful
2	Enter invalid	Software should not display main window	successful

CASE 2:

S.NO	Test Case	Input	Expected Output	Obtained Output
1	Admin Login	Login the owner	Login page open	Login page open Owner page open
2	User Login	Register the user	User page open	User page open
3	File Upload	Fill the details	Select a file	Upload a file
4	File Download	Search the items	View the item type	Download file
5	Machines	Searches the machine	Accepts the requests	Send the file

6	Save	Download the file	Open the location	Save the file
---	------	-------------------	-------------------	---------------

NEGATIVE TEST CASES

CASE 3: LOGIN FORM:

SL.No	Test Case	Excepted Result	Test Result
1	Enter invalid name and password & click on login button	Software should display main window	UnSuccessful
2	Enter invalid	Software should not display main window	Unsuccessful

7. CONCLUSION

In this paper, we consider customer satisfaction in solving optimal configuration problem with profit maximization. Because the existing works do not give a proper definition and calculation formula for customer satisfaction, hence, we first give a definition of customer satisfaction leveraged from economics and develop a formula for measuring customer satisfaction in cloud. Based on the affection of customer satisfaction on workload, we analyze the interaction between the market demand and the customer satisfaction, and give the calculation of the actual task arrival rate under different configurations. In addition, we study an optimal configuration problem of profit maximization. The optimal solutions are solved by a discrete hill climbing algorithm. Lastly, a series of calculations are conducted to analyze the changing trend of profit. Moreover, a group of calculations are conducted to compare the profit and optimal configuration of two situations with and without considering the affection of customer satisfaction on customer demand. The results show that when considering customer satisfaction, our model performs better in overall.

8. FUTURE SCOPE

Cloud computing is an emerging computing paradigm in which resources of the computing infrastructure are provided as services over the Internet. As promising as it is, this paradigm also brings forth many new challenges for data security and access control when users outsource

sensitive data for sharing on cloud servers, which are not within the same trusted domain as data owners. To keep sensitive user data confidential against untrusted servers, existing solutions usually apply cryptographic methods by disclosing data decryption keys only to authorized users. However, in doing so, these solutions inevitably introduce a heavy computation overhead on the data owner for key distribution and data management when fine-grained data access control is desired, and thus do not scale well. The problem of simultaneously achieving fine-grainedness, scalability, and data confidentiality of access control actually still remains unresolved. This paper addresses this challenging open issue by, on one hand, defining and enforcing access policies based on data attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in fine-grained data access control to untrusted cloud servers without disclosing the underlying data contents. We achieve this goal by exploiting and uniquely combining techniques of attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption. Our proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability. Extensive analysis shows that our proposed scheme is highly efficient and provably secure under existing security models.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

- [1] P. Mell and T. Grance, "The nist definition of cloud computing," *Communications of the Acm*, vol. 53, no. 6, pp. 50–50, 2011.
- [2] J. Cao, K. Hwang, K. Li, and A. Y. Zomaya, "Optimal multiserver configuration for profit maximization in cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1087–1096, 2013.
- [3] "Amazon EC2," <http://aws.amazon.com>, 2015.
- [4] "Microsoft Azure," <http://www.microsoft.com/windowsazure>, 2015.
- [5] "Salesforce.com," <http://www.salesforce.com/au>, 2014.
- [6] J. Mei, K. Li, A. Ouyang, and K. Li, "A profit maximization scheme with guaranteed quality of service in cloud computing," *IEEE Trans. Computers*, vol. 64, no. 11, pp. 3064–3078, Nov 2015.
- [7] R. N. Cardozo, "An experimental study of customer effort, expectation, and satisfaction," *Journal of marketing research*, pp. 244–249, 1965.
- [8] J. A. Howard and J. N. Sheth, *The theory of buyer behavior*. Wiley New York, 1969, vol. 14.

- [9] G. A. Churchill Jr and C. Surprenant, "An investigation into the determinants of customer satisfaction," *Journal of marketing research*, pp. 491–504, 1982.
- [10] D. K. Tse and P. C. Wilton, "Models of consumer satisfaction formation: An extension," *Journal of marketing research*, pp. 204–212, 1988.
- [11] A. Parasuraman, V. A. Zeithaml, and L. L. Berry, "Reassessment of expectations as a comparison standard in measuring service quality: implications for further research," *the Journal of Marketing*, pp. 111–124, 1994.
- [12] K. Medigovich, D. Porock, L. Kristjanson, and M. Smith, "Predictors of family satisfaction with an Australian palliative home care service: a test of discrepancy theory," *Journal of palliative care*, vol. 15, no. 4, p. 48156, 1999.
- [13] J. J. Jiang, G. Klein, and C. Saunders, *Discrepancy Theory Models of Satisfaction in IS Research*. New York, NY: Springer New York, 2012, pp. 355–381.
- [14] Y. Hu, J. Wong, G. Iszlai, and M. Litoiu, "Resource provisioning for cloud computing," in *Proceedings of the 2009 Conference of the Center for Advanced Studies on Collaborative Research*. IBM Corp., 2009, pp. 101–111.
- [15] M. Mazzucco, D. Dyachuk, and R. Deters, "Maximizing cloud providers' revenues via energy aware allocation policies," in *2010 IEEE 3rd International Conference on Cloud Computing (CLOUD)*. IEEE, 2010, pp. 131–138.
- [16] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy aware resource allocation heuristics for efficient management of data centers for cloud computing," *Future Generation Computer Systems*, vol. 28, no. 5, pp. 755–768, 2012.
- [17] J. Cao, K. Li, and I. Stojmenovic, "Optimal power allocation and load distribution for multiple heterogeneous multicore server processors across clouds and data centers," *IEEE Trans. Computers*, vol. 63, no. 1, pp. 45–58, 2014.
- [18] S. Chaisiri, B.-S. Lee, and D. Niyato, "Profit maximization model for cloud provider based on windows azure platform," in *2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, May 2012, pp. 1–4.
- [19] S. Liu, S. Ren, G. Quan, M. Zhao, and S. Ren, "Profit aware load balancing for distributed cloud data centers," in *2013 IEEE 27th International Symposium on Parallel & Distributed Processing (IPDPS)*. IEEE, 2013, pp. 611–622.
- [20] J. Chen, C. Wang, B. B. Zhou, L. Sun, Y. C. Lee, and A. Y. Zomaya, "Tradeoffs between profit and customer satisfaction for service provisioning in the cloud," in *Proceedings of the 20th international symposium on High performance distributed computing*. ACM, 2011, pp. 229–238.
- [21] W. Gao and F. Kang, "Cloud simulation resource scheduling algorithm based on multi-dimension quality of service," *Information Technology Journal*, vol. 11, no. 1, pp. 94–101, 2012.
- [22] L. Wu, S. K. Garg, and R. Buyya, "Sla-based admission control for a software-as-a-service provider in cloud computing environments," *Journal of Computer and System Sciences*, vol. 78, no. 5, pp. 1280–1299, 2012.
- [23] K. C. Wu, H. C. Jiau, and K.-F. Ssu, "Improving consumer satisfaction through building an allocation cloud," in *The Fifth International Conference on Dependability (DEPEND 2012)*, 2012, pp. 31–37.
- [24] C. Jing, Y. Zhu, and M. Li, "Customer satisfaction aware scheduling for utility maximization on geodistributed cloud data centers," in *2013 IEEE International Conference on HPCC EUC*. IEEE, 2013, pp. 218–225.
- [25] K. Tsakalozos, H. Kllapi, E. Sitaridi, M. Roussopoulos, D. Paparas, and A. Delis, "Flexible use of cloud resources through profit maximization and price discrimination," in *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*, April 2011, pp. 75–86.
- [26] R. Chen, Y. Zhang, and D. Zhang, "A cloud task scheduling algorithm based on users' satisfaction," in *2013 Fourth International Conference on Networking and Distributed Computing (ICNDC)*, Dec 2013, pp. 1–5.
- [27] M. Unuvar, S. Tosi, Y. Doganata, M. Steinder, and A. Tantawi, "Selecting optimum cloud availability zones by learning user satisfaction levels," *Services Computing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.
- [28] H. Morshedlou and M. Meybodi, "Decreasing impact of violations: a proactive resource allocation approach for cloud computing environments," *IEEE*