International Journal of Grid and Distributed Computing
Vol. 9, No. 6 (2016), pp.169-180
http://dx.doi.org/10.14257/ijgdc.2016.9.6.17

Evaluation Model Queuing Task Scheduling Based on Hybrid Architecture Cloud Systems

Zeyu Sun[1, 2], Yaping Li[1], Yangjie Cao[3*], Yuanbo Li[1]

[1]Computer and Information Engineering,
Luoyang Institute of Science and Technology, Luoyang 471023, China.
[2]Electrical and information Engineering,
Xi'an Jiaotong University, Xi'an 710049, China
[3]School of Software Technology, Zhengzhou University,
Zhengzhou, 450000 China
lylgszy@163.com

Abstract

The applications based on cloud computing platform usually need to use a number of computing resources and storage resources to completing computing tasks, so the fault-tolerant capability of system has become increasingly important. Aiming to solve this problem, an evaluation model of task scheduling is proposed based on cloud system (TSCS). TSCS can effectively model and simulate complex cloud systems due to its strong capabilities of quantitative evaluation and behavioral description resulting from combining the theoretical characteristics of queuing task theory and Petri net. The algorithm solves the problem that meeting customer service satisfaction and load balancing at the same time. In addition, consider single backup task status, for the failure of more than one processor at the same time, present the minimum cost of backup scheduling algorithm, the algorithm to solve the problem that require a lot of backup cost. Experimental results show that TSCS is able to effectively reflect the architecture characteristics of various cloud system at the perspectives of performance, service, etc., and highly simulated various kinds of dynamic service behaviors of cloud system, single workload and multi workloads shows that the proposed policy can finish the user's queuing task scheduling before deadline as well as obtain approving cost efficient.

***Keywords**: Cloud System; Task Scheduling; Wireless Networks; Evaluation Model*

1. Introduction

The cloud computing of wireless network had laid the foundation theory and application [1-3]. cloud computing has rapidly developed in the fields such as government, economy and scientific research because of its advantages involving low cost, high expansibility, simplicity and so forth. Various cloud systems with different architectures, technologies and goals have appeared and show fast popularization and application. With the promotion of the world's leading companies, Cloud computing is attracting more and more attention for providing a new type of information and services that broadens new vision of information technology (IT) services. Commercial Cloud service providers offer computing facilities and resources to users in a manner of on-demand, pay-as-you-go model and in charge of fee, which is also known as Infrastructure as a Service (IaaS) providers.

It is one of the key factors for successfully implementing cloud computing to find out the ways for efficiently and accurately evaluating cloud architectures considering diverse cloud systems with miscellaneous technologies. Essentially, cloud computing is a kind of

Yangjie Cao is the corresponding author.

ISSN: 2005-4262 IJGDC

new service mode, in which, the service types and characteristics required by different service fields may significantly differ from each other. Therefore, it is difficult to obtain a kind of architecture or scheme of cloud computing which can provide optimal service and adaptability for all fields [4-5]. Thus, it is more likely to result in problems such as high cost of cloud computing, low quality of services and waste of resources to deploy and construct cloud computing blindly in the absence of evaluation and prediction. Consequently, it fails to effectively satisfy practical service demands and performance requirements [6-7].

In order to evaluate cloud computing system, traditional evaluating techniques such as simulation systems, prototype system and static mathematical model have been employed in many investigations. However, the characteristics such as huge public scale, complex structures, outstanding dynamics, and strong associations among layers may induce defects including poor capability of index quantification, high evaluation cost, low simulation degree, limited scale, *etc* [8-9]. in practical evaluation process. Generally, the frequent lack of adequate theoretical support of simulating evaluation system makes it difficult to effectively simulate the new characteristics and details of features. Especially, the characteristics of cloud computing including large amount of users, complicated behaviors and dispersed distribution bring more difficulties to simulate cloud computing environments. Moreover, prototype system calls for high evaluation cost, and is difficult to be changed once achieved [10-12]; therefore, it impedes the comparative evaluation on multiple architectures and platforms. As the scale and structure of cloud system are increasingly complex, the formalized model of the cloud system is increasingly huge and the complexity increases exponentially. Traditional methods based on mathematical derivation and theoretical explanation also fail to solve the problem of the explosive growths of data quantity, data amount and calculation amount existing in the analytical process of the formalized model of cloud systems. Stochastic queuing task Petri net [1], as a kind of powerful graphics-oriented modeling language, not only can describe complex architectures of cloud computing, but also is able to dynamically analyze all kinds of behaviors of cloud computing system strongly. It inherits and develops the advantages of queuing task network model and stochastic Petri net. Therefore, in addition to quantitatively modeling the operation performance of sources, it also can depict the dependency relationship among multilayered systems considering the characteristics of cloud system involving diverse application service, complex service behavior and superimposed hierarchical structures [13]. Therefore, Petri net is the first choice for system modeling because it is applicable for the evaluation and prediction of cloud computing both in terms of theoretical support and quantitative analysis.

According to the aforementioned problems, an evaluation model of cloud system on the basis of Petri net was proposed. Based on the system features of “Chinese cloud” of the State Key Laboratory of High-end Server & Storage Technology, a detailed and complete simulation modeling was performed on typical cloud architecture using this model by combining the characteristics of queuing task Petri net theory. On the basis of this, the dynamic service performance provided by cloud computing was quantitatively analyzed. Experimental results revealed that the evaluation model was capable of reflecting the service characteristics and performance features of various kinds of cloud architectures. Meanwhile, the results of quantitative analysis were valuable references for promoting further development and application of cloud computing.

2. Cloud System Architectures

Figure 1 illustrates the basic working process, during which, cloud platform deals with user’s tasks and works in accordance with the characteristics of the “Chinese cloud” project in terms of basic architectures. Dispatchers correspondingly distribute the works and tasks which reach the underlying resources. The obtainment of works and tasks by a computational node calls for a corresponding treatment, during which, the communication among computer nodes through the network is needed so as to interact computing information. Meanwhile, the computing nodes are also asked to communicate with storage nodes in order to fulfill the I/O operation of data, the communication of which can be accomplished through network or directly through SCSI, *etc*. The data of final results are returned to users via the network. Additionally, in some cases, the transfer operation of data among storage devices is conducted through the network as well.

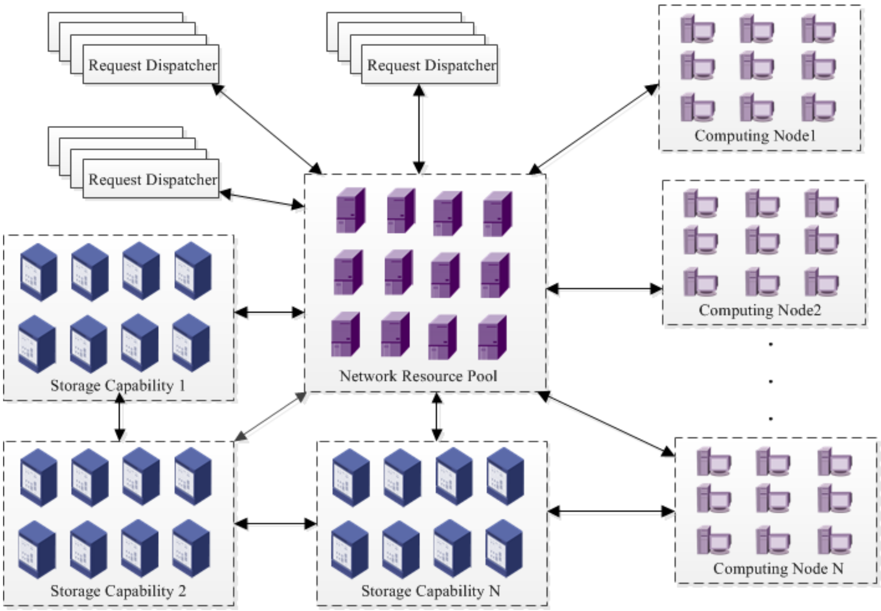


Figure 1. Task Scheduling on Cloud Platforms

Based on the current cloud systems in the world, the Chinese cloud system proposed three representative and universal architectures of cloud system as the alternative scheme of the system architecture of Chinese cloud by combing the characteristics of the project. They are distributed, centralized and hybrid architectures of cloud system, as shown in Figures 2, 3 and 4, respectively, and described in details as follows.

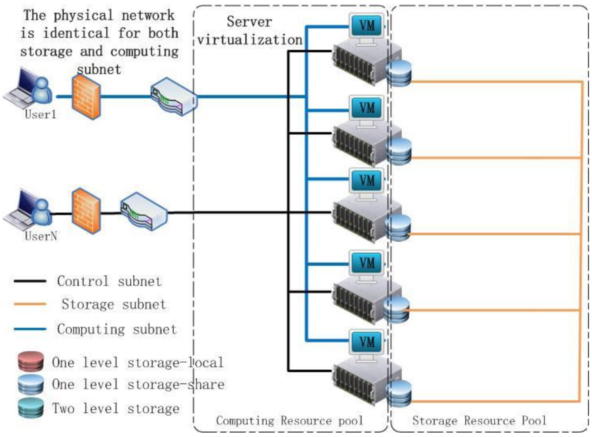


Figure 2. Distributed Architecture of Cloud System

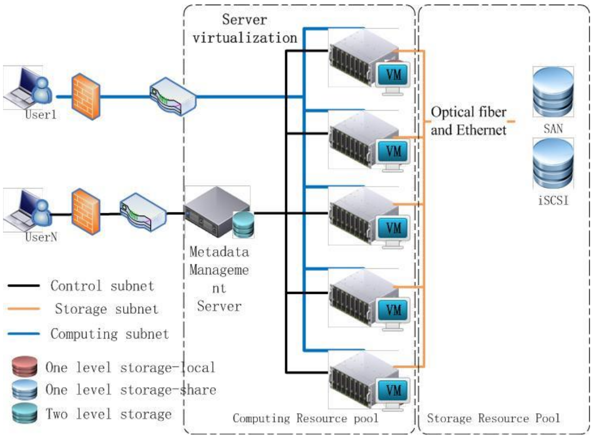


Figure 3. Centralized Architecture of Cloud System

Figure 2 shows the distributed system architecture, in which, virtual machines are placed on many computing nodes. Therefore, a computing resource pool is constituted, and represented and described using a computing subnet in the layered queuing task networks. These peer-to-peer independent computing nodes are connected via the computing subnet. Meanwhile, all computing nodes possess their own incidental storage devices which are organized in a sharing form. The data are shared through the storage subnet which is used for the indication and description in the layered queuing task Petri net. It is noteworthy that the storage and computing subnets are identical physically. Consequently, the competition of systems on network resources is induced, and expressed and illustrated using the network subnet in the layered queuing task Petri net. Network resources may be the bottlenecks controlling the evaluation index such as performance and expansibility of distributed cloud system under high loads. Therefore, the competitive characteristics on network resources need to be reflected in subsequently layered queuing task Petri net.

Figure 3 illustrates the centralized system architecture. As shown in the figure, virtual machines are laid on multiple computing nodes and therefore a computing resource pool is formed, which is denoted and expressed using a computing subnet in the layered queuing task networks. There are one or several server nodes of metadata mainly providing metadata service on computing, storage, *etc*. in a centralized form. Metadata servers and computing nodes are connected through computing subnet. Therefore, the cloud system is required to obtain metadata information on computing and storage of virtual machine through metadata service in advance of providing various services of computing and data operation subsequently. In addition, the centralized architecture of cloud system also provides centralized storage service. By using optical fiber or ordinary Ethernet, the computing subnet is connected on storage pools such as SAN and ISCSI. The interconnection of storage devices and interaction of data are fulfilled using the storage subnet inside the storage pools, for example, SAN, NAS, iSCSI and so on. Therefore, the storage and computing subnets are independent in the centralized system architecture physically, and this favors the concurrent processing of storage and computing information. They are expressed and depicted utilizing the storage and computing subnets in the layered queuing task networks. However, the metadata server, as the center of computing and storage, may be the performance bottleneck under high loads. Thus, it is required to reflect the independence of the subnets and the potential bottleneck problems of metadata server involving performance and expansion in subsequently layered queuing task Petri net.

The hybrid system architecture is plotted in Figure 4. As shown, a computing resource pool of cloud system is composed by placing virtual machines on many computing nodes. It is characterized and described using the computing subnet in the layered queuing task networks. The distributed, peer-to-peer independent computing nodes are connected through computing subnet. Meanwhile, each computing node possesses their own storage devices which preserve the data replica of current computing nodes rather than participating in the sharing among data. The sharing and transfer of data are accomplished through centralized storage service, and the computing subnet is connected on storage pools such as SAN and ISCSI using optical fiber or ordinary Ethernet. While, the interconnection of storage devices and interaction of data are fulfilled through the subnet of storage inside the storage pools (SAN, NAS, iSCSI, *etc*.). Physically, in the hybrid system architecture, the computing nodes of the computing subnet are distributed while the storage subnet is centralized. These results are expressed and demonstrated using computing subnet and storage subnet, respectively in the layered queuing task networks. Additionally, the independence between storage and computing subnets is conducive to the concurrent processing of storage and computing information. The subnets are described and illustrated via network subnet in the layered queuing task networks. However, the hit rates of data replica and data synchronization in the local storage device greatly affect the expansibility and performance of the whole system under the hybrid architecture. Therefore, the structural feature and aforementioned problems of hybrid architectures need to be focused on in subsequently layered queuing task Petri net.

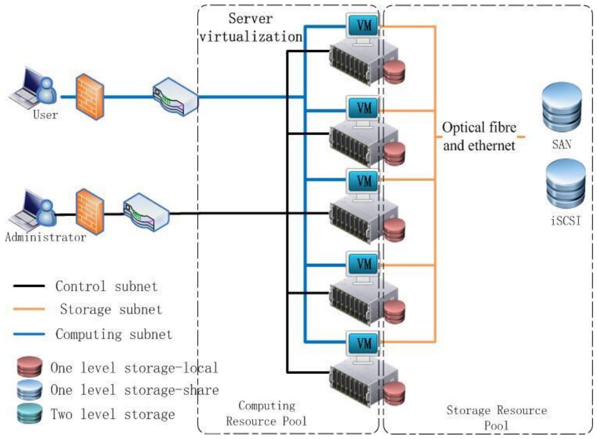


Figure 4. Hybrid Architecture of Cloud System

3. Task Scheduling of Cloud System

In the cloud computing task scheduling mechanism: From the user's point of view, each user can expect their task execution completed within the expected time; but from the perspective of task scheduling, the best result is to every task scheduler to pre.

The minimum completion time of the measured resource and this resource is minimal load. However, a resource may not be able meet these two requirements, so how do we find between the two key issues are an optimal balance to be addressed. This algorithm is the total time to complete the overall task load balancing system to shorten the point of view, by the following formula plane makes task scheduling between the two seek an optimal balance to finalize all the scheduling.

Based on the mentioned analysis on the characteristics of the alternative architecture scheme of China Cloud, the queuing task Petri net models of distributed centralized and hybrid architectures of cloud system are constructed. All architecture models of cloud system contain corresponding subnets of storage, network and computing and thus constitute the layered queuing task Petri net model of China Cloud system.

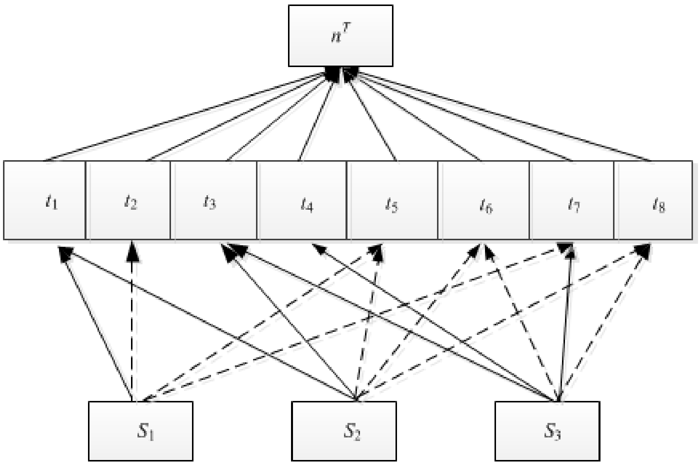


Figure 5. Task Scheduling Petri of Distributed Architecture of Cloud System

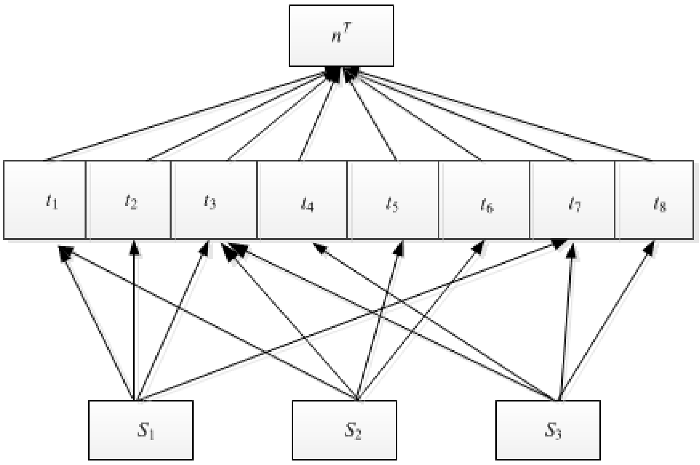


Figure 6. Task Scheduling Petri of Centralized Architecture of Cloud System

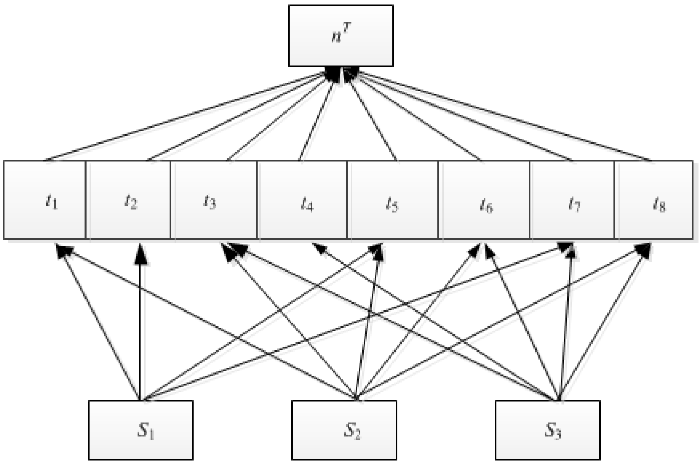


Figure 7. Task Scheduling Petri of Hybrid Architecture of Cloud System

The Petri net model of distributed cloud system and centralized cloud system are displayed in Figure 5 and Figure 6. Confined to the length of the paper, merely most important places and transitions are listed.

The main transitions in the model are expressed as follows. Tran 6 indicates that the tasks occur, enter into the virtual machine and conduct calculation. The trigger of this transition calls for corresponding support of computing resources, one of which is occupied by the tasks. Tran 3 illustrates that task data are not stored in local storage devices, triggers the transmission of the network data in next stage and obtains data from remote storage devices. This transition can be caused by corresponding network resources, one of which is occupied by the tasks. Meanwhile, this transition possesses weight of trigger. Tran 4 explains that task data are stored in the local storage devices and sources the operation of local data in next stage. The occurrence of this transition requires corresponding support of I/O resources, one of which is occupied by the tasks as well. This transition also shows the weight of trigger. Tran 10 demonstrates that the tasks reach remote storage devices and results in the operation of remote data in next stage. This transition can be induced by corresponding I/O resources, one of which is occupied by the tasks.

Resources are a common position and refer to the metadata server resource of the centralized cloud system. The resource quantity determines the performance of the whole system to a great extent owing to the centralized resources. Mqueue is a queue of the queuing task and describes the queue processing of the tasks on metadata server. It is required to firstly obtain corresponding virtual machine and data information through metadata server in the case of the arrival of new tasks. Snet_sn is a subnet of the queuing task and indicates that the task transmits relevant data in storage subnet in a network queuing task. Msresource is a common position and refers to the network resources of the storage subnet in the centralized cloud system. The application of a centralized storage results in the mutual independence of storage and computing subnets. Snet_s is a subnet of the queuing task, describes the queue of data processing of tasks in storage subsystem and calls for the support of relevant storage resources. Snet_cn is also a subnet of the queuing task and describes the computing operation of relevant tasks in network queuing task in a computing subnet. Snet_c is a subnet of the queuing task, depicts the task computing queuing task of tasks in computing subsystem and requires the support of relevant computing resources. Csource is a common position, which illustrates the computing resources in current computing subsystem. The resource quantity is dynamically adjusted during simulation process. The transitions are not given in detail because they are similar to those corresponding to Figure 5.

The Petri net model of hybrid architecture of cloud system is shown in Figure 7. The main places and transitions are similar to those in Figures 5 and 6 and are not described in detail as space is limited.

The Petri net model of hybrid architecture of cloud system in Figure 7 presents that the hit rate of local replica of computing nodes is essential for improving the performance of the whole system. High hit rate of local replica can save the consumption of the network resource in resource pools and thus improves the performance of data operation.

4. Evaluation and Simulation

Currently, there are many kinds of software and tools used for the automatic calculation of mathematical models at home and abroad. According to the aforementioned characteristics of queuing task Petri net, the QPME tool [12-13] was adopted to simulate the layered queuing task Petri net model of cloud system in this investigation. In this way, the concrete manifestations of evaluation indexes such as performance and expansibility under different system architectures and allocations were explored. Therefore, it offered helpful reference and advice for deploying and constructing cloud systems.

In order to evaluate the service characteristics and performances of China Cloud under different scales and loads, the load was further divided into 3 types by the model. X%：Y% refers to that the data of X% can be acquired through local storage system rather than the network in distributed or hybrid system architecture, while the data of Y% need to be read through the network. All data in centralized system architecture are obtained through the remote call of data network. Therefore, they are compute-intensive (Computing), data-intensive (Data), and calculation data balanced-type (Balance) architectures according to loads. The performances of the 3 types of cloud system architectures under different loads are displayed in Figures 8, 9 and 10, respectively.

Figure 8 presents the changes of the task throughput in the distributed cloud system under different loads. Simulation results showed that the throughput of the whole system increased with the increase of loads. Simulated data also presented that as the range of the working set continuously increased, the data directly obtained through local storage device decreased because more data needed to be acquired from remote storage device via network transmission. All these increased the overall postpone of the system and thus reduced the throughput. However, the network was the performance bottleneck of the whole cloud architecture and thus the throughput decreased rapidly as the data passing through the network constantly increased to approximate or exceed network process ability. As can be seen, the throughput of the state of 80%：20% decreased by 36.9% compared with that of the state of 20%：80%.

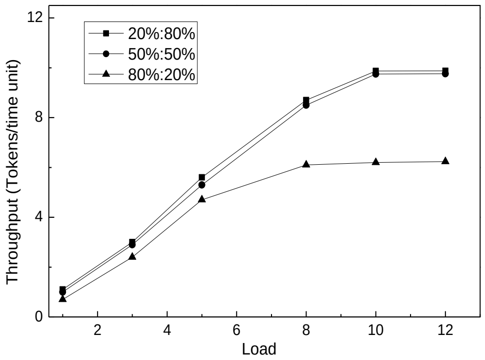


Figure 8. Different Ratio Throughput of Hybrid Architecture of Cloud System

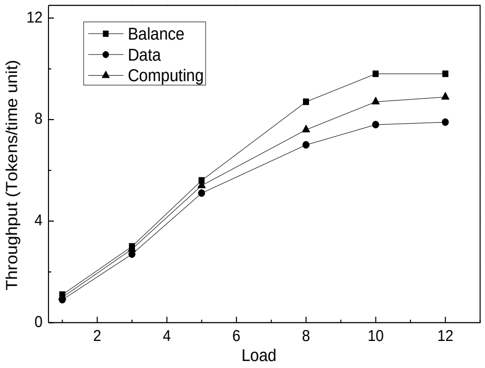


Figure 9. Different Parameters Throughput of Hybrid Architecture of Cloud System

Unlike the distributed architecture of cloud system, the simulation data in Figure 9 indicated that the variations of application types resulted in unbalanced resources of the cloud system. The subsystem with inadequate resources was likely to be the performance bottleneck of the whole system, and thus reduced the throughput of the system. As shown, the throughput of the system in balanced type was 9.803. The final throughput was merely 7.894 in data-intensive application owing to the great control of storage performance showing a decrease of 19.5% compared with the ideal balanced state. However, the cloud system presented an overall throughput of 8.88 which was between the former two ones under the compute-intensive application state. It was because that under this model, although the storage system was underutilized, the storage resources exerted more significant influence on the performance of the whole system than computing resources.

Figure 10 demonstrates the performance of hybrid architecture of cloud system. Simulation data showed that as the hit rate of local replica data decreased, the overall throughput of the system decreased gradually. However, the throughput was basically stable in case of the hit rate exceeding 8, during which, all data could not be acquired from local replica (0% ∶ 100%). Finally, in case of the hit rate of the replica data exceeding 20%, its increase insignificantly affected the increase of the overall throughput of the system.

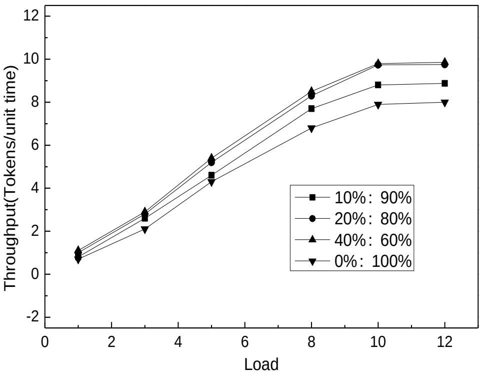


Figure 10. Throughput of Hybrid Architecture Cloud System

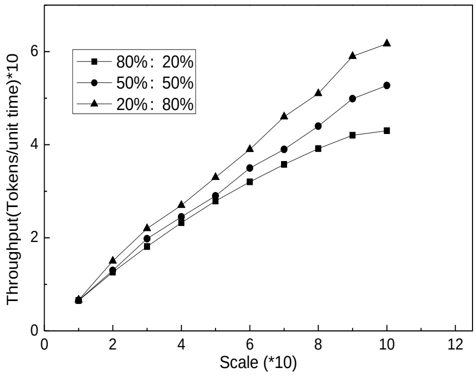


Figure 11. Throughput of the Distributed Architecture of Cloud System

The experimental results in Figure 11 showed that the expansibility of the distributed cloud system was controlled by network resources. With the increase of the system scale, the entire performance of the cloud system continuously increased. As the scale expanded from 10 to 40, the throughput of the system increased from 6.51 to 23.24 showing a 3.56 times of increase and a basically linear expansion in the case of adequate network resources. However, as the scale enlarged, the network resources gradually became scarce, especially, when the resources quantity needed by the system approximated or reached the amount provided by the network resources, the cloud system presented a sharply declined expansibility. Furthermore, the simulation data also indicated that the smaller the proportion of obtaining remote data via network, the better the expansibility of the system and the more insignificant control of network resources in the distributed architecture of cloud system. It could be observed that the distributed architecture of cloud system could more favorably adapt to compute-type application or the applications with intensively distributed working set of the data instead of those applications with large data and large range of working sets.

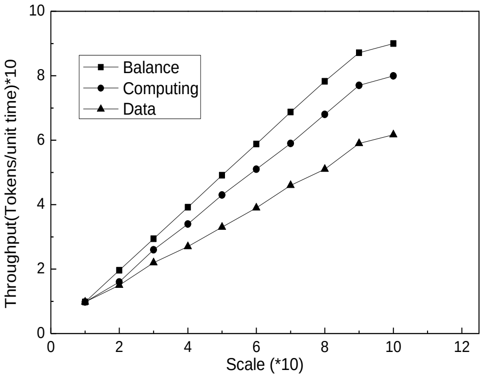


Figure 12. Throughput of the Centralized Architecture of Cloud System

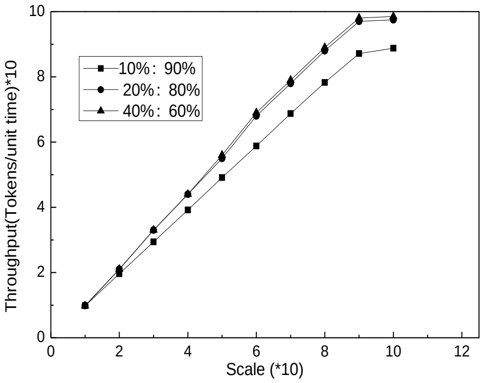


Figure 13. Throughput of the Hybrid Architecture of Cloud System

The results in Figure 12 presented that with the enlargement of the scale, the metadata server showed a scarce resource. The expansibility of the system decreased sharply in the case of the resource quantity required by the system approximating or reaching the ultimate service of the metadata server. The simulation data indicated that the main performance indexes such as throughput merely increased by 1.03 times as the scale of the system expanded from 90 to 100 with a 1.1 times of enlargement. However, as the scale increased by 2 times enlarging from 100 to 200, the throughput only increased by1.002 times owing to the control of metadata server. The balanced application type usually brought more preferable throughput and expansibility to the architecture in the centralized architecture of cloud system.

The experimental results in Figure 13 presented better expansibility of the system in the hybrid cloud architecture. Compared with the former two architectures, the hybrid cloud architecture possesses more favorable throughput and expansibility especially in the case of high hit rate of local replica data.

5. Conclusions

Because cloud computing systems composed by the tens of thousands of computing resources, making highly dynamic and heterogeneous characteristics, therefore, relatively traditional computing platforms, the higher the probability of error. Thus, to mention

QOS high cloud computing, cloud computing research task scheduler must have fault tolerance and the ability to tolerate failure strategy. It introduces the main task scheduling algorithm based on data from the local drive. The algorithm from a global point of view test considers scheduling, status of the network and system load data dynamically adjusted according to the task locality. In poor, under the network environment, the algorithm *TSCS* highlighted data locality, and in the case of system load overload algorithm *TSCS* local rate decrease data to reduce the completion time of the job. Experimental results show that the proposed algorithm can solve the problem of large-scale computing in the cloud.

Acknowledgments

Projects (U1304603,61503174) supported by the National Natural Science Foundation of China; Project (2014B520099) supported by Henan Province Education Department Natural Science Foundation; Project (142102210471) supported by Natural Science and Technology Research of Foundation Project of Henan Province Department of Science; Projects (1401037A) supported by Natural Science and Technology Research of Foundation Project of Luoyang Department;.

References

[1] M. A. Marsan, G. Conte and G. Balbo, A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems, ACM Transactions on Computer Systems, vol. 2, no. 2, **(1984**), pp. 93-122

[2] S. Kounev, R. Non and J. Torres, Autonomic QoS-aware resource management in grid computing using online performance models, Proc. of 2nd International Conference Performance Evaluation Methodologies and Tools (VALUETooLS-2007), Nantes, France, **(2007)** March16-18.

[3] S. M. Han, M. M. Hassan, C. W. Yoon and E. N. Huh, Efficient service recommendation system for cloud computing market, Proc. of 2nd International Conference on Interaction Sciences Information Technology, Culture and Human-ICIS 09, **(2009)** October 20-22.

[4] L. Wang, R. Ranjan and J. Chen, Cloud Computing: Methodology, System and Applications, CRC Press, **(2011)** January 10-12.

[5] S. Wu and X. Bai, Interview on Petri net analysis tool on Internet, Computer Science, vol. 23, no. 4, **(2011)** pp: 27-30.

[6] C. Lindeman, DSPNexpress: a software package for the efficient solution of deterministic and stochastic Petri Nets, Performance Evaluation, vol. 22,no.1 **(1995)** pp: 3-21.

[7] Hui, C., Xiaoyong, L., Shuling, D.: A dynamic load balancing algorithm for sort-first rendering clusters. In: 2nd IEEE International Conference on Computer Science and Information Technology (ICCSIT), IEEE, Beijing **(2009)** August 23-25.

[8] E. M. Thurner. TOMSPN: A tool for modeling with SPN, Proc. on computer System and Software Engineering, **(1995)** June 15-17.

[9] X. Zeng, H. Xiang and F. Huang. The design and implement of Petri net visibility tool, Journal of Huazhong University of Science and Technology (nature), vol. 30, no. 6, **(2013)**,pp. 43-45.

[10] J. Zhang, Z. Liu and H. Li, The study and development for Petri net graph modeling simulation system, Journal of Beijing University of Chemical Technology (Natural Science Edition), vol. 31, no. 2, **(2014)**,pp. 100-103

[11] X. Pan, T. Li and Q. Liu, A design and implement of Petri net hierarchy model modeling tool, Computer Application and Software, vol. 25, no. 8, **(2012)**, pp. 33-35.

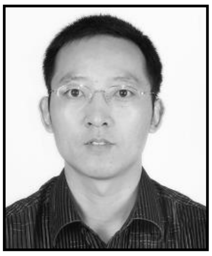
[12] S. Kounev and A. Buchmann, SimQPN - A tool and methodology for analyzing queuing task Petri net models by means of simulation, Performance Evaluation, vol. 63, no. 4-5, **(2006)** ,pp. 364-394.

[13] S. Spinner, S. Kounev and P. Meier, Stochastic modeling and analysis using QPME: Queuing task Petri net modeling environment v2.0, Proc. of 33rd International Conference on Application and Theory of Petri Nets and Concurrency, vol. 73, no. 47, **(2012)**, pp. 388-397

Authors

Zeyu SUN, was born in 1977 in Changchun city, Jilin province, in 2010 graduated from Lanzhou University, Master of Science; Xi'an Jiaotong University study for PhD at present. He is an associate professor in Luoyang institute of technology of computer and information engineering, is also a member of China computer society. The main research interest is in wireless sensor networks, parallel computing and Internet of things.

Yaping Li, was born in 1968 in Kaifeng city Henan province. In 1990 graduated from Zhengzhou University of Computer & Science Department, Master of Science; she is an associate professor in College of Computer and Information Engineering, Luoyang Institute of Science and Technology. The main research interest is wireless sensor network, parallel computing.



Cao Yangjie, born in 1976, Associate professor and PhD. member of China Computer Federation. His research interests include wireless sensor networks and internet of things.