A PGSA Based Data Replica Selection Scheme for Accessing Cloud Storage System

Bang Zhang, Xingwei Wang, and Min Huang

College of Information Science and Engineering, Northeastern University, Shenyang, 110819
Key Laboratory of Network Control Systems,
Chinese Academy of Sciences, Shenyang, 110016, China
cucumberbb@163.com,{wangxw,mhuang}@mail.neu.edu.cn

Abstract. The data replica management scheme is a critical component of cloud storage system. In order to enhance its scalability and reliability at the same time improve system response time, the multiple data replica scheme is adopted. When a cloud user issues an access request, a suitable replica should be selected to respond to it in order to shorten user access time and promote system load balance. In this paper, with network status, storage node load and historical information of replica selection considered comprehensively, a PGSA (Plant Growth Simulation Algorithm) based data replica selection scheme for cloud storage is proposed to improve average access time and replica utilization. The proposed scheme has been implemented based on CloudSim and performance evaluation has been done. Simulation results have shown that it is both feasible and effective with better performance than certain existent scheme.

Keywords: cloud storage, multiple data replica, replica selection, PGSA (Plant Growth Simulation Algorithm).

1 Introduction

Cloud storage is developed from cloud computing [1]. With the support of clustering, networking, virtualization, distributed operating system and distributed file system and others, it collects huge amount of heterogeneous storage devices in network and make them work cooperatively by application software, providing the unified storage system to enable users to store and access data transparently without the need to know where the data physically locate. The users only need to acquire online storage space through storage service provider without setting up their own data storage center, thus avoiding repeated construction of storage systems as well as saving the investment on hardware and software infrastructure.

Cloud storage is generally composed of massive storage devices distributed over different data centers. These devices are connected through network and have different reliability and performance with temporary even permanent faults happened occasionally. The network status also has influence on the timeliness and dependability of data

J. Wu et al. (Eds.): ACA 2014, CCIS 451, pp. 140–151, 2014.

access. In the face of huge amount of access requests from massive cloud users, deploying multiple replicas of one single data object in cloud storage system is generally needed in order to get timely, reliable and efficient data access services, and thus improve cloud storage system response time to user access and enhance system QoS (Quality of Service) and user QoE(Quality of Experience) [2-6]. A multiple data replica scheme can deploy right quantity of replicas for suitable data objects in due time and due place. It needs an efficient management mechanism to realize the suitable placement of multiple replicas, the suitable selection of the specific replica for user access and consistency maintenance among replicas, etc. In this paper, we discuss how to select the suitable data replica to respond to the user access request to the cloud storage system when multiple data replicas exist. In fact, the problem of data replica selection can be divided into two sub-problems: data replica location and suitable replica selection. The former refers to locate one or more replicas by the data's logical name. The latter refers to select the suitable one from multiple replicas in order to minimize, for example, the user access cost and thus help improve the overall system performance. In this paper, our research focus is put on the latter.

There are many factors which have major influences on the suitable replica selection, including network status, performance of the storage node where the data replica resides, and cloud user access cost to the replica, etc. A replica selection scheme is said to be good if it can shorten the cloud user access time to data as much as possible with the above factors taken into account thoroughly and comprehensively.

Some of the data replica selection mechanism has been proposed. In [7], an adaptive replica selection strategy was presented to employ network bandwidth information and account for its fluctuation in the wide-area environment, taking advantage of multiple replicas and concurrent data transfers. In [8], a two phased replica selection scheme was proposed. In the first coarse-grain phase, replicas with low latency (located at the uncongested network segments) and replicas with high latency (located at the congested network segments) were distinguished. In the second fine-grain phase, the replicas admissible for user access requirements were selected through applying a modified minimum cost and delay policy. In [9], a balanced QoS replica selection strategy was proposed to select a suitable replica which was the closest to the user with almost equal values of QoS parameters (such as availability, time and security).In [10], a workload-driven replica selection algorithm was proposed to minimize query latency in terms of the average query span, i.e., the average number of machines that were involved in processing of a query. In [11], based on the min-max balancing workload method, a dynamic replica selection strategy was proposed to upgrade the efficiency of execution in data grid environments. In [12], a new replica selection strategy was presented. It used the concept of association rules of data mining approach to the most stable network segments and could adapt its replica selection criteria dynamically so as to satisfy user access requirements and reduce system response time. In [13], an ant algorithm based replica selection scheme was proposed. The calculation formula of the probability of a replica being selected was devised, and the candidate replicas were predicted based on the historical replica accessed records. It could help balance the access loads among replicas dynamically. In [14], a hybrid replica selection strategy was proposed. It tried to combine the advantages of genetic algorithm and ant algorithm to find the suitable replica for the user access request. By speeding up the convergence to the optimum, the system response time to access was improved.

Inspired by the above research, a data replica selection scheme for accessing cloud storage system is proposed in this paper. It comprehensively considers network status, performance and load of the storage node where the replica resides, and historical information of replica selection. It tries to select the suitable data replica to respond to the cloud user access request based on the idea of PGSA (Plant Growth Simulation Algorithm) [15], so that the QoE of the user to use the cloud storage system can be improved in terms of replica utilization and average access time.

2 Problem Formulation

The suitable replica selection scheme is one of critical components of the data replica management scheme in cloud storage system. It selects the suitable replica from multiple candidates to respond to the cloud user access request. A good replica selection strategy should at first determine the related factors which have significant influence on the QoE of the user to access data, and then a suitable replica selection algorithm should be devised based on these factors. In this paper, the following major factors are considered when the suitable replica is selected to respond to the user request [13].

(1) Network status. It is an important factor to be considered when replica selection is done, because, for example, the available bandwidth affects the data transmission time, the end-to-end delay affects user access time, and the packet loss rate affects the data transfer reliability. Therefore, in the process of replica selection to respond to the cloud user request, the replicas which locate in those network segments in good conditions should be chosen preferentially. In addition, under the environment of cloud storage, the distance between the cloud user and the storage node where the replica resides should be considered when making the replica selection decision. It affects the data transfer latency and further influences the system response time. Thus, choosing the nearer replica to the user is more appropriate when responding to the access request.

(2) Storage node performance. It mainly refers to the node's data access speed, computing capability and networking capacity to the external network.

(3) Storage node access load. In general, the more heavy the access load of a storage node, the weaker its ability to respond to the cloud user access request, thus the replica which resides in the light-loaded storage node is preferred when selecting the suitable in replica.

(4) Historical information. The historical information of a specific replica being chosen to respond to the cloud user access request can reflect the user QoE on using this replica to certain degree and represent the measurement of the system on the effect of setting up this replica in a sense. The more the specific replica being selected to respond to the user requests, the more suitable and valuable it being setup. Thus, in this paper, the ratio of the times of a replica being chosen to respond to the user requests to the total times of its corresponding data object being accessed is also considered to be a reference to the process of suitable replica selection.

In summary, selecting suitable data replica can reduce the cloud user access latency and the network resource consumption, it can effectively achieve load balance among storage nodes where replicas reside at the same time improve the reliability of data access, etc. Therefore, how to select the suitable data replica from multiple candidates to respond to the cloud user access request is critical to the effective and efficient data replica management. The purpose of the proposed suitable data replica selection scheme in this paper is to achieve fast and reliable data access as well as efficient use of replicas with the above mentioned factors taken into account thoroughly.

3 Algorithm Design

PGSA is an intelligent optimization algorithm using the plant phototropism mechanism [15]. Based on the characteristics of the plant phototropism, it uses the plant growth environment as the problem solution space and determines the corresponding plant morphactin concentration according to the objective function value of the problem solution. The greater its difference from the root (i.e., the initial solution), the bigger the plant morphactin concentration, and thus the higher the plant growth chance. Therefore, it simulates the plant growth dynamic model to converge to the global optimum rapidly. In this paper, we use morphactin concentration to represent the data replica's fitness to the user access request, and select the suitable data replica by adjusting and comparing its morphactin concentration.

3.1 Calculation of Replica Morphactin Concentration

We refer to the basic idea of PGSA to devise the proposed suitable replica selection scheme. It mainly uses the morphactin concentration as the criteria to decide which replica being selected. When a new data replica R_j is set up in the cloud storage system, the Eqn. (1) is used to calculate the initial morphactin concentration of the replica.

$$\tau_j(0) = {}^{f_j}\!/_{r_j} \tag{1}$$

where the size of the R_j is denoted by f_j and the access speed of the storage node in which R_j resides is denoted by r_j.

After the replica setup, its morphactin concentration is adjusted when it being selected and accessed, and its calculation is defined in the following Eqn. (2):

$$\tau_j^{new} = \rho \tau_j^{old} + \Delta_{\tau_j} \tag{2}$$

where the proportion of the historical morphactin concentration information is denoted by ρ and the variation of the morphactin concentration is denoted by Δ_{τ_j}.

In fact, the morphactin concentration adjustment is involved in the following three different cases.

When a data replica has been selected to respond to the user request, its morphactin concentration will be reduced and the reduction is defined in the following Eqn. (3).

$$\Delta_{\tau_j} = -\left(f_j / bw_j + dl_j \right) \tag{3}$$

where the available bandwidth of the storage node in which R_j resides, is denoted by bw_j and the transfer latency between the storage node and the user is denoted by dl_j. The reason why the morphactin concentration is reduced is that load balance should be considered, once a replica was selected to respond to a user request, its probability of being selected to respond to other user requests should be decreased in order to prevent it being overloaded.

After a replica selected, if it is accessed successfully, its performance is considered good, thus as a kind of encouragement, its morphactin concentration is increased to make its probability of being selected to respond to user requests in the future higher, and the increase is defined in the following Eqn. (4).

$$\Delta_{\tau_j} = \vartheta_e \cdot \left(f_j / bw_j + dl_j \right) \tag{4}$$

where the reward factor is denoted by ϑ_e.

If the access to the selected replica failed, it means that the selected replica does not work normally at present, its morphactin concentration is decreased to make its probability of being selected to respond to user requests in the future lower, and the decrease is defined in the following Eqn. (5).

$$\Delta_{\tau_j} = -\vartheta_p \cdot \left(f_j / bw_j + dl_j \right) \tag{5}$$

where the penalty factor is denoted by ϑ_p.

3.2 Calculation of Replica Selection Probability

The change of a replica's morphactin concentration affects the probability of its being selected to respond to user requests. In this paper, the replica selection probability are determined by two parts: p_j and p'_j. The former is derived from the replica morphactin concentration and the node load, reflecting the influence of the status of the network and the node where the replica resides, i.e., the instant information on the selection decision. The latter is based on the historical information, reflecting the influence of the historical access frequency to the replica on the decision of whether it being selected in the future to respond to user requests.

In order to reflect the load status of the node where the replica resides, in this paper we divide the node load into 3 levels, i.e., light-loaded, moderate-loaded, and heavy-loaded. At the same time, a load factor is introduced to characterize the node load and defined as follows.

$$t(\lambda_j) = \begin{cases} e^{-b} & \lambda_j \in [0, \lambda_j^{\text{light}}] \\ e^{-b^2} & \lambda_j \in (\lambda_j^{\text{light}}, \lambda_j^{\text{heavy}}] \\ e^{-b^3} & \lambda_j \in (\lambda_j^{\text{heavy}}, \lambda_j^{\max}) \end{cases} \tag{6}$$

where λ_j indicates the current access volume of the storage node in which R_j resides, λ_j^{light} and λ_j^{heavy} are boundary values of light-loaded and heavy-loaded respectively, $\lambda_j^{\max}$ is the upper bound of the access volume which the storage node can accept, b is a constant greater than 1. The calculation of p_j is defined as follows.

$$p_j = {\tau_j + \alpha t(\lambda_j)} \Big/ {\sum_{j=1}^{k} (\tau_j + \alpha t(\lambda_j))} \tag{7}$$

where α is the adjustment coefficient determined by experience or experiment.

For $p_j^{'}$, it is set be the ratio of the times of a replica being chosen to respond to the user requests to the total times of its corresponding data object being accessed.

Taking p_j and $p_j^{'}$ into account comprehensively, the probability of a replica being selected is defined as follows.

$$p_j^{''} = \beta p_j + (1-\beta) p_j^{'} \tag{8}$$

where β is a weighting coefficient to reflect the relative importance of the instant information and historical information on replica selection decision.

The selection probability of each replica is calculated by Eqn. (8) and then the roulette method is used to select the specific replica. The selection process is described as follows.

Assume that there are k replicas of the specific data object in the cloud storage system, the selection probability of each replica is $p_j^{"}$ $(1 \le j \le k)$, then we have $ps(j) = p_0^{"} + p_1^{"} + \cdots + p_j^{"}$, $1 \le j \le k$. Generate a random number r between $r \in [0, ps(k)]$, if $ps(j-1) < r \le ps(j)$, then R_j is selected.

3.3 Algorithm Description

Based on the above discussion, the algorithm of the PGSA based suitable replica selection scheme is described as follows.

Step 1: Initialization: for each data object in the cloud storage system, generate k growing points corresponding to k replicas, and use Eqn. (1) to calculate the initial morphactin concentration of each growing point; input the user request set.

Step 2: Take one user request out of the input set and determine the specific growing points corresponding to the data object requested by the user.

Step 3: Calculate the selection probability of each replica corresponding to the growing point by the Eqn. (8), use the roulette method to select the matched growing point, i.e., the suitable replica to the specific user request.

Step 4: Do the regeneration process of growing point to update the corresponding morphactin concentration.

Step 4.1: Update the corresponding replica's morphactin concentration to the growing point selected in Step 3 by Eqn. (2).

Step 4.2: Access the selected replica as the response to the user request. If succeeded, use the Eqn. (4) to update the replica's morphactin concentration morpheme as encouragement; otherwise, use the Eqn. (5) to update it as punishment.

Step 5: Check whether the user request set becomes empty: if empty, the algorithm ends, otherwise go to Step 2.

4 Performance Evaluation

The simulation experiment of and performance evaluation on the proposed scheme in this paper have been done on the CloudSim [16]. The main functions of the proposed scheme, which have been implemented by simulation, are listed in Table. 1.

Table 1. Main functions of the proposed scheme

Name	Description
void get_ replica_info (…)	Get information about replica.
double cal_ morphactin (…)	Calculate morphactin concentration for a replica.
double cal_ probability (…)	Calculate probability of a replica being selected.
void roulette(…)	Use roulette method to select a replica.
double tuning (…)	Reduce a selected replica's morphactin concentration.
double reward (…)	Increase a successfully accessed replica's morphactin concentration.
double penalty(…)	Decrease an access failed replica's morphactin concentration.

In order to evaluate the performance of the PGSA based data replica selection scheme proposed in this paper (simply call it P scheme below), it is compared with the ant algorithm based one proposed in [13] (simply call it A scheme below) in terms of replica utilization and average access time.

4.1 Replica Utilization

Replica utilization (RU) is defined as the ratio of the response of a replica to the access requests to the corresponding data object. Apparently, the sum of RUs of all replicas of the same data object is 1. From Fig. 1, it can be seen that the RU distribution of P scheme is more balanced than that of A scheme. This is because when replica selection decision made, P scheme considers not only the performance of the storage node where the replica resides and the network status, but also the storage node load and the historical information of replica access, once the above factors changed, it will re-select the suitable replica; in addition, after a replica selected, its morphactin concentration will be adjusted and thus influence its chance to be selected in the future, promoting the selection balance among replicas. By contrast, A scheme does not take the above factors into account thoroughly, leading to a little poor RU distribution.

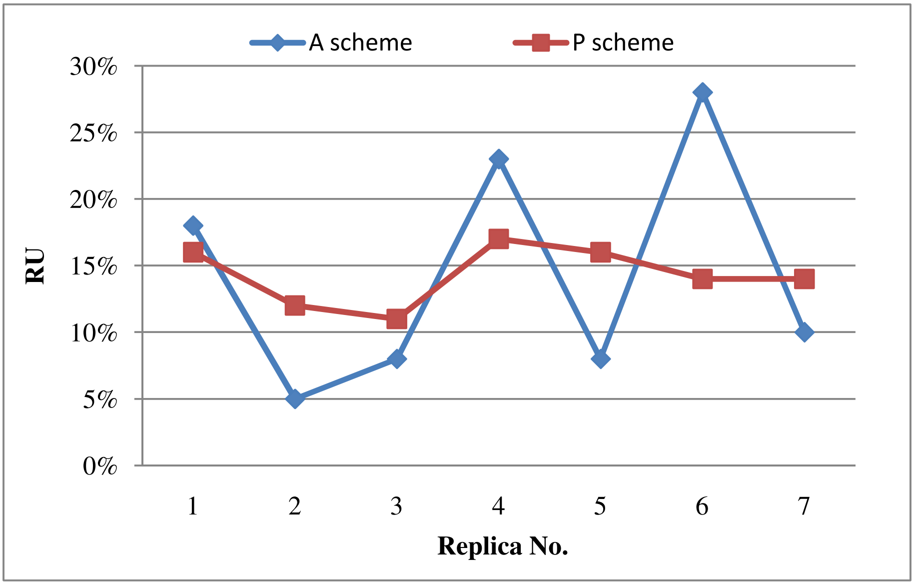
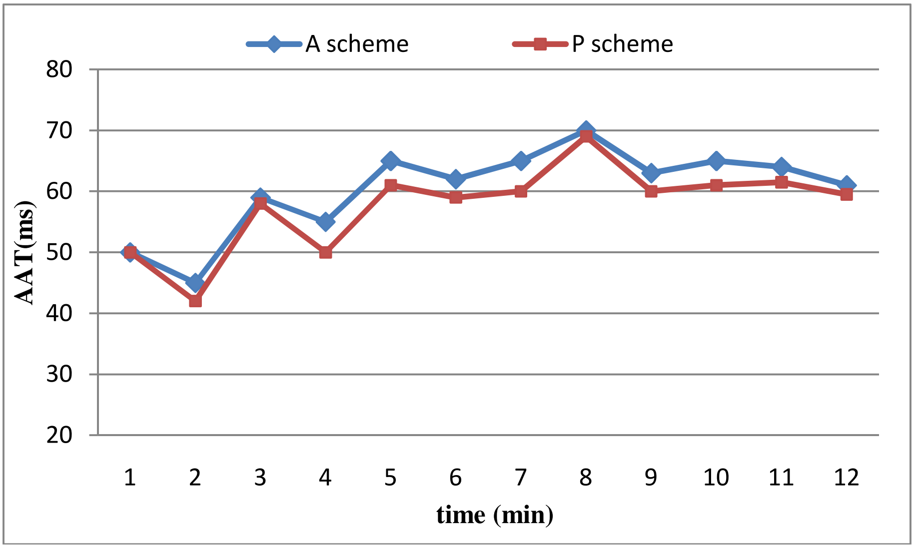


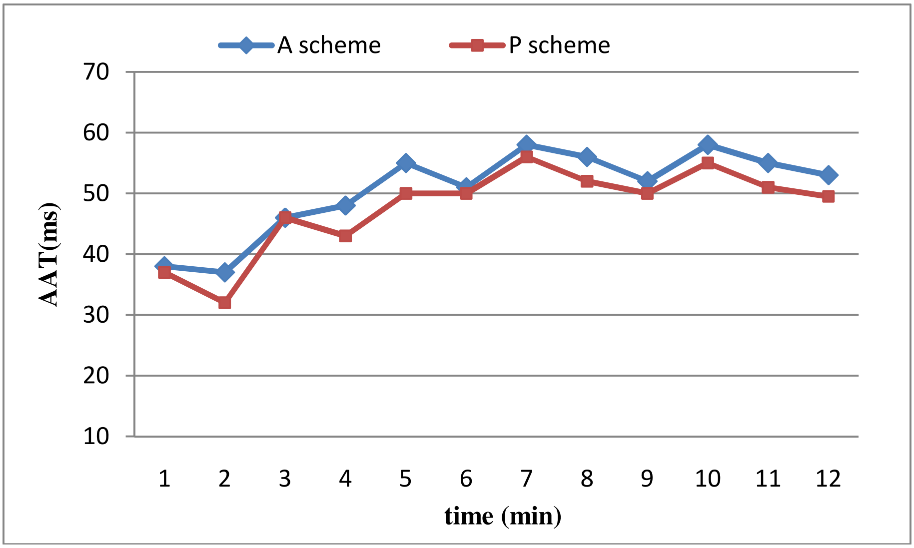
Fig. 1. Replica utilization distribution

4.2 Average Access Time

A user request's access time is defined as the time interval from it being issued to the end of the selected replica access, and the average access time (AAT) is defined as the average value of the access time of the simultaneously arrived user requests. From Fig. 2, it can be seen that the AAT of P scheme is better that of A scheme. This is because P scheme tries to achieve balance between the replica access time and the load of the storage node where the replica resides, thus it often selects the replica with smaller access cost and lower corresponding storage node load. Meanwhile, it uses roulette method to determine the selected replica by probability, this can further balance the access load among replicas, in the long run, it helps reduce AAT significantly. It also can be seen that, as the increase of the amount of replicas, P scheme has more chance to select the suitable replica from more replicas, each replica's access load is further reduced and becomes more balanced, thus AAT is reduced further. By contrast, A scheme is inflexible in reasonable apportion of replica access load and lacks dynamic adjustment ability, leading to the bigger AAT.



(a) AAT with seven replicas



(b) AAT with eleven replicas

Fig. 2. Average access time

5 Conclusions

The data replica management scheme is an important method to improve the efficiency of cloud storage system and a good data replica selection scheme is critical to improve its performance. It can not only help balance system load and shorten user access time, but also help improve replica utilization and make them work well. In this paper, based on the basic idea of PGSA, a data replica selection scheme for cloud storage is proposed with network status, performance and load of storage node and historical information of replica selection taken into account. It has been implemented based on CloudSim and performance evaluation has been done. Simulation results have shown that it has good performance in terms of replica utilization and average access time. In the near future, we will make the prototype implementation of the proposed scheme over the Northeastern University Campus Cloud platform in order to verify and enhance its practicability.

Acknowledgements. This work is supported by the National Science Foundation for Distinguished Young Scholars of China under Grant No. 61225012 and No. 71325002; the Specialized Research Fund of the Doctoral Program of Higher Education for the Priority Development Areas under Grant No. 20120042130003; the Fundamental Research Funds for the Central Universities under Grant No. N110204003 and No. N120104001.

References

1. Wang, X., Sun, J., Li, H., Wu, C., Huang, M.: A reverse auction based allocation mechanism in the cloud computing environment. Applied Mathematics & Information Sciences 7(1L), 75–84 (2013)
2. Lin, C., Hu, J., Kong, X.: Survey on models and evaluation of quality of experience. Chinese Journal of Computers 35(1), 1–15 (2012)
3. Wang, X., Cheng, H., Li, K., Li, J., Sun, J.: A cross-layer optimization based integrated routing and grooming algorithm for green multi-granularity transport networks. Journal of Parallel and Distributed Computing 73(6), 807–822 (2013)
4. Wang, X., Cheng, H., Huang, M.: Multi-robot navigation based QoS routing in self-organizing networks. Engineering Applications of Artificial Intelligence 26(1), 262–272 (2013)
5. Xie, X., Wang, X., Wen, Z., Huang, M.: A QoS routing protocol for cognitive networks. Chinese Journal of Computers 36(9), 1807–1815 (2013)
6. Qu, D., Wang, X., Huang, M.: An aware ant routing in mobile peer-to-peer networks. Chinese Journal of Computers 36(7), 1456–1464 (2013)
7. Gui, B., Yi, K.: A global dynamic scheduling with replica selection algorithm using GridFTP. Challenges in Environmental Science and Computer Engineering 1, 106–109 (2010)
8. Almuttairi, R.M., Wankar, R., Negi, A., Rao, C.R., Agarwal, A., Buyya, R.: A two phased service oriented Broker for replica selection in data grids. Future Generation Computer Systems 29(4), 953–972 (2013)
9. Jaradat, A., Amin, A.H.M., Zakaria, M.N.: Balanced QoS replica selection strategy to enhance data grid. In: International Conference on Networking and Information Technology, vol. 17, pp. 356–364 (2011)
10. Ashwin Kumar, K., Deshpande, A., Khuller, S.: Data placement and replica selection for improving co-location in distributed environments. Cornell University

11. Cheng, K., Wang, H., Wen, C., Lin, Y., Li, K., Wang, C.: Dynamic file replica location and selection strategy in data grids. In: IEEE International Conference on Ubi-Media Computing, pp. 484–489 (2008)
12. Almuttairi, R.M., Wankar, R., Negi, A., Chillarige, R.R., Almahna, M.S.: New replica selection technique for binding replica sites in data grids. Energy, Power and Control, 187–194 (2010)
13. Sun, M., Sun, J., Li, M., Yu, C.: Research of replica selection scheme based on ant algorithm in data grid. Computer Engineering and Application 43(1), 145–147 (2007)
14. Han, Z., Lin, Z., Wang, R.: A novel replica selection strategy based on combination algorithm in data grid. Journal of Nanjing University of Posts and Telecommunication (Natural Science) 31(5), 78–82 (2011)
15. Ding, X., You, J.: Plant growth simulation algorithm, pp. 1–59. Shanghai People's Publishing House, Shanghai (2011)
16. Calheiros, R.N., Ranjan, R., Beloglazov, A., et al.: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software: Practice and Experience 41(1), 23–50 (2011)