

Improving the selection of Service provider in Federated Cloud Architecture using adaptive clustering algorithm

Saisanthiya.D, M Saravanan, Ranjani.M

Email: saisanthiya.d@ktr.srmuniv.ac.in, saran84gct@gmail.com, ,ranjani.mo@ktr.srmuniv.ac.in

Information Technology, Assistant professor

SRM Institute of Science and Technology, Kattankulathur

ABSTRACT

Cloud is a trending technology now days. In this technology world now everyone are moving with the cloud. A cloud federation technique is one of the best ways to select any services from the service providers by the user at any time. As large number of service providers in the federation, user need to be select best service provider and the selection of service provider get delayed in the federation. We proposed the cluster based approach to select the best service provider for user request and reducing the selection time of service provider. In previous papers there is no Automated Selection for the selection of best service provider. Our proposed work is based on Automated Selection model i.e., Adaptive Clustering Algorithm (ACA) for grouping the similar services and categorizing the service providers into three groups. Such as Cluster-1, Cluster -2 and Cluster -3, after cluster these cluster sorted in order based on the final centroid of each cluster. Now user request to be compared with the middle cluster centroid and directed to respective cluster for service selection using Request Distribution Algorithm (RDA). If more than one service provider offers similar services in the cluster then Entropy based Priority Selection Algorithm (EPSA) to be invoked for tie break. This technique works better than existing algorithms and search time also minimized at selection of service.

Keywords: Adaptive Cluster Algorithm, Request Distribution Algorithm, Entropy, Information Gain

1. Introduction:

The cloud is having terms, attributes, and administrations related to web-based registering, so it has alluded as distributed computing. The cloud is having a few Characteristics, for example, framework, provisioning, organize get to, and oversaw metering are displayed. The business benefit models being sent, (for example, programming, stage, and foundation as an administration) and normal sending models utilized by specialist co-ops and clients to utilize and keep up the cloud administrations, (for example, the private, open, group, and crossover mists)

are talked about. We are having a few advantages and difficulties in distributed computing, and we are utilizing distributed computing administrations in various ways.

"Distributed computing" was instituted for what happens when applications and administrations are moved into the web "cloud." The cloud is only one access point from one point we can have N number of servers. Many organizations are conveying administrations from the cloud. Distributed computing isn't the simple thing which came at one overnight. The individuals endured such a great amount to execute this model

Characteristics

Cloud computing has a variety of characteristics, with the main ones being:

- 1) **On-demand self-service:** Customer might singularly plan preparing abilities, to example, server time and framework stockpiling as needed regularly without requiring human correspondence for master center.
- 2) **Broad network access:** Abilities are receptive through framework and got should through standard part that progresses utilization by heterogeneous their alternately thick client projects and likewise other traditional alternately cloud-based modifying administrations.
- 3) **Resource pooling:** Provides for figuring holdings would pooled to serve various clients using a multi-inhabitant demonstrate, with Different physical and virtual possessions progressively delegated Also reassigned Eventually by perusing purchaser request.
- 4) **Rapid elasticity:** Abilities can be quickly and flexibly provisioned, naturally scaled out and quickly discharged to rapidly scale in. Capabilities accessible for provisioning seem, by all accounts, to be boundless and bought in any amount whenever
- 5) **Measured services:** Cloud frameworks consequently control and streamline asset use by utilizing metering ability at some level of reflection proper to the kind of administration. Asset use can be checked, controlled, revealed. giving straightforwardness to the supplier and buyer benefit

To put it plainly, distributed computing takes into consideration the sharing and versatile arrangement of administrations, as required, from any area, and for which the client can be charged in light of genuine use.

2. Related Work:

Unified cloud computing surrounding named Inter-Cloud help the scaling from claiming requisitions crosswise over different merchant cloud [13]. The thought behind their acquainted league idea might have been on upgrade cloud suppliers provisioning competencies in the event of sudden passing spikes clinched alongside workload by renting accessible computational and stockpiling abilities starting with other administration suppliers [6], Inclination offers Inclination based Choice for cloud administration suppliers examined by the writers [5,8], it employments an comprehensive multi-criteria choice making methodology [14,15] for putting forth streamlining as an firm administration. To fuzzy AHP methodology used to tackle issue from claiming administration ranking [3] and permits those multi objective evaluation of cloud benefits. This methodology gives additional expressive also unified way to catch and transform

client feelings and inclination over universal administration positioning systems [4,16]. Cloud Broker Manager (CBM) doled out for consume obligation for asset provisioning and positioning [7].

Recommended an arrangement that calculates the necessity weights for each personal satisfaction quality as stated by the Quality of Service(QoS) inclination of a client effectively and reliably and reflects those weights to selecting those web service. Will figure those weights, apply a pairwise comparison matrix and a eigenvector of the matrix. Collective web administration QoS prediction [18] by means of neighborhood coordinated matrix factorization' Quality-of-service (QoS) prediction methodology for web services [17,21] suggested a schema and an instrument that measure those quality also prioritize cloud administrations. The outranking methodology [19] may be in light of the guideline of the review from claiming you quit offering on that one alternative's predominance over another, as opposed acknowledging that an single best elective could be distinguished [20] connected the essential standards done understanding for the fuzzy set hypothesis also modelled those choice making issue with the fuzzy different Criteria choice making (FMCDM) model.

3. Proposed System:

Our proposed work is based on Automated Selection model i.e., applied ACA algorithm for grouping the similar services and categorizing the service providers into three groups. Such as Cluster-1, Cluster -2 and Cluster -3, after cluster these cluster sorted in order based on the final centroid of each cluster. Now user request to be compared with the middle cluster centroid and the request greater than the middle centroid value, then it to be directed to cluster -1. If lesser than the middle cluster centroid, then it to be directed to cluster-3. At last if approximately equal to middle cluster centroid, then request to be processed by cluster – 2. If more than one service provider offers similar services in the cluster then priority decision tree to be invoked for selection of best among the similar group. In this work consider centroid as user requirements and we will take the nearest values which satisfies the user requirements. All these ranked services are provided through federated cloud. Service models in cloud categorized as follows Infrastructure as a service (like Amazon/ Rackspace), SaaS, platform as service PaaS (like Salesforce.Com, Windows Azure).

4. Broker Based Federate Cloud Architecture Model

Broker Manager (BM) entrenched integration between the services providers and the users that also extends safety and integrity in the proposed architecture. Based on the trust level estimation service provider and their reputation can be analyzed, through that we need to calculate trust between users and service providers. As a consequence it is compulsory to calculate the important of the trust, based on the service provider system with the related attribute. And make decision based on the aggregation of different types of trust at the user side the level of performance guarantee by the provider by using the SLA. Monitoring and validation are the significance process to manage trust in service after establishing. Between broker manager different types of trust mechanisms are suggested. And a set of brokers to provide the reliability and security in federated cloud architecture. In this architecture trust engine is proposed and combined to process and maintaining the trust information related to the user submission task. To describe the reliable criteria for the selection process of service provider,

SLA validation based trust model is incorporated. Broker trust level can be identified and classified into four levels such as fully trusted, moderately trusted, less trusted and not trusted.

In the proposed framework, Broker Manager (BM) is answerable for resource offering and each cloud provider has integrated with broker. For Secure communication between broker and Broker Manager we established a trust mechanism based on Broker Policy[1]. By using the Service Measurement Attributes SMI [8], application attributes the trust of the broker is validated, and this is used to prove level of commitment SLA extended by the provider. It will be recommended that SLA built trust will be actualized between requisition and administration supplier for check those proficient fruition of errand over unified cloud framework.

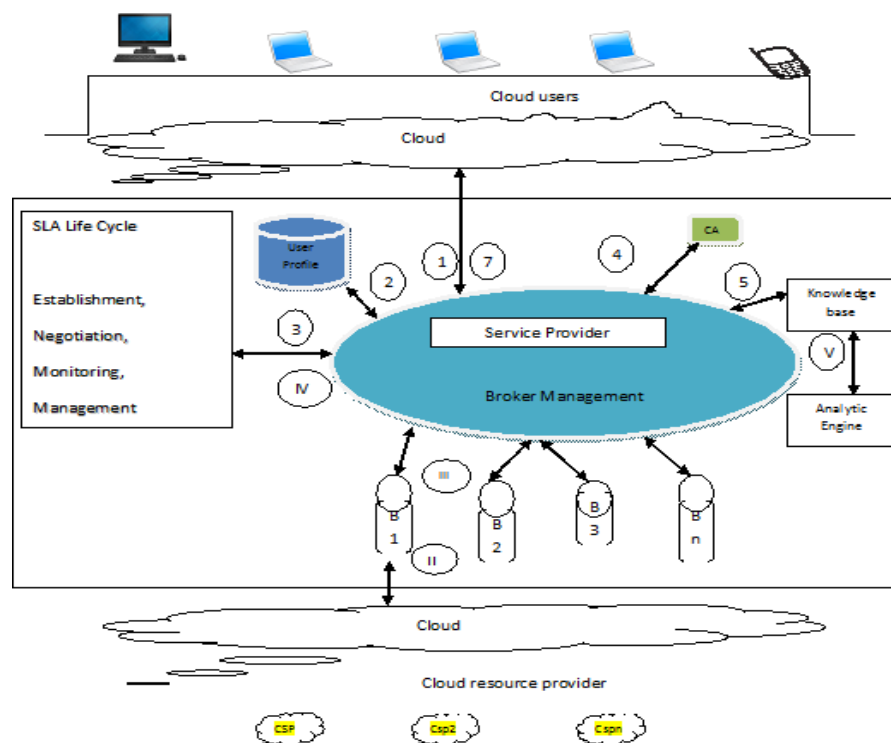


Figure 1 Broker based Federated Cloud Framework

Those intermediary based unified cloud architecture[2] is demonstrated in the figure 1 gives large amount integration, interoperability Also offering from claiming cloud data "around suppliers and clients. The fundamental segments for this structural engineering need aid separated module, Broker Manager Interface (BMI), Brokers (B), SLA life cycle Management and Certificate Authority, Analytic Engine, Knowledge Base (KB), Cloud Service Users (CSU)

and Cloud Service Providers (CSP). The approaching solicitations are arranged under whichever SLA cloud part alternately Non-SLA cloud part.

SLA part solicitations are sent with auxiliary BM while Non-SLA parts would moved on elementary BM. BM communicates for those brokers, collects the general data's over those cloud administration suppliers (CSPs) What's more redesign the majority of the data in the registry. Whether there may be whatever overhaul in the status of the brokers, that point it will be intimated of the BM and the KB may be updated instantly. In light of the registry data in the KB, those suppliers need aid short recorded utilizing choice tree positioning algorithm and select the ideal supplier to the cloud administration.

Broker service registry collects those particular trust qualities of the supplier registered utilizing Trust Engine (TE) and more updated those KB occasionally. Those upgrading of trust esteem conveyed out best there will be a deviation clinched alongside it but, the screening exercises performed about it.

4.1 Steps to Cluster

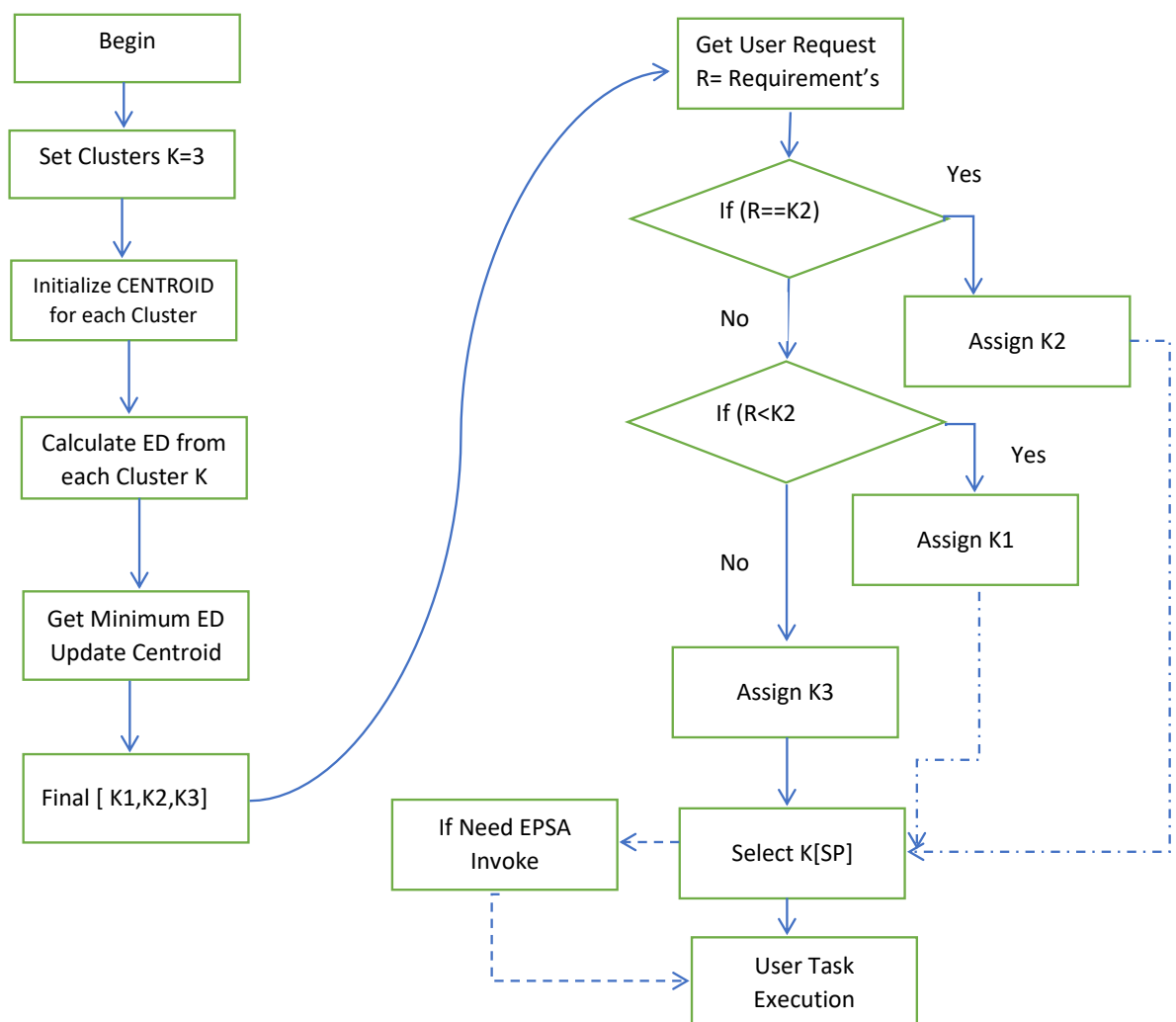


Figure 1 : Procedure for Cluster

Algorithm 1: Adaptive Cluster Algorithm(ACA)**Step 0: Begin****Step 1:** Number of service providers 1,2,3,4.....N**Step 2:** Initialize number of clusters $k = 3$.**Step 3:** Number of attributes to be considered for cluster 1,2,3,4A**Step 4:** Initialize centroid $C = C_1, C_2, C_3, \dots, C_k$ **Step 5:** Calculate the Euclidian Distance(ED) for the Selected Attributes.For $K = 1$ to 3

$$ED[K] = \sqrt{\sum (K[A_i] - SP1[A_i])^2}$$

Step 6 : Compare and Find minimum distance from the Cluster K_1, K_2 , and K_3 **Step 7 :** Group to the Respective Cluster**Step 8 :** Update the centroid of cluster for new centroid.

$$K[C] = (K[A_{1..N}] + SP[A_{1..N}]) / N$$

$$K[C_{new}] \leftarrow K[C_{old}]$$

Step 9 : Final Clusters $K[C_1, C_2, C_3]$ **Step 10 :** End Procedure**Algorithm2 : Request Distribution Algorithm (RDA)****Step 0 : Begin****Step 1:** Get User Request as Attributes $R \rightarrow R\{A_1, A_2, \dots, A_N\}$ **Step 2:** Compare $R[A]$ with $K[C_1$ to $C_3]$ **Step 3 :** If $R[A] < K[C_2]$ Assign Cluster K_1 ;**Step 4 :** If $R[A] > K[C_2]$ Assign Cluster K_3 ;**Step 5 :** If $R[A] \approx K[C_2]$ Assign Cluster K_2 ;**Step 6 : If** More than one SP in K_1 or K_2 or K_3 provides similar services**Then :** Invoke Entropy based Priority Selection Algorithm(EPsA)**Step 7 :** Assign $K[SP]$;**Step 8 : End**

4.2 Entropy based Priority Selection Algorithm (EPSA)

EPSA used in choice prediction algorithm, it helps to calculate the information gain. Entropy calculated for the selected SMI attributes and using these entropy Information Gain(IG) calculated for each attribute, highest information gain attribute decides the priority in the selection of service provider. It infers its classes starting with an altered situated from claiming preparation instances. Those classes made Toward EPSA need aid inductive, that is, provided for a little situated from claiming preparation instances, those particular classes made by EPSA need aid required will worth of effort for constantly on future instances. Those circulation of the unknowns must a chance to be the same concerning illustration those test situations. Incitement classes can't make turned out will energizes each instance since they might arrange a limitless number for instances.

Entropy based Priority Selection Algorithm (A, P, T)

Step 1 : Set as Input variables $A[i]$, $i=1,2,3,\dots,N$

Step 2 : Choose training set $T = \{t_1, t_2, t_3, \dots, t_n\}$

Step 3 : Output of each trained data set is $P = \{p_1, p_2, p_3, p_4, \dots, p_n\}$

Step 4: When all fields in T set then the same value for P

return a single node with that value

Step 5 : Compute information gain for each attribute in A relative to T

Step 6 : Select attribute with the largest Information gain as G

Step 7 : Let $\{G_1, G_2, G_3, \dots, G_n\}$ be the values of attribute G

Step 8 : Partition T into $\{t_1, t_2, t_3, \dots, t_n\}$ based on to the values of G

Step 9 : Return a tree with root G and branches labeled in the sub tree.

Step 10: Assign the G as Priority Attribute (Node)

$A \rightarrow$ Input Attributes,

$P \rightarrow$ Output of the class label(End results)

$T \rightarrow$ Training Data set

$G \rightarrow$ Information Gain Set

5. Performance analysis and Results Discussion:

The recommended ranking component will be executed clinched alongside CloudSim [20] utilizing java examined by Rodrigo et al. (2010). In view of this, Recreation of the recommended worth of effort will be conveyed crazy under CloudSim [21] utilizing JAVA, it evaluates the suggested worth of effort In view of the SMI measurements for example, such that Cost, flexibility, adaptability and also utilizing other measurements for example, such that availability, processor speed, capacity, reaction time, transforming memory like essential memory,back up, asset provisioning kind What's more stockpiling types[12]. It likewise figures that those chose supplier Eventually Tom's perusing those suggested method may be successful alternately not utilizing SMI measurements. We considered the sample cost for 10 services and it is grouped into three clusters. User request may be any other attribute that service cost compared with the centroid of the middle cluster and directed to specified cluster.

Table 1 Cost vs Service Providers

SP	Max. cost	Cluster K
1	\$3.20	K3
2	\$1.24	K1
3	\$3.04	K3
4	\$2.23	K2
5	\$0.15	K1
6	\$2.45	K2
7	\$3.00	K3
8	\$2.50	K2
9	\$1.20	K1
10	\$0.40	K1

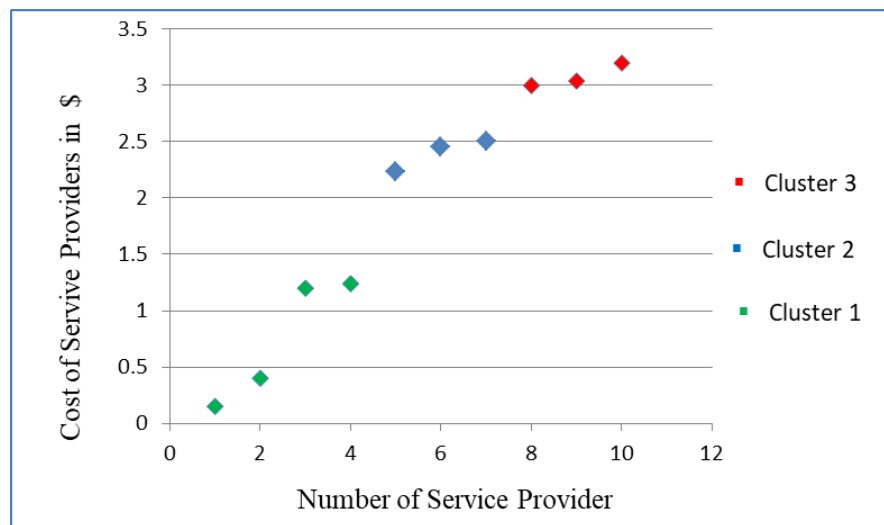


Figure 2: Cost based Cluster

Conclusion:

The architecture produces effective result when number of SMI attributes minimum. Clustering of CSPs becomes complicated when the numbers of parameters more. Architecture has no provision for emergency service request of non-SLA members may require at certain times. This architecture concentrating on prediction of service request, success rate of with a selection of service provider. Prediction mechanism helps to minimize the failure rate of any request in future.

References:

1. Saravanan, M., Aramudhan, M., Sundara Pandiyan, S., Avudaiappan, T.: Priority based prediction mechanism for ranking providers in federated cloud architecture. *Clust. Comput.* (2018). <https://doi.org/10.1007/s10586-017-1593-x>
2. European Grid Infrastructure, Federated Clouds Task Force.: Federated Clouds Task Force. <https://wiki.eui.eu/wiki/Fedcloudtf>. (2013). Accessed 20 Oct 2013
3. Garga, S.K., Versteeg, S., Buyyaa, R.: A framework for ranking of cloud computing services. *Future Gener. Comput. Syst.* **29**, 1012 - 1023 (2013)
4. Rajarajeswari, A.: Ranking model for SLA resource provisioning management. *Int. J. Cloud Appl. Comput.* **4**(3), 68–80 (2014) 4.
5. Patiniotakis, I., Verginadis, Y., Mentaz, G.: PuLSaR: preferencebased cloud service selection for cloud service brokers. *J. Internet Serv. Appl.* **6**, 26 (2015)
6. Celesti, A., Tusa, F., Villari, M., Puliafito, A.: How to enhance cloud architectures to enable cross-federation. In: *Proceedings of the 3rd International Conference on Cloud Computing (CLOUD 2010)*, pp. 337–345. IEEE, Miami, FL (2010)

7. Choi, C.R., Jeong, H.Y.: A broker-based quality evaluation system for service selection according to the QoS preferences of users. *Inform. Sci.* **277**, 553–566 (2014)
8. Key Performance Indicators.: <http://cloudtweaks.com/2012/03/kpis-for-cloud-service-providers-customers/>. Accessed 25 Jul 2015
9. Cloud Service Measurement Index Consortium (CSMIC), SMI framework.: <http://beta-www.cloudcommons.com/servicemeasurementindex>
10. Lee, W.-C.: Joint distribution of rank statistics considering the location and scale parameters and its power study. *J. Stat. Distrib. Appl.* **1**, 6 (2014)
11. Jrad, F., Tao, J., Streit, A.: SLA based service brokering in inter cloud environments. In: 2nd International Conference on Cloud Computing and Services Science (2012)
12. Buyya, R., Garg, S.: Calheiros: SLA oriented resource provisioning for cloud computing. In: International Conference on Cloud and Service Computing (2011). <https://doi.org/10.1109/csc.2011.6138522>
13. Buyya, B., Ranjan, R., Calheiros, R.N.: InterCloud: utility-oriented federation of cloud computing environments for scaling of application services. In: *Lecture Notes in Computer Science: Algorithms and Architectures for Parallel Processing*, vol. 6081, p. 20. Springer, New York (2010)
14. Dyer, M.: Multi Attribute Utility Theory, Multiple Criteria Decision Analysis: State of the Art Surveys. *International Series in Operations Research & Management Science*, pp. 265–292. Springer, New York (2005)
15. Figueira, J., Greco, S., Ehrgott, M.: Multiple Criteria Decision Analysis: State of the Art Surveys. *International Series in Operations Research & Management Science*, vol. 78. Springer, New York (2005)
16. Metsch, T., Edmonds, A., Bayon, V.: Using cloud standards for interoperability of cloud frameworks. *Technology Report*, pp. 1–13 (2010)
17. Zheng, Z., Ma, H., Lyu, M.R.: Collaborative web service QoS prediction via neighborhood integrated matrix factorization. *IEEE Trans. Serv. Comput.* **6**(3), 289–299 (2013)
18. Al-Masri, E., Mahmoud, Q.H.: QoS-based discovery and ranking of web services. In: *Proceedings of the 16th International Conference on Computer Communications and Networks*, Honolulu, Hawaii, USA, pp. 529–534 (2008)
19. Li, A., Yang, X., Kandula, S., Zhang, M.: CloudCmp: comparing public cloud providers. In: *Proceedings of the 10th Annual Conference on Internet Measurement*, Melbourne, Australia (2010)
20. Xiong, P., Fan, Y.: QoS-aware web service selection by a synthetic weight. In: *Fourth International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 632–637 (2007)
21. Calheiros, N., Ranjan, R., Beloglazov, A., De Rose, C.A., Buyya, R.: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Softw. Pract. Exp.* **41**(1), 23–50 (2010)

