Journal of Computer Science 9 (6): 671-677, 2013
ISSN: 1549-3636
© 2013 Science Publications
doi:10.3844/jcssp.2013.671.677 Published Online 9 (6) 2013 (http://www.thescipub.com/jcs.toc)

PERFORMANCE IMPROVEMENT IN CLOUD COMPUTING USING RESOURCE CLUSTERING

[1]Malathy, G., [2]Rm Somasundaram and [3]K. Duraiswamy

[1]Department of Computer Science and Engineering,
KSR Institute for Engineering and Technology, Thiruchengode, TamilNadu, India
[2]Department of Computer Science and Engineering, SNS College of Engineering, Coimbatore, TamilNadu, India
[3]Department of Computer Science and Engineering,
K.S. Rangasamy College of Technology, Thiruchengode, TamilNadu, India

Received 2012-10-05, Revised 2013-01-19; Accepted 2013-05-30

ABSTRACT

Cloud computing is a computing paradigm in which the various tasks are assigned to a combination of connections, software and services that can be accessed over the network. The computing resources and services can be efficiently delivered and utilized, making the vision of computing utility realizable. In various applications, execution of services with more number of tasks has to perform with minimum intertask communication. The applications are more likely to exhibit different patterns and levels and the distributed resources organize into various topologies for information and query dissemination. In a distributed system the resource discovery is a significant process for finding appropriate nodes. The earlier resource discovery mechanism in cloud system relies on the recent observations. In this study, resource usage distribution for a group of nodes with identical resource usages patterns are identified and kept as a cluster and is named as resource clustering approach. The resource clustering approach is modeled using CloudSim, a toolkit for modeling and simulating cloud computing environments and the evaluation improves the performance of the system in the usage of the resources. Results show that resource clusters are able to provide high accuracy for resource discovery.

Keywords: Aggregation, Clustering, Cloudsim, Cloud Computing, Resources, Tasks

1. INTRODUCTION

Large scale computing environments such as clouds propose to offer access to a vast collection of heterogeneous resources. A cloud is a parallel and distributed structure consisting of a group of interconnected and virtualized computers that are dynamically accessible as one or more unified computing resources (Buyya *et al*., 2009). The shared resources, software and information provided through the cloud to computers and other devices are normally offered as a metered service over the Internet. A user in the cloud system need not know about the place and other details of the computing infrastructure. Thus the user can comfortably concentrate on their tasks rather than utilizing time and knowledge on knowing the resources to manage the tasks. Internet is one basis of the cloud computing, therefore an unavoidable issue with Internet is that the network bottlenecks often occur when there is a large amount of data to be transferred. In this case, the complexity of resource management stick on to users and the users have normally limited management tools and authentication to deal with such issues (Armbrust *et al*., 2010). Clouds are classified into three categories named public clouds, private clouds and hybrid clouds (Sotomayor *et al*., 2009). Public clouds are publicly available remote interface for masses creating and managing resources, private clouds gives the local

Corresponding Author: Malathy G., Department of Computer Science and Engineering, KSR Institute for Engineering and Technology, Thiruchengode, TamilNadu, India

users a flexible and responsive private infrastructure to manage the workloads at their own cloud sites and the hybrid cloud enables the supplementing local system with the computing capacity from an external public cloud. Public cloud services like Google's App Engine are open to all anywhere round the clock. An example for private cloud is the usage of GFS, map reduce and big table by Google inside the enterprise. The following are the features of cloud computing:

- High flexibility
- High security
- Easy to maintain
- Location independent
- Reduction in capital expenditure on hardware and software

Figure 1 shows the four layer architecture of cloud computing and clouds are viewed as a large pool of computing and storage resources that are accessed through standard protocols with an abstract interface (Foster *et al*., 2008). Computing resources, storage resources and network resources are hardware level resources which are present in the fabric layer. Resources are virtualized to upper layer and end users as integrated resources are done in the unified resource layer. To develop and deploy platform on top of unified resources, a collection of specialized tools, middleware and services are added with the platform layer. The various applications that would be executed in the cloud environment are included in the application layer. The delivery mechanism in cloud computing is considered as services and is categorized in three different levels named; software service, platform service and infrastructure service (Armbrust *et al*., 2009). The Software as a Service (SaaS) is a software delivery model in that the applications are accessed by simple interface like web browser over Internet. Examples of SaaS-based services are web Mail, Google Docs, Facebook. The Platform as a Service (PaaS) gives a high-level integrated environment to build, test, deploy and host customer-created or acquired applications. Examples of PaaS-based service are Google App Engine, Engine Yard, Heroku. Infrastructure as a Service (IaaS) ensures processing, storage, networks and other fundamental computing resources to the users. Examples of IaaS-based services are Amazon EC2, IBM's Blue Cloud, Eucalyptus, Rackspace Cloud.

Clustering is a primary and cost-effective platform for executing parallel applications that computes large amount of data with the nodes of a cluster through the interconnected network.

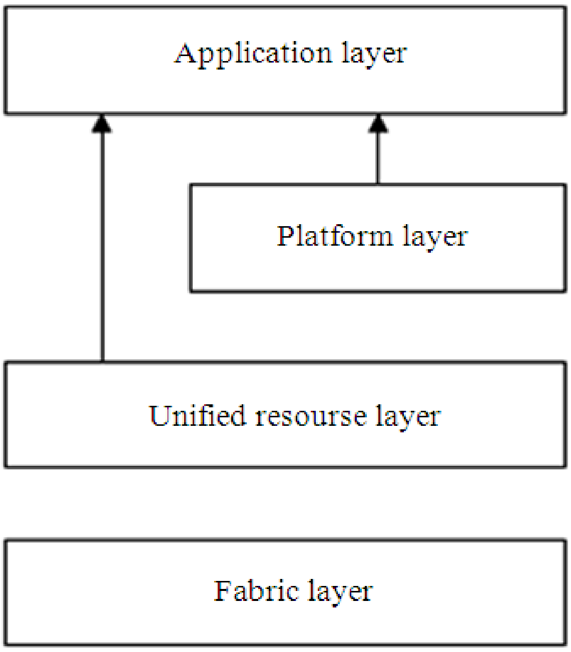


Fig. 1. Architecture of cloud computing

Clustering is traditionally been used in many data mining applications to group together the statistically similar data elements. The algorithms used for clustering must not assume the existence of a standard distribution of certain parameters. The performance of the cluster for scientific applications by the use of fully utilizing computing devices with idle or underutilized resources requires the scheduling and load balancing techniques in an effective way. Some of the applications of cluster based services include 3D perspective rendering technique, molecular dynamics simulation. Moreover, the performance between the effective speed of processor and the various network resources continues to grow faster, which raises the need for increasing the utilization of networks on clusters using various techniques (Kee *et al*., 2005). This study is organized as follows. It reviews about the related literature and focuses on the detailed description of the proposed fault identification in tasks using checkpoint based cloud computing approach. It also details the experimental setup and analysis of the proposed approach.

2. MATERIALS AND METHODS

The prior work on improving the design strategy in cloud computing is reviewed. Zhu and Agrawal (2011) proposed the use of cloud resources for a class of adaptive applications, where application-specific flexibility in computation is required with fixed time-limit and resource budget. The adaptive applications are maximized with Quality of Service (QoS) very precisely and by dynamically varying the adaptive parameters the value of application-specific benefit function is obtained. A multi-input multi-output feedback control model based dynamic resource provisioning algorithm is developed that adopts reinforcement learning to adjust adaptive parameters to guarantee the optimal application benefits within the time constraints.

Tayal (2011) proposed a task scheduling optimization for the cloud computing system based on Fuzzy-GA which makes a scheduling decision by evaluating the entire group of task in a job queue. The fuzzy sets were modeled to imprecise scheduling parameters and also to represent satisfaction grades of each objective. GA with various components is developed on the technique for task level scheduling in Hadoop MapReduce. To obtain better balanced load execution time of tasks assigned to processors are predicted using scheduler and making an optimal decision over the entire group of tasks.

Son and Sim (2012) proposed a service-level agreement while making reservations for cloud services. The presented multi-issue negotiation mechanism supports both price and time-slot negotiations between cloud agents and tradeoff between price and time-slot utilities. The agents make multiple proposals in a negotiation round to generate aggregated utility with variations in individual price and time-slot utilities.

Banerjee *et al*. (2009) proposed an initial heuristic algorithm to apply modified ant colony optimization approach for the diversified service allocation and scheduling mechanism in cloud computing framework. The proposed optimization technique is used to minimize the scheduling throughput to service all the diversified requests according to the different resource allocator available under cloud computing environment.

Wang *et al*. (2012) proposed a public cloud usage model for small-to-medium scale scientific communities to utilize elastic resources on a public cloud site. Also, implemented an innovative system named Dawning Cloud, at the core of which a lightweight service management layers running on top of a common management service framework. The system has been evaluated and found that Dawning Cloud saves the resource consumption to a maximum amount.

Buyya *et al*. (2011) presented the vision, challenges and architectural elements of service level agreement-oriented resource management. The architecture supports integration of market-based provisioning policies and virtualization technologies for flexible allocation of resources to applications. The performance results obtained from the working prototype system shows the feasibility and effectiveness of service level agreement-based resource provisioning in cloud systems.

Kim *et al*. (2007) it is proposed a decentralized algorithm for maintaining approximate global load information and a job pushing mechanism that uses the global information to push jobs towards underutilized portions of the system. The system effectively balances load and improves overall system throughput.

Belalem *et al*. (2011) proposed an algorithm to improve the quality of service of real world economy and to extend and enrich the simulator CloudSim by auction algorithms inherited from GridSim simulator. The work satisfies the users by reducing the cost of processing cloudlets and improved implementation on GridSim to reduce the time auction and to assure a rapid and effective acquisition of computing resources.

The stochastic models to mitigate the risk of poor QoS in computational markets is studied (Sandholm and Lai, 2007). The computational needs are typically expressed using performance levels; hence the worst-case bounds of price distributions to mitigate the risk of missing execution deadlines. The new proposal is a model-agnostic, distribution-free both in prices and prediction errors and does not require extensive sampling nor manual parameter tuning.

Hasham *et al*. (2011) proposed a pilot job concept that has intelligent data reuse and job execution strategies to minimize the scheduling, queuing, execution and data access latencies. By this approach, significant improvements in the overall turnaround time of a workflow can be achieved. This is evaluated using CMS Tier0 data processing workflow and then in a controlled environment. Gu *et al*. (2012) proposed a resource scheduling strategy based on genetic algorithm to produce best load balancing and reduces dynamic migration. To measure the overall load balancing effect of the algorithm an average load distance method is introduced. The method solves the problems of load imbalance and high migration cost after system virtual machine being scheduled.

A decentralized resource discovery service that is designed to satisfy queries over an extensible set of per-node and inter-node measurements that are relevant to deciding on which nodes of an infrastructure to place instances of distributed applications is introduced (Oppenheimer *et al*., 2004). The SWORD's operation on PlanetLab are scalable, distributed query processor for satisfying the multi-attribute range queries that describe application resource requirements and its ability to support queries over not just per-node characteristics such as load, but also over inter-node characteristics such as inter-node latency.

Erdil (2011) described a general purpose peer-to-peer simulation environment that allows a wide variety of parameters, protocols, strategies and policies to be varied and studied. For proof utilization of the simulation environment is presented in a large-scale distributed system problem that includes a core model and related mechanisms.

Kee *et al*. (2005) described an abstraction for providing resource specification, resource selection and effective binding in a complex Virtual Grid environment. The elements include a novel resource description language and a resource selection and binding component. The goal of the binding component is efficiency, scalability, robustness to high resource contention and the ability to produce results with quantifiable high quality.

The aggregated result for San Fermin node by swapping data with other nodes to dynamically create its own binomial tree (Cappos and Hartman, 2008). San Fermin node is a system for aggregating large amounts of data from the nodes of large-scale distributed systems. Because of the binomial tree the node are resilient to failures and ensures that the internal nodes of the tree have high capacity and reduction in completion time.

A compromised-time-cost scheduling algorithm which considers the characteristics of cloud computing to accommodate instance-intensive cost-constrained workflows by compromising execution time and cost with user input enabled on the fly. Simulations show that the algorithm can achieve a lower cost than others while meeting the user-designated deadline or reduce the mean execution time than others within the user-designated execution cost.

A technique for predicting availability and test them using traces taken from three distributed systems (Mickens and Noble, 2006). Also, described three applications of availability prediction. The first availability-guided replica placement, reduces object copying in a distributed data store while increasing data availability. The second shows how availability prediction can improve routing in delay-tolerant networks. The third combines availability prediction with virus modeling to improve forecasts of global infection dynamics.

2.1. Resource Identification and Clustering

In recent years the use of cloud systems for scientific computations, data sharing and dissemination are increasing in a rapid manner. Each cloudlet available in this distributed system will differ widely in resource capabilities such as CPU performance, bandwidth, memory availability and so on. So identifying the resource is needed to satisfy the application requirements. The traditional resource discovery systems use only the recent cloudlet usage information for scalability and simplicity. This s proposes a resource discovery approach based on resource usage information from cloudlets both in temporal and spatial manner. The temporal manner deals with the resource usage in a long-term pattern and spatial manner deals with the number of cloudlet with similar usage patterns. A resource cluster represents the resource usage distribution for a group of cloudlets with similar resource usages. Resource cluster requires two complementary techniques to capture the temporal details of the cloudlets. The two techniques are resource usage histogram and clustering-based resource aggregation. Histograms gives information about the statistical usage of resource capacities and the aggregation is used to achieve compact representation of a set of similar cloudlets for scalability. That is the resource clusters are used to find group of cloudless satisfying a common requirement. To handle the various resource requirement specifications the resource usage histogram is used and its associated observation is viewed at time period‘t’. The histogram shown in **Fig. 2** illustrates the resource capacity distributions in the cloudlets from observations over the past‘t’ units. Separate histogram is to be used for each resource types like CPU performance, bandwidth, memory availability and for each time granularities like hour, day, week, month, year.

The histogram representation gives the information about the percentiles for a particular resource capacity and also histogram helps us to preserve all usage data. That is the histogram can be used to capture different tolerances even if different applications specify different resource capacity requirements. Thus statistical information for each cloudlet would be represented by several histograms, corresponding to various resources and time scales.

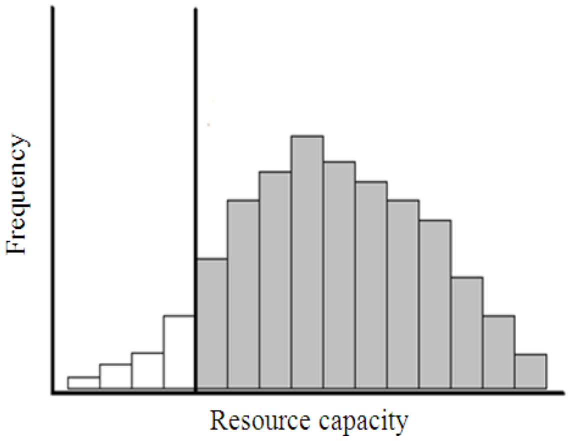


Fig. 2. Histogram representation

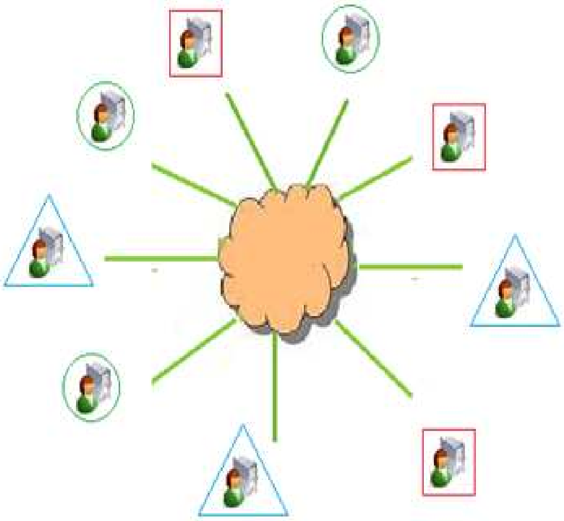


Fig. 3. Resources with similarities identified

But disseminating all the details over the network creates additional overheads for the network traffic making the system infeasible. Therefore, resource aggregation is done to preserve the overall information in the system and to achieve a desirable value between the quality of resource discovery and the overhead in transmission.

To classify the cloudlets with similar distributions and to preserve the information aggregation is provided through clustering algorithms. By the use of histogram cloudlets with similar features are identified, this is shown in **Fig. 3**. In the diagram three different resources are available in the cloud system.

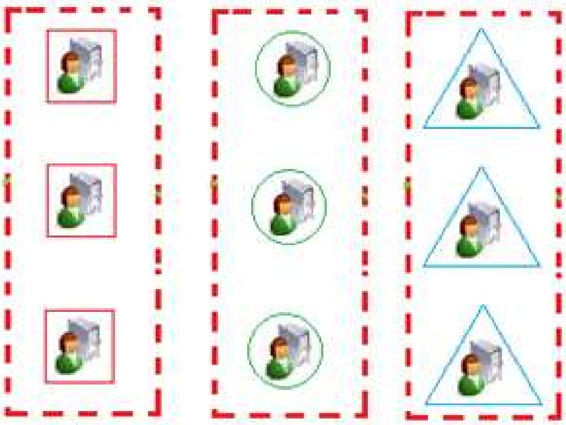


Fig. 4. Formation of resource clusters

Then by the use of clustering algorithm, resources having similar cloudlets are grouped as a resource cluster and are shown in **Fig. 4**. The clustering algorithm is capable to handle multi-element data in the cloud environment. For this multinomial model-based expectation maximization clustering algorithm (Zhong and Ghosh, 2003) is used. This algorithm will group the cloudlets together that have similar histogram distributions.

3. RESULTS AND DISCUSSION

3.1. Simulation Results

The simulation describes the implementation method for the proposed resource clustering in cloud computing system. For the simulation hundred cloudlets and five clusters are considered. Implementation is carried out on Cloudsim, because the rich set of simulation facilities in Cloudsim empowers us to implement and evaluate the histogram approach for identifying the similar cloudlets. **Figure 5 and 6** shows the implementation screenshot of tasks executed with the resources in the cloud system using CloudSim.

In resource clustering the choice of an appropriate value for the number of clusters has to be viewed seriously. If the number of cluster is too small, the resource cluster representatives will poorly represent their constituent cloudlets due to high heterogeneity. If the number of cluster is too large, then the technique for aggregation will not be effective due to the large overhead in the transmission. The resource usage gives the average amount of usage of resources in the cloud system.

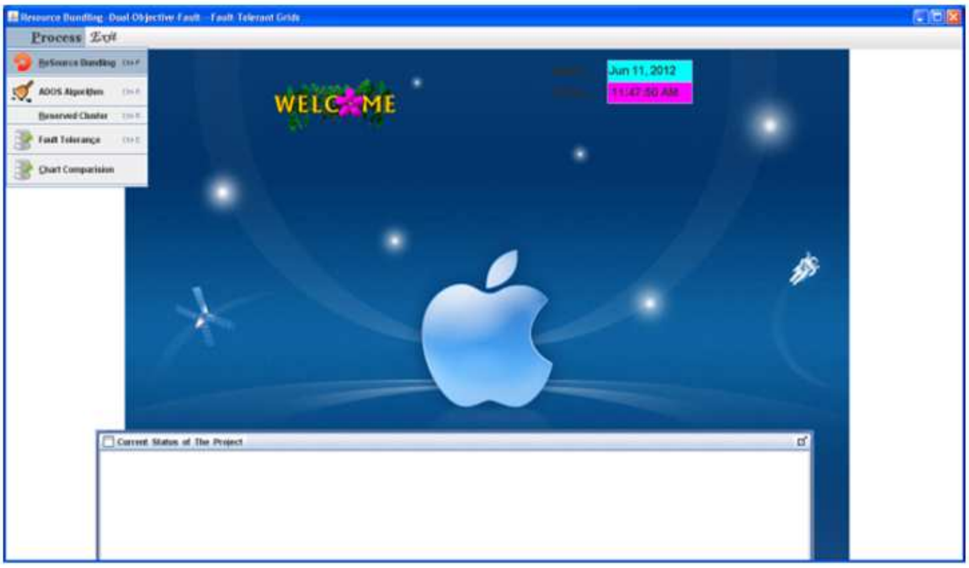
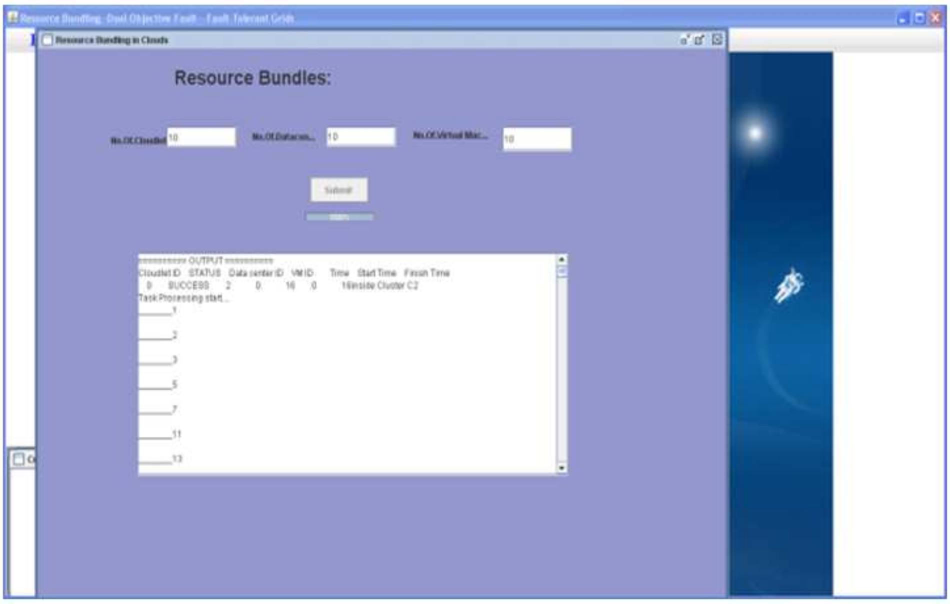


Fig. 5. Implementation screenshot using CloudSim


Resource Bundling -Dual Objective Fault - Fault Tolerant Grids
Resource Bundling in Clouds
Resource Bundles:
Submit

No.Of.Cloudlet 10 No.Of.Datacen... 10 No.Of.Virtual Mac... 10

```
=========== OUTPUT ===========
Cloudlet ID   STATUS   Data center ID   VM ID    Time   Start Time   Finish Time
  0     SUCCESS     2        0       16     0       16inside Cluster C2
Task Processing start...
______1

______2

______3

______5

______7

______11

______13
```

Fig. 6. Resource clustering

4. CONCLUSION

This study addresses the problem of resource discovery in cloud computing system. The proposed resource clustering approach employs two complementary techniques; resource usage histograms to provide statistical assurance for resource capacities and clustering-based resource aggregation to achieve scalability. Analysis has been done using the CloudSim simulator, in which the maximum cloudlets used in the cloud system is 100 and number of resource clusters are limited to five. Thus the cloud system can be efficiently used and further enhancement can be carried out for the fault tolerant in the system.

5. REFERENCES

Armbrust, M., A. Fox and R. Griffith, 2009. Above the clouds: A Berkeley view of cloud computing. University of California, Berkeley.

Armbrust, M., A. Fox, R. Griffith, A.D. Joseph and R. Katz *et al*., 2010. A view of cloud computing. ACM Commun., 53: 50-58. DOI: 10.1145/1721654.1721672

Banerjee, S., I. Mukherjee and P.K. Mahanti, 2009. Cloud computing initiative using modified ant colony framework. World Acad. Sci. Eng. Technol., 56: 221-224.

Belalem, G., S. Bouamama and L. Sekhri, 2011. An Effective economic management of resources in cloud computing. J. Comput., 6: 404-411. DOI: 10.4304/jcp.6.3.404-411

Buyya, C.S., R. Yeo, S. Venugopal, J. Broberg and I. Brandic, 2009. Cloud computing and emerging IT platforms: Vision, hype and reality for delivering computing as the 5th utility. Future Generat. Comput. Syst., 25: 599-616. DOI: 10.1016/j.future.2008.12.001

Buyya, R., S.K. Garg and R.N. Calheiros, 2011. SLA-oriented resource provisioning for cloud computing: Challenges, architecture and solutions. Proceedings of the International Conference on Cloud and Service Computing, (CSC' 11), IEEE Computer Society Washington, USA., pp: 1-10. DOI: 10.1109/CSC.2011.6138522

Cappos, J. and J.H. Hartman, 2008. San fermín: Aggregating large data sets using a binomial swap forest. Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation, (NSDI'08), USENIX Association USA., pp: 148-160.

Erdil, D.C., 2011. Simulating peer-to-peer cloud resource scheduling. Peer-to-Peer Network Applic., 5: 219-230. DOI: 10.1007/s12083-011-0112-8

Foster, I., Y. Zhao, I. Raicu and S. Lu, 2008. Cloud computing and grid computing 360-degree compared. Proceedings of the Grid Computing Environments Workshop, Nov. 12-16, IEEE Xplore Press, Austin, TX., pp: 1-10. DOI: 10.1109/GCE.2008.4738445

Gu, J., J. Hu, T. Zhao and G. Sun, 2012. A new resource scheduling strategy based on genetic algorithm in cloud computing environment. J. Comput., 7: 42-52.

Hasham, K., A.D. Peris, A. Anjum, D. Evans and S. Gowdy *et al*., 2011. CMS workflow execution using intelligent job scheduling and data access strategies. IEEE Trans. Nucl. Sci., 58: 1221-1232.

Kee, Y.S., D. Logothetis, R. Huang, H. Casanova and A.A. Chien, 2005. Efficient resource description and high quality selection for virtual grids. Proceedings IEEE International Symposium of Cluster Computing and the Grids, (CCGrid' 2005), pp: 598-606.

Kim, J.S., P. Keleher, M. Marsh, B. Bhattacharjee and A. Sussman, 2007. Using content-addressable networks for load balancing in desktop grids. Proceedings of the 16th IEEE International Symposium on High Performance Distributed Computing, (HPDC' 07), ACM Press, New York, USA., pp: 189-198. DOI: 10.1145/1272366.1272391

Mickens, J.W. and B.D. Noble, 2006. Exploiting availability prediction in distributed systems. Proceedings of the 3rd conference on Networked Systems Design and Implementation, (NSDI'06), USENIX Association Berkeley, USA., pp: 6-6.

Oppenheimer, D., J. Albrecht, D. Patterson and A. Vahdat, 2004. Distributed resource discovery on planetlab with SWORD. Proceedings of the WORLDS, (WORLDS '04).

Sandholm, T. and K. Lai, 2007. A statistical approach to risk mitigation in computational markets. Proceedings of the 16th IEEE International Symposium on High Performance Distributed Computing, (HPDC' 07), ACM Press, New York, USA., pp: 85-95. DOI: 10.1145/1272366.1272378

Son, S. and K.M. Sim, 2012. A price- and-time-slot-negotiation mechanism for Cloud service reservations. IEEE Trans. Syst. Man. B Cybern., 42: 713-728. PMID: 22194246

Sotomayor, B., R.S. Montero, I.M. Llorenteband I. Foster, 2009. Virtual infrastructure management in private and hybrid clouds. IEEE Internet Comput., 13: 14-22. DOI: 10.1109/MIC.2009.119

Tayal, S., 2011. Tasks scheduling optimization for the cloud computing systems. Int. J. Adv. Eng. Sci. Technol., 5: 111-115.

Wang, L., J. Zhan, W. Shi and Y. Liang, 2012. In cloud, can scientific communities benefit from the economies of scale. IEEE Trans. Parallel Distribut. Syst., 23: 296-303.

Zhong, S. and J. Ghosh, 2003. A comparative study of generative models for document clustering. Proceedings of the SDM Workshop on Clustering High Dimensional Data and its Applications, (CHDDIA' 03).

Zhu, Q. and G. Agrawal, 2011. Resource provisioning with budget constraints for adaptive applications in cloud environments. IEEE Trans. Services Comput.