Circuits and Systems, 2016, 7, 3121-3130
Published Online August 2016 in SciRes. http://www.scirp.org/journal/cs
http://dx.doi.org/10.4236/cs.2016.710265

Residual Resource Defragmentation Based on ECRC (Enhanced Cloud Resource Consolidating)

G. Soniya Priyatharsini, N. Malarvizhi

Department of Computer Science and Engineering, Veltech Dr. RR & Dr. SR Technical University, Chennai, India
Email: sonigeorge85@gmail.com, hodit.jec@gmail.com

Received 6 May 2016; accepted 18 May 2016; published 19 August 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
http://creativecommons.org/licenses/by/4.0/
Open Access

Abstract

In cloud computing, server consolidation is the part where very few persons go through the same. By consolidating the unused server space, memory can be reused for another data allocation. The objective of this paper is to improve resource utilization. Residual resource fragmentation refers to the state of the data center where sufficient amount of residual resources are available for any new VM allocation. To achieve this, there are three methods followed here. Active physical servers are identified. Then the maximum utilization of the resources is found out. Finally the resources are allocated and scheduled using the developed algorithm. In this work, we have proposed a new algorithm enhanced cloud consolidating algorithm. This algorithm improves some of the qualities of the cloud consolidating algorithm. Here the allocation technique is based on the cost and the memory.

Keywords

Server Consolidation, Residual Resource Defragmentation, Server Sprawl, Resource Pooling

1. Introduction

Cloud computing is raised in its popularity nowadays. Data center providers, due to their competition, have started many new user applications and expanded in wide range. Such applications are catalogued in the data center's physical machines or virtual machines. Virtualization is the method of running multiple independent virtual operating systems on a single physical computer. This approach maximizes the return on investment for the computer. The management and the creation of the virtual machines have often been called platform virtualization.

How to cite this paper: Priyatharsini, G.S. and Malarvizhi, N. (2016) Residual Resource Defragmentation Based on ECRC (Enhanced Cloud Resource Consolidating). *Circuits and Systems*, **7**, 3121-3130. http://dx.doi.org/10.4236/cs.2016.710265

The software applications can be shared using the virtual machines with minimum sharing of the resources. These kinds of hardware isolation lead to the optimal resource utilization.

The cloud can offer customers the flexibility to specify the exact amount of applications, computing power or data they need to satisfy their business requirements. A big desire for their desire to use the cloud is the availability of collaborative services. Perhaps because of the great flexibility of open source, which facilitates the efforts of vendors, large commercial users, cloud implementers most of all, the successful applications of open source have evolved from within consortia.

1.1. Server Consolidation

The above described server consolidation is an accession to an effective usage of the resources of the cloud server. It reduces the total number of the servers/server locations. Here this identified problem server sprawl is responded by the server consolidation process. Server sprawl is a situation where underutilized servers take more space and consume more resources. **Figure 1** explains the server consolidation process.

1.2. Residual Resource Fragmentation

In the cloud computing, the allocation of the resource (RA) is the process of allocating the resources which are available for the needed cloud applications through the internet. If the allocation is not managed properly, the resource allocation declined the services. Resource provisioning clarifies that problem by allowing the cloud service providers to manage the resources for each individual phase. Resource allocations strategy integrates cloud provider activities for allocating resources within the cloud environment. Optimal resource allocation strategies should avoid the following pattern:

Resource Fragmentation: Situation appears when the resources are segregated.

Resource dissent: Situation appears when two applications try to use the same resource at the time.

Scarcity of resources: appears when there are limited resources.

Under Provisioning: It occurs when the application is allotted with fewer numbers of resources than the demand.

Over Provisioning: It appears when the application gets surplus resources than the demanded one.

Resource allocation strategy can be categorized as follows [1], Execution Policy-Matchmaking, VM-Load, cost, speed, type, Policy-Security, Processor, Utility-Response time, Profit, Application satisfaction, Gossip-Peer information, Peer resources, Expert knowledge, Auction-Market Bid, H/W resource dependency-Storage, CPU, I/O, Communication, SLA-Response time, throughput, QoS. Application-Large scale, Data intensive, real time, Shared DB. The resource allocation should satisfy the following parameters, a) throughput b) latency c) response time. Even the cloud service providers are reliable they offer a crucial problem in dynamic resource allocation. To overcome this, input parameters are needed as shown in following **Table 1**.

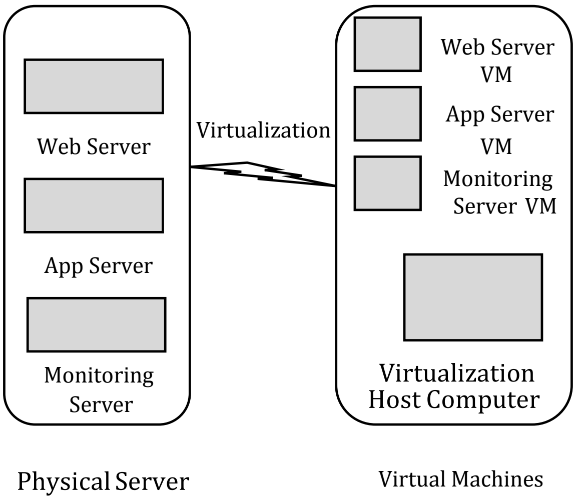

Figure 1. Server consolidation.

The contributions of the paper are as follows:

1. Firstly the active physical servers are optimally found with the help of cuckoo search algorithm.
2. Then the data defragmentation process is carried out with some equations.
3. It finally allocates the resources with the help of the Enhanced Cloud Resource Consolidation algorithm.

1.3. Organization of the Paper

The remaining part of this paper is organized as follows: Section 2 describes the related work. Section 3 describes the problem of residual resource fragmentation and its consequences. Section 4 describes the solution methodology. Section 5 describes the experimental setup. Section 6 discusses the results and analysis. Section 7 provides the conclusion analysis and the future scope of the work.

1.4. Related Work

In this paper we discuss about the problem of reducing active PMs. While considering the server consolidation problem many of the earlier authors in their work they have projected the consequences of their server consolidation process [2]. In sercon algorithm [3] the metrics identified are cpu and memory load capacity. In the Entropy [4] algorithm the no op migrations are identified and the resources considered are the cpu and memory. In the Miyako [5] algorithm the metrics used are the dirty page bit. Here the resource considered is the memory. In sandpiper [6] algorithm they set hotspot as the goal and the resources identified are CPU, memory and the network. In memory buddies [7] identified the resource considered as the memory. In khanna's algorithm [8] server consolidation is considered as goal. Residual capacity is identified as the metrics. The above explanation is given in the following **Table 2**.

This problem is technically identified as Server sprawl. In the state of art for server consolidation different approximation approaches have been proposed using various techniques. Many of the inquiring techniques such as First fit and best fit decreasing [9], Harmonic and Cardinality Constrained Harmonic approach [10], Improvised First Fit Decreasing [11], Modified Best Fit Decreasing [12], Sercon [13] are the applications of the most widely used bin packing algorithms. Some of the approaches like Dynamic Round Robin Approach [14], Adaptive Threshold based approach [15], Control Theoretic solution [16], pMapper [17], 2-Phase Optimization Method [18], Genetic Algorithm Based Approach [19] afford different methods and techniques for reducing energy consumption and reducing power methods such as LP formulation.

Table 1. Parameter inputs in resource allocation.

Parameter	Service Provider	Data Owner
Resource Status	Yes	No
Agreed Contract between data owner and provider	Yes	Yes
Provider Offerings	Yes	No
Available Resources	Yes	No
Application Requirements	No	Yes

Table 2. Algorithms for server consolidation.

Algorithms Used	Goal of the alg	Metrics Concentrated	Resource Considered
Sercon	Server Consolidation	Cpu & Memory Load Capacity	cpu, memory
MiyakoDori [4]	Server Consolidation	Dirty Page Bit	Memory
Entropy [5]	Server Consolidation	No. of Migrations	cpu, memory
Sandpiper [6]	Hotspot Mitigation	Volume, Volume/size	cpu, memory & network
Memory Buddies [7]	Server Consolidation & Hotspot Mitigation	Memory Size	memory
Khanna's Algorithm [8]	Server Consolidation	Residual Capacity, Variance	cpu, memory

2. Problem Statement

In existing systems, residual resource fragmentation is not much concentrated. Some papers concentrate on reducing the physical servers which leads to lower power consumption. Still when residual resources are concentrated and the resources are reused means it gives much cost reduction. Residual Resource Fragmentation refers to the state of the data center where sufficient amount of residual resources are available for any new VM allocation or VM reallocation, but are fragmented and distributed across multiple active PMs, rendering them unusable. The process of reducing residual resource fragmentation is called Residual Resource Defragmentation. Improving resource utilization only, it is not possible to improve utilization of CPU, Memory and Network Bandwidth simultaneously using a single scalar score. If a scalar score is used to decide on the destination PM, it leads to fragmentation of residual resources. The problem of resource fragmentation renders the residual resources useless or less useful, thereby adding to the cost incurred to the data center provider. While reducing the residual resource fragmentation in each consolidation interval, thereby reducing migrations that may be required for dynamic resource consolidating. So, here we will propose a dynamic resource consolidating and allocation technique with optimal residual resource fragmentation process.

3. Solution Methodology

The proposed method comprises of 3 phases.

a) Initially, active physical servers are selected by optimization algorithm

i.e., optimal servers can be selected by employing binary cuckoo search algorithm. The cost will be the fitness to select the optimal servers. Here the active Physical machines are identified using the Binary cuckoo search algorithm. Binary cuckoo search algorithm is the binary version of the cuckoo search algorithm. It is proposed by Nozarian, where CS algorithm is developed by the Xin-she yang in 2009. For simplicity in describing Cuckoo Search, three rules are executed: 1) Each cuckoo lays one egg at a time, and dump its egg in randomly chosen nest; 2) The best nests with high quality of eggs will carry over to the next generations; 3) The number of available host nests is fixed, and the egg laid by a cuckoo is discovered by the host bird with a probability pa ∈ [0, 1].

Let rc(i), rm(i) and rn(i) be the CPU requirement, Memory requirement & Network Bandwidth requirement of VM v(i), where i = 1, 2, 3…n. Let *Rc*, *Rm*, *Rn* be the thresholds for CPU utilization, Memory utilization and Network Bandwidth utilization respectively in the PMs. Let the current state of the PMs be represented by N(j), the set of VMs present in the jth PM, where $j = 1, 2, 3, \cdots, m$ [1]. The problem definition is as follows. To find the threshold N(j), the set of VMs consolidated into a PM p(j),where $j = 1, 2, 3, \cdots, l$ after performing server consolidation the following Equations (1)-(3)should be solved.

$$\sum_{k \in N(j)} rc(k) \le Rc \quad (1)$$

$$\sum_{k \in N(j)} rm(k) \le Rm \quad (2)$$

$$\sum_{k \in N(j)} rn(k) \le Rn \quad (3)$$

For all $j = 1, 2, 3, \cdots, l$.

b) After that, the maximum utilization of resources on these optimal servers is calculated.

Here the resources are categorized based on the type and allocated to suitable physical servers or virtual machines which will reduce the data defragmentation in data centers. Here the maximum utilization of the resources is identified. The VMs which are capable of migrating can be found out and the migration process takes place. Here the VMs and PMs Migration can be carried out only if the following conditions are satisfied:

- The PM from which the set of VMs is chosen should have all 3 resource utilization less than or equal to the corresponding resource utilizations in the candidate PM.
- The exchange of VMs between the PMs should improve the residual resource defragmentation in the two PMs.

The conditions are given in the following **Table 3**.

Here the Central Processing unit, Memory and the Network bandwidth are playing the main roles. Here the above six conditions covers the all satisfactory conditions for the resource utility. The defragmentation process is carried out here with the following Equation (4).

$$\text{Defragment} = \frac{\sum_{i=1}^{m}\left(T_j - C_{ij}\right)^2}{\sum_{i=1}^{m}\left(T_j - C_{ij}\right)} \tag{4}$$

where M: the number of active PMs,

T_{ji}: The threshold for the jth resource,

C_{ij} : The current utilization of the jth resource in the ith P.

Higher the value of the defragmt, the lesser is the resource fragmentation. Thus the value of the defragment matter for the residual resource defragmentation. Thus for the concluded defragmentation value, it can be calculated with the formula of consolidation of the defragmented values. *i.e.* Total defragmt T = defragmt CPU+ defragmt MEM + defragmt BW.

Thus in the second phase the residual resources are defragmented. The thickness is calculated here by on the basis of the cost and memory. In this part the server consolidation takes place and the defragmentation process is taking place. Now the task is ready for the allocation process.

c) Finally, the resources are allocated and scheduled using Enhanced cloud resource consolidating (ECRC).

Maximum utilization of resources on these optimal servers is calculated in the second phase. The resources are allocated and scheduled using enhanced cloud resource consolidating (ECRC). Here the traditional cloud resource consolidating algorithm is enhanced. In the paper [20], the author explained the ant colony algorithm for the easiest path finding technique. Levy's Flight algorithm [21] was explained the random walk method. It is useful in the efficient routing in the network.

In ECRC algorithm the data defragmentation size will be the fitness to select the resource corresponding to the user task. The server consolidation process should carry out the residual resource defragmentation so that the left over resources will be usable to the maximum extent. So we have to select the resource based on the user's requirement. If the available resource in the cloud is high when compared to user requirement we have to fix one score value based on the score value the fitness for firefly algorithm is fixed. Based on the optimal value we defragment the data for user requirement. The ECRC algorithm workflow is given below: (**Figure 2**)

ECRC Algorithm:

Output: best node from the given better nodes.

```
public class ecrc

{public static void update()

{ArrayList<ArrayList<Double>> tem=new ArrayList<ArrayList<Double>>();

for(int i=0;i<velo.size();i++)

tem.add(velo.get(i));

velo.clear();

for(int i=0;i<tem.size();i++)

{ArrayList<Double> t=new ArrayList<Double>();

for(int j=0;j<tem.get(i).size();j++)

{double d=tem.get(i).get(j)+(0.2*(genran()*((initial.get(pb).get(j))-(initial.get(i).get(j)))))+(0.2* (genran()*(gbarray.get(j)-(initial.get(i).get(j)))));

double d1=tem.get(i).get(j)+(0.2*(genran()*((initial.get(pbh).get(j))-(initial.get(i).get(j)))))+(0.2*(genran()*(gbharray.get(j)-(initial.get(i). get(j)))));

if(d<d1)

t.add(d);

else

t.add(d1);}velo.add(t);}

for(int i=0;i<velo.size();i++)

System.out.println(velo.get(i)+" nv");

}}
```

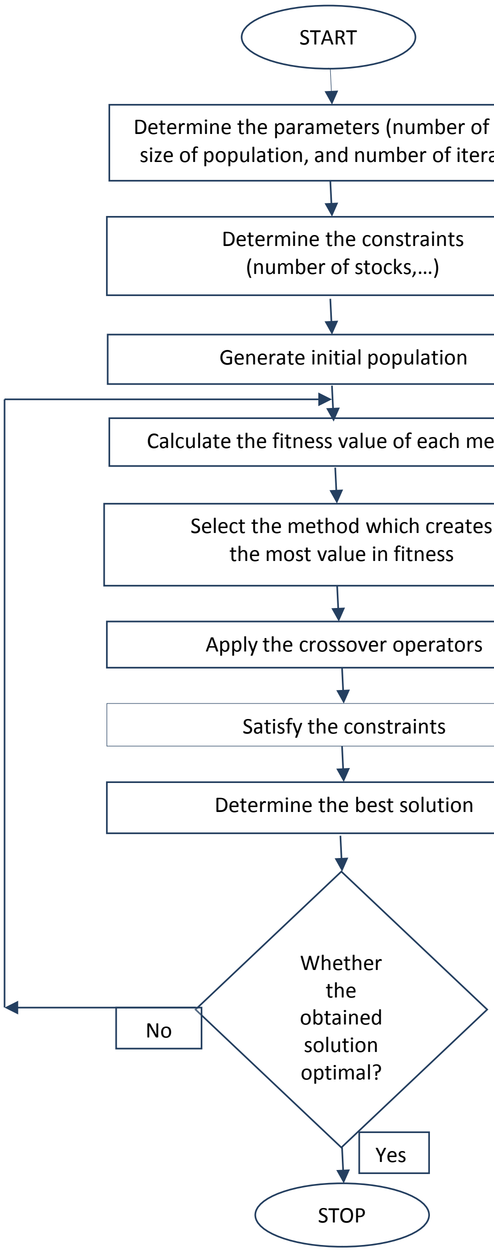
START
Determine the parameters (number of stocks, size of population, and number of iterations)
Determine the constraints (number of stocks,...)
Generate initial population
Calculate the fitness value of each member
Select the method which creates the most value in fitness
Apply the crossover operators
Satisfy the constraints
Determine the best solution
Whether the obtained solution optimal?
No
Yes
STOP

Figure 2. Workflow for the ECRC algorithm.

Table 3. Conditions for PM migrations.

Category	Relative resource utilization
1	CPU >= MEM >= BW
2	CPU >= BW >= MEM
3	MEM >= BW >= CPU
4	MEM >= CPU >= BW
5	BW >= MEM >= CPU
6	BW >= CPU >= MEM

Workflow of ECRC Algorithm: (Figure 2)

Using the ECRC algorithm, the better nodes are found out. From the better nodes, top of the table node is found according to the user's task requirement. If two or more nodes found out, then the thickness is calculated with the help of the cost. Thus by using this technique the server sprawl problem can face new changes. In multiple provisioning stages as well as a long-term plan, the ECRC algorithm can provision the resources is being used.

The implementation will be done in cloudsim along with java. The performance of the proposed system will be compared with the existing system.

4. Experimental Results

4.1. Experimental Study 2 (Data Defragmentation)

Figure 3 shows that the residual resource fragmentation for the server consolidation with the comparison of the existing consolidation algorithm OVMP and OCRP approaches with our work ECRC algorithm. The datasets for the residual resource defragmentation is listed below in Table 4. It is noted that in each and every consolidation interval, residual resource defragmentation is higher for ECRC while compared to the OVMP and OCRP algorithm. Thus it is concluded that the proposed ECRC algorithm gives better result. This is because higher the defragment lesser the resource fragmentation.

Table 4. The sample datasets for the defragmentation.

	OVMP	OCRP	ECRC
1800	71	64	65
5400	28	31	32
9000	24	30	30
12,600	20	21	22
16,200	25	26	28
19,800	15	17	18
23,400	10	15	16
27,000	12	11	15
30,600	14	13	15
34,200	10	14	15
41,400	12	16	16
45,000	14	17	17
48,600	13	17	18
52,200	18	18	18
55,800	10	11	11
59,400	20	20	21
63,000	30	30	32
66,600	12	15	19
70,200	18	21	23
73,800	10	14	15
77,400	11	14	15
81,000	10	14	15
84,600	11	14	14

4.2. Experimental Study 2 (Energy Consumption)

From Figure 4, the cost reduction can be illustrated. Table 5 explains the dataset for the data center energy consumption. Optimal virtual machine placement, Optimized cloud resource provisioning, and Enhanced cloud resource consolidating approaches considering different workloads. From Figure 4 we can infer that the energy consumptions in the three approaches are almost same. Hence the proposed approach ECRC succeeds in effectively reducing the energy consumption of the data center and thereby reduces overall costs comparatively with OVMP and OCRP.

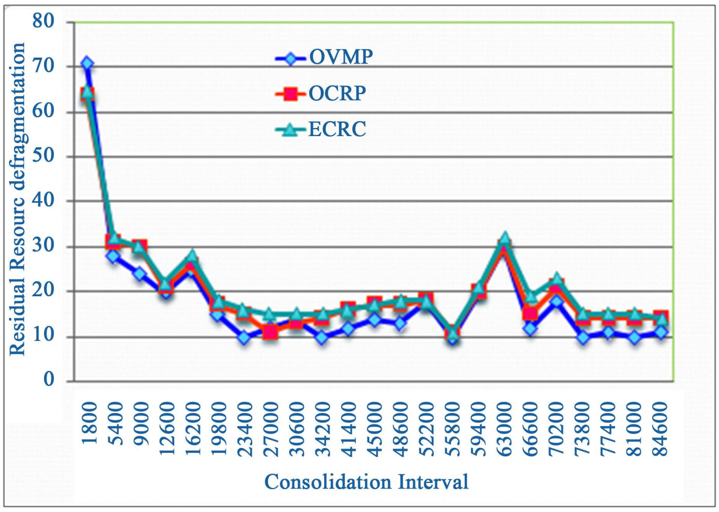

Figure 3. Residual resource defragmentation at different consolidation intervals.

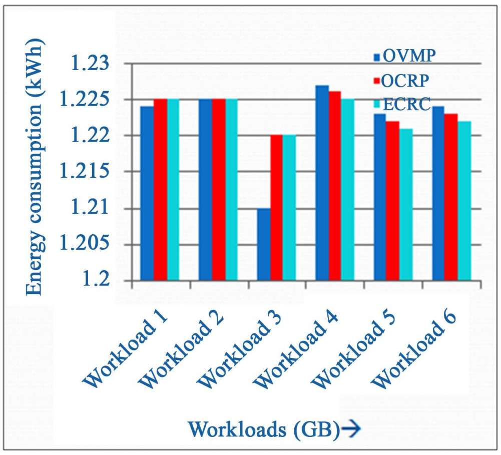

Figure 4. Data center energy consumption for different consolidation algorithms.

Table 5. Sample data sets for the data center energy consumption.

Workload in size	OVMP	OCRP	ECRC
0.75 GB	1.224	1.225	1.225
1 GB	1.225	1.225	1.225
0.55 GB	1.21	1.22	1.22
1.25 GB	1.227	1.226	1.225
0.85 GB	1.223	1.222	1.221
0.80 GB	1.224	1.223	1.222

5. Conclusions

In Cloud computing, server sprawl is one of the rarely identified problems. Here the server consolidation plays a vital role.

By using binary cuckoo search, the active servers are identified. By using enhanced cloud resource consolidating algorithm, the exact match of the particular task can be identified. From the top identified nodes, the best fit for our task is identified. Here, defragmentation assigns the thickness of the values. Here the parameter used for identifying the best node is the cost. Binary cuckoo search is used here for the grouping the task and for searching purpose. ECRC algorithm used here can find the appropriate top 5 match node for our task. This work can be mainly held using the ECRC technique. In this technique, the best node from the first five nodes which is selected using the ECRC algorithm can be listed to the user. While comparing with the existing server allocation techniques our ECRC technique give more cost fitness. By using ECRC, the results vary from the existing Residual resource allocation techniques. The algorithm used here are binary cuckoo search and ECRC. This is because the survey conducted from the various author's works. There are various parameters to calculate the output. Here the memory allocation and the cost are used.

References

[1] Sunil Rao, K. and Santhi Thilagam, P. (2014) Heuristics Based Server Consolidation with Residual Resource Defragmentation in Cloud Centers. Elsevier, 87-92.

[2] Thakur, S. and Jawahar, A.K. (2013) Server Consolidation Algorithms for Cloud Computing Environment: A Review. *IJARCSSE*, **3**.

[3] Murtazaer, A. and Oh, S. (2011) Sercon: Server Consolidation Algorithm Using Live Migration of Virtual Machines for Green Computing. *IETE Technical Review*, **28**.

[4] Hermenier, F., Muller, G., Lorca, J.L.X. and Menuad, J.-M. (2008) Entropy: A Consolidation Machine Manager for Clusters. *Proceedings of the* 2009 *ACM SIGPLAN/SIGOPS Internal Conference on Virtual Execution Networks*, 41-50.

[5] Akiyama, S., Hirofuchi, T., Honiden, S. and Takano, R. (2012) MiyakoDori: A Memory Reusing Mechanism for Dynamic VM Consolidation. *IEEE Fifth International Conference on Cloud Computing*. http://dx.doi.org/10.1109/cloud.2012.56

[6] Wood, T., Shenoy, P., Venkataramani, A. and Yousif, M. (2009) Sandpiper: Black-Box and Gray-Box Resource Management for Virtual Machines. *Journal of Computer Networks*, **53**, 2923-2938.

[7] Wood, T., Tarasuk-Levin, G., Shenoy, P., Corner, M.D., Cecchet, E. and Desnoyers, P. (2009) Memory Buddies: Exploiting Page Sharing for Smart Colocation in Virtualized Data Centers. *Proceedings of the ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments*, 31-40.

[8] Khanna, G., Kochut, A., Kar, G. and Beaty, K. (2006) Application Performance Management in Virtualized Server Environments. 10*th IEEE/IFIP Conference on Network Operations and Management in Virtualized Server Environments*, NOMS.

[9] Ho, Y., Liu, P. and Wu, J.-J. (2011) Server Consolidation Algorithms with Bounded Migration Cost and Performance Guarantees in Cloud Computing. 2011 *Fourth IEEE International Conference on Utility and Cloud Computing*, *UCC*, 154-161.

[10] Bein, D., Bein, W.W. and Venigella, S. (2011) Cloud Storage and Online Bin Packing. *IDC*, 63-68. http://dx.doi.org/10.1007/978-3-642-24013-3_7

[11] Ajiro, Y. and Tanaka, A. (2007) Improving Packing Algorithms for Server Consolidation. *International CMG Conference*, 399-406.

[12] Beloglazov, A. and Buyya, R. (2010) Energy Efficient Allocation of Virtual Machines in Cloud Data Centers. *Proceedings of the* 2010 10*th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, *CCGRID* 10, Washington DC, 577-578.

[13] Marzolla, M., Babaoglu, O. and Panzieri, F. (2011) Server Consolidation in Clouds through Gossiping. *WoWMoM*'11: *Proceedings of the* 12*th IEEE International Symposium on a World of Wireless*, *Mobile and Multimedia Networks*, Washington DC, 1-6.

[14] Lin, C.-C., Liu, P. and Wu, J.-J. (2011) Energy-Aware Virtual Machine Dynamic Provision and Scheduling for Cloud Computing. 2011 *IEEE International Conference on Cloud Computing*, *CLOUD*, 736-737.

[15] Beloglazov, A. and Buyya, R. (2010) Adaptive Threshold and Based Approach for Energy Efficient Consolidation of Virtual Machines in Cloud Data Centers. *Proceedings of the* 8*th International Workshop on Middleware for Grids, Clouds and e-Science*, *MGC*'10, 41-46.

[16] Zhang, Q., Zhani, M.F., Zhang, S., Zhu, Q., Boutaba, R. and Hellerstein, J.L. (2012) Dynamic Energy-Aware Capacity Provisioning for Cloud Computing Environments. *Proceedings of the* 9*th International Conference on Autonomic Computing*, *ICAC*'12, 145-154. http://dx.doi.org/10.1145/2371536.2371562

[17] Verma, A., Ahuja, P. and Neogi, A. (2008) pMapper: Power and Migration Cost Aware Application Placement in Virtualized Systems. *Proceedings of the* 9*th**ACM/IFIP/USENIX International Conference on Middleware*, *Middleware*'08, 243-264.

[18] Taheri, M.M. and Zamanifar, K. (2011) 2-Phase Optimization Method for Energy Aware Scheduling of Virtual Machines in Cloud Data Centers. *The International Conference of ICITST*, 525-530.

[19] He, L., Zou, D., Zhang, Z., Chen, C., Jin, H. and Jarvis, S.A. (2014) Developing Resource Consolidation Frameworks for Moldable Virtual Machines in Clouds. *Future Generation Computer Systems*, **32**, 69-81.

[20] Visu, P. and Kannan, E. (2013) Combined Swarm Intelligence Routing Protocol for MANET. **12**, 791-797.

[21] Yang, X.-S. and Deb, S. (2009) Cuckoo Search via Lévy Flights. *IEEE Transactions World Congress on Nature & Biologically Inspired Computing*.

Scientific Research Publishing

Submit or recommend next manuscript to SCIRP and we will provide best service for you:

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.
A wide selection of journals (inclusive of 9 subjects, more than 200 journals)
Providing 24-hour high-quality service
User-friendly online submission system
Fair and swift peer-review system
Efficient typesetting and proofreading procedure
Display of the result of downloads and visits, as well as the number of cited articles
Maximum dissemination of your research work

Submit your manuscript at: http://papersubmission.scirp.org/