Green Cloud: Smart Resource Allocation and Optimization using Simulated Annealing Technique

AkshatDhingra
M.Tech Research Scholar, Department of Computer Science and Engineering,
Birla Institute of Technology, Ranchi, India

Sanchita Paul
Assistant Professor, Department of Computer Science and Engineering,
Birla Institute of Technology, Ranchi, India

Abstract

Cloud computing aims to offer utility based IT services by interconnecting large number of computers through a real-time communication network such as the Internet. There has been a significant increase in the power consumption by the data centres that host the Cloud applications because of the growing popularity of Cloud Computing in more and more organisations involved in various fields. Hence, there is a need to develop solutions that aim to save energy consumption without compromising much on the performance. Building such solutions would not only help in reducing the carbon footprint but would also cut down the costs without much compromise on SLA violations thereby benefitting the cloud service providers. In this paper, Simulated Annealing Optimizing Technique has been used for the purpose of continuously optimizing the placement of the VMs (Virtual Machines) over the hosts in order to minimize the power consumption hence providing cost benefits to the service provider. The results make it clearly evident that by making use of virtualisation at the data centre level and the optimizing the virtual resource allocation could significantly reduce the power consumption by the servers.

Keywords: Energy Efficient Data Centre, Simulated Annealing, Resource Allocation Optimization, Power Consumption Minimization, Maximum Utilisation, Minimum Utilisation Minimum Migration Overhead, Best Fit Bin Packing Algorithm.

1. Introduction

Cloud computing offers a unique model in which the users access services based on their requirements without regard to where the services are hosted. IT majors like Google, Microsoft, Yahoo, and IBM are rapidly deploying data centres in various locations around the world solely for the purpose of providing Cloud computing services to the users. Use of cloud computing to access various services offered is hence rapidly rising.

The designers have been mainly focussing on improving the performance of computing systems and hence the performance has been steadily growing driven by more efficient system designs and increasing the density of the components described by Moore's law. Although the performance per watt ratio has been constantly rising, the total power draw by computing systems is hardly decreasing. In fact, it has been increasing every year.

Apart from the overwhelming operating costs due to high energy consumption, another rising concern is the increasing greenhouse gas emissions caused by this high energy consumption. The total carbon footprint of the IT industry that includes personal computers, mobile phones, and telecom devices and servers was 830 $MtCO_2e$ in 2007. This is 2% of the estimated total emissions and this figure is expected to grow in the coming years [2]. Therefore, designing the modern computing systems that consume minimum possible power has become one of the main objectives.

IT systems could be made greener either by making them energy efficient or by making use of renewable sources of energy. As the latter is hard to find in abundance so that it could be used on a mass scale, energy efficiency is expectedto be the main focus of research in the near future. IT companies are learning that cutting emissions and cutting costs naturally go hand in hand.Making systems energy efficient consequently reduces operating costs.

Cloud Computing has the potential to have a massive impact, positive or negative, on the future carbon footprint of the IT sector. Cloud computing data centres are now consuming 0.5% of all the generated electricity in the world, a figure that continues to grow as Cloud Computing becomes more popular. However, the large data centres required by clouds have the potential to provide the most efficient environments for computing.The main aim of Energy-Aware Computing is to promote awareness of energy consumption at both software and hardware levels and hence consumes lesser amount of power.

Dhingra et. al [13] gives a brief review of the various power management schemes at the data centre level with the help ofvirtualization making use of controllers at the local and global level. It also gives a detailed comparison of the various framework proposed for energy efficient cloud computing. This paper begins with a brief introduction of cloud computing, it's potential and the need to make cloud computing energy aware in Section 1. Section 2, gives an introduction to a power consumption model [3] that explains how could the power consumed by the servers be estimated. In Section 3, an architectural framework of an energy aware cloud is presented [1]. Section 4 presents an energy efficient resource allocation optimisation algorithm with the help of Simulated Annealing Technique. Section 5 presents the results of the proposed technique and their analysis. Section 6 concludes this paper with limitations and future directions.

2. Modelling Server Power Consumption

To develop new policies for Dynamic Power Management and understand their impact, it is necessary to create a model of dynamic power consumption. Such a model has to be able to predict the actual value of the power consumption based on some run-time system characteristics. One of the ways to accomplish this is to utilize power monitoring capabilities that are built-in modern computer servers. This instrument provides the ability to monitor power usage of a server in real time and collect accurate statistics about the power usage. Based on this data it is possible to derive a power consumption model for a particular system. However, this approach is complex and requires collection of the statistical data for each target system.

Fan et al. [4] have found strong relationship between the CPU utilization and total power consumption by a server. Hence, the idea behind the proposed model is that the power consumption by a server grows linearly with the growth of CPU utilization from the value of power consumption in the idle state up to the power consumed when the server is fully utilized. This relationship can be expressed as:

$$P(u) = P_{idle} + (P_{busy} - P_{idle}) * u \qquad (1)$$

Where P is the estimated power consumption at a given instant of time; P_{busy} is the power consumed when the server is fully utilized; P_{idle} is the power consumed by the idle server; and u is the CPU utilization. The CPU utilization may change over time due to variability of the workload.

Due to the proliferation of multi-core CPUs and virtualisation, modern servers are typically equipped with large amounts of memory, which begins to dominate the power consumption by the server. Hence, real data on power consumption provided by SPECpower benchmark has been used instead of using an analytical model to determine the power consumption at a particular instant.

3. Framework

This paper aims to optimize the VM allocation in order to reduce the energy consumption without compromising on the SLA violations. A framework for the same has been proposed [1] by Dhingra et.al that briefly explains the objectives involved.

4. Data Centre Resource Allocation Optimisation

The allocation of resource to the physical hosts and the optimisation of the resource usage in the data centre happen in two steps.

In the first step, the VM is allocated to that physical host that offers to execute it consuming the minimum amount of power. In the second step, the resource usage optimisation with simulated annealing takes place, which considers the power consumption by all the hosts in the data centre and then aims to minimize it.

The steps involved in this algorithm are given below for better understanding.

i. Initialize all the physical hosts (servers).
ii. Initialize all the VMs.
iii. Sort VMs in decreasing order of their CPU requirements.
iv. Allocate the VM to the physical machine that executes this VM consuming minimum amount of power.
v. Find out all the over-utilised and under-utilised hosts and the VMs that could be migrated from these hosts.
vi. The VM to be migrated from the set of migratable VMs is governed by the following heuristics (briefly explained later)
    a. Minimum Utilisation.
    b. Maximum Utilisation.
    c. Minimum Migration Overhead.
    d. Random Choice
vii. In each step of finding the host for the VM, find that physical host (server) which would offer to minimise the power consumption by the data centre if the VM is migrated to the host under consideration.

A point to be noted here is that each physical host has upper utilisation threshold and lower utilisation thresholds associated with it that determine whether the host is over-utilised or under-utilised respectively. This also means that a VM will not be allocated to a physical machine if its allocation takes the resource utilisation of that particular physical host beyond the set threshold and those hosts whose utilisation is below the threshold must be turned-off in order to save energy.

The heuristics used for performing optimum migrations are explained below.

a) **Maximum Utilisation**: Perform the migration of that VM from the overloaded hosts that have the maximum CPU utilisation.
b) **Minimum Utilisation**: Perform the migration of that VM from the overloaded hosts that have the minimum CPU utilisation.
c) **Random Choice**: This policy selects a VM to be migrated according to a uniformly distributed discrete random variable X, whose values index a set of VMs V_j allocated to a host j.
d) **Minimum Migration Overhead:** Select that VM which has the minimum CPU utilisation and requires minimum amount of memory (RAM) [10][11].

4.1. Simulated Annealing: An Introduction

Simulated Annealing is probabilistic meta-heuristic for the global optimization problem of locating a good approximation to the global optimum in a given function in a large and discrete search space [5]. This method was proposed in Kirkpatrick, Gellet and Vecchi (1983) and Cerni for finding the global minimum of a function that may possess several local minima. This technique is inspired by the physical process 'annealing' wherein a solid is slowly cooled so that eventually when the structure is frozen, it happens at the minimum energy configuration.

In simulated annealing, a temperature variable is kept to simulate the heating process. It is initialised to a high value and then is reduced slowly as the algorithm iterates. When the value of the temperature variable is high, the probabilities of accepting the solution that are worse than the current solution have a high chance of getting accepted. As the temperature goes down, the chance of worse solutions getting accepted becomes lesser. The algorithm gradually focuses in on the optimum solution. The gradual cooling process is governed by the 'cooling factor' which makes the simulated annealing technique more effective to find a close to optimum solution.A pseudo code is shown below for better understanding [12].

```
s ← s0; e ← E(s)                    // Initial state, energy.
sbest ← s; ebest ← e                // Initial "best" solution
k ← 0                               // Energy evaluation count.
while k <kmax and e >emax           // While time left & not good enough:
  T ← temperature(k/kmax)           // Temperature calculation.
snew ← neighbour(s)                 // Pick some neighbour.
enew ← E(snew)                      // Compute its energy.
if P(e, enew, T) > random() then    // Should we move to it?
s ← snew; e ← enew                  // Yes, change state.
ifenew<ebestthen                    // Is this a new best?
sbest ← snew; ebest ← enew          // Save 'new neighbour' to 'best found'.
k ← k + 1                           // One more evaluation done
returnsbest                         // Return the best solution found.
```

A flowchart has been shown below explaining the above algorithm.

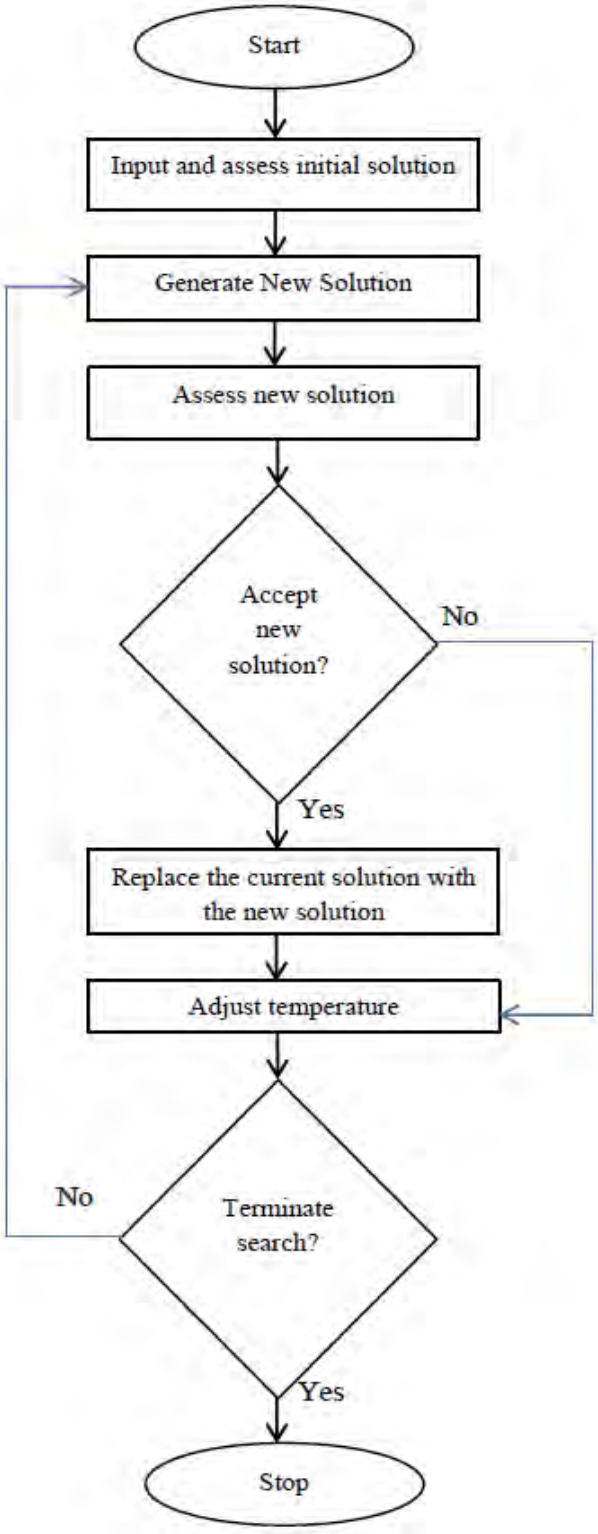

Figure 1: Flowchart showing the Simulated Annealing procedure

4.2. Resource Allocation Optimisation Algorithm using Simulated Annealing Technique

Input: Set of hosts, VM to be migrated
Output: Most Suitable Host

1. Initialise Temperature and Cooling Rate.
2. Calculate initial total power consumed by the data centre. (TotalPower)
3. For each suitable host in the data centre
    a. Calculate New Power if VM is migrated to this host. (NewPower)
    b. If NewPower<TotalPower.
        i. Accept New Solution.
    c. Else calculate Acceptance Probability=$e^{(TotalPower-NewPower)/Temperature}$.
        i. If Acceptance Probability>=0 and Acceptance Probability<1
            1. Accept New Solution. (NewPower)
4. Modify Temperature. (Temperature=Temperature*(1-Cooling Rate)).

The optimisation process finds out the over-utilised and under-utilised hosts in each time frame and hence the VMs to be migrated. The above algorithm takes the set of available hosts and the VM that is to be migrated.

In the first step, the initial temperature and cooling rate are fixed. In the second step, the total power consumed by the data centre is measured. It is the initial power measured at the beginning and is not optimised. The objective of this algorithm is to optimize this value (reduce the energy consumption) as the algorithm proceeds. The third step tries to reduce the total power consumed by the data centre by causing the best VM migration possible. It also calculates the 'Acceptance Probability' in case the VM migration (if caused) does not reduce the total power consumption. A point to be noted here is that at higher values of 'Temperature' there is a higher probability of the worst solution to be accepted, but in every iteration of the algorithm, the 'Temperature' value is reduced by the factor of 'Cooling Rate' thereby increasing the tendency of the algorithm to move closer to the most optimum solution.

5. Experimental Setup and Results

As the system under consideration is a generic cloud computing environment, it would be necessary to evaluate the proposed techniques on a large scale virtualised data centre infrastructure. However, it would be difficult to conduct large scale experiments on a real infrastructure, especially in our case where it is necessary to repeat the experiment with the same conditions but with different set of heuristics. Hence, simulation has been chosen as a way to evaluate the proposed technique and heuristics. The toolkit used for the simulation of the cloud computing environment is CloudSim as it is a modern simulation framework aimed at the Cloud Computing environments and supports modelling of on-demand virtualisation enabled resource and application management unlike others (SimGrid, GandSim). Apart from the power consumption modelling and accounting, the ability to simulate service applications with variable overtime workload has also been incorporated.CloudSim makes use of Java programming language and the screenshots have been shown below.

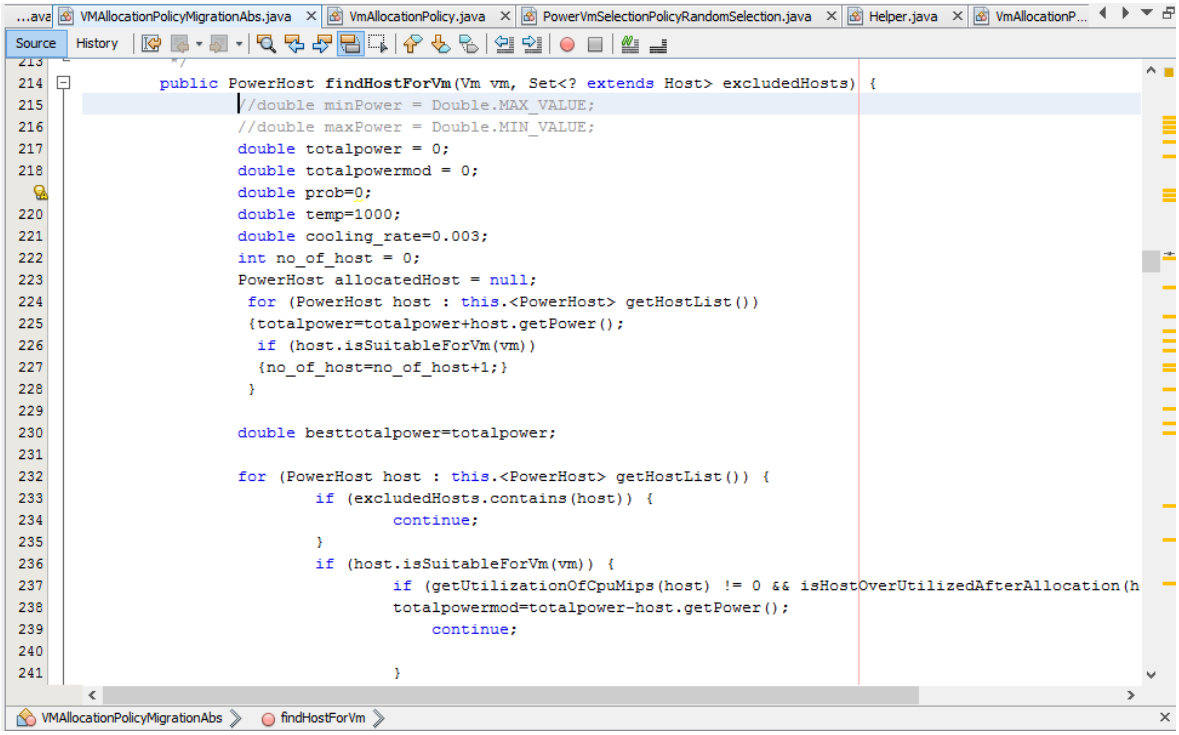

Figure 2: Screeshot 1 (CloudSim 3.0.3)

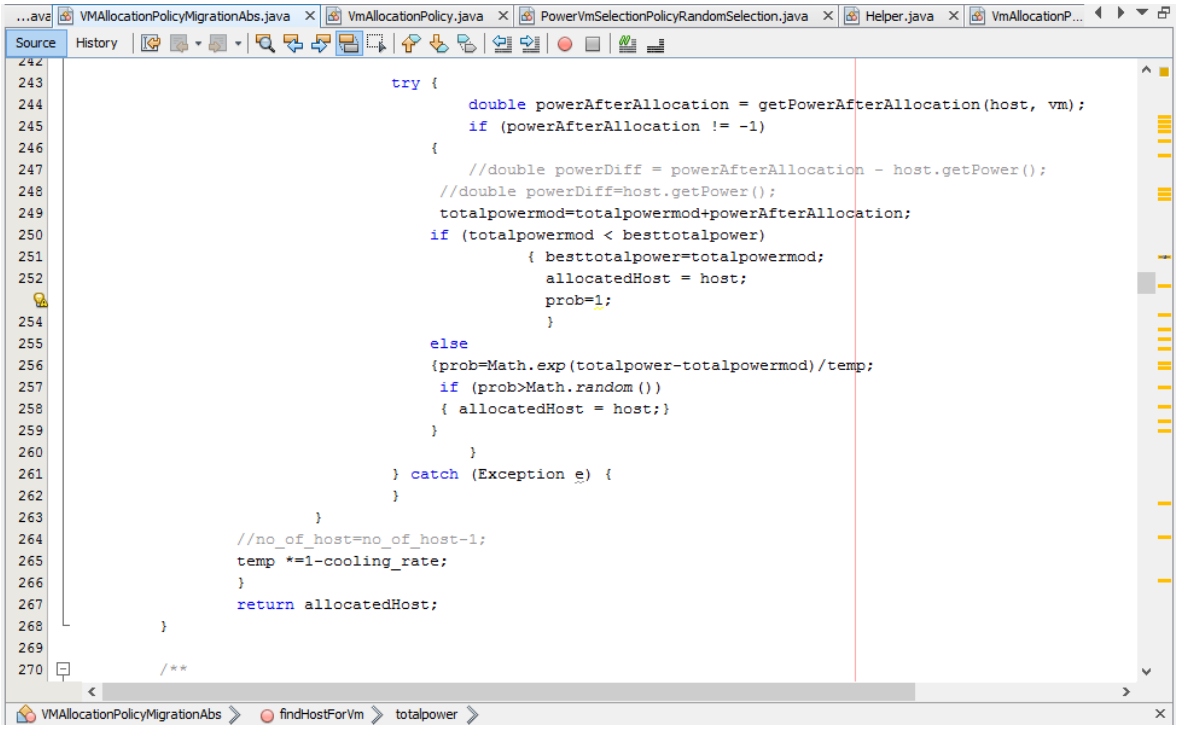

Figure 3: Screenshot 2: (CloudSim 3.0.3)

It has been assumed that the cost of VM migration is negligible considering the fact that the efficiency of VM migration is going to be improved with the advancement of virtualisation technologies. The workload for VMs is assumed to be random as it is not possible to build the exact model of such a mixed workload.

A data centre is simulated with 100 heterogeneous physical nodes. Each node is modelled to have 4 CPU cores with performance of each core equivalent to 2930 and 3000 Million Instructions per Second, 4 GB of RAM and 1 TB of storage.

Table 5: Server Configurations

Server	**Processor**	**Cores**	**MIPS**	**RAM**	**Hard Disk**
IBM X3250	Intel Xeon 3470	4	2930	4 GB	1 TB
IBM X3250	Intel Xeon 3480	4	3000	4 GB	1 TB

*MIPS: Million Instructions per Second

As explained in section 2, there is linear relationship between the power consumption and CPU utilisation. In the recent times however, due to the growth of muti-core CPUs and virtualisation, the modern servers are typically equipped with large amounts of memory, which begins to dominate the power consumption by a server [7] thereby making it difficult to model power consumption. Hence we make use of the actual data on power consumption.

The data on their power consumption values at various levels of CPU utilisation is summarised in the table below. [6] [8]

Table 6: Power Consumption by the Servers at various CPU Utilisation Levels (in Watts)

Server	0 %	10 %	20 %	30 %	40 %	50 %	60 %	70 %	80 %	90 %	100 %
IBM X3250	41.6	46.7	52.3	57.9	65.4	73	80.7	89.5	99.6	105	113
IBM X3250	42.3	46.7	49.7	55.4	61.8	69.3	76.1	87	96.1	106	113

Four types of VMs have been created whose characteristics have been summarised in the table below.

Table 7: VM Characteristics

Type	MIPS	RAM	Hard Disk
1	2500	870 MB	2.5 GB
2	2000	1740 MB	2.5 GB
3	1000	1740 MB	2.5 GB
4	500	613 MB	2.5 GB

The users submit the requests for the provisioning of 100 VMs. Each VM runs an application with varying workload which creates a CPU utilisation according to a uniformly distributed random variable.

For the results, the Non Power Aware (NPA) policy, which does not apply any power aware optimizations and assumes that all hosts run at 100% CPU utilisation consuming maximum power, has been used as the benchmark for the sake of comparison.

Several experiments were conducted for each heuristic combined with the optimization technique and various threshold values. The results have been summarised below.

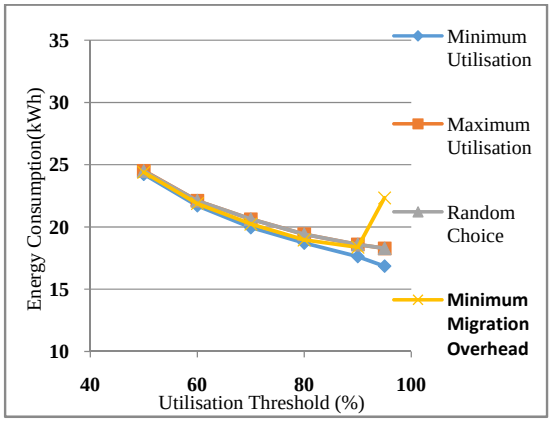

Figure 4: Power Consumption under various heuristics

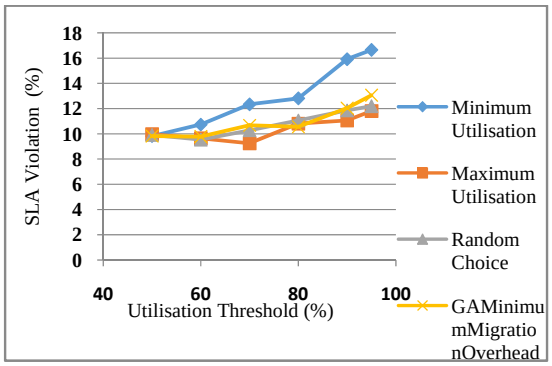

Figure 5: SLA violations under various heuristics

From the above graphs we observe that as we increase the utilisation threshold, the energy consumption decreases and the SLA violations increase. The energy consumption could be significantly reduced in comparison to the Non Power Aware policy but at the cost of some SLA violations. The results have been summarised in the table below for the purpose of comparison of various heuristics with the Non Power Aware (NPA) and Dynamic Voltage Frequency Scaling (DVFS) policies.

Table 1: Simulation Results

Policy	Energy(kWh)	Host Shutdowns	Average SLA Violations %
NPA	270.26	0	0
DVFS	29.26	0	0
Minimum Utilisation 50-95%	19.85	395	13.08
Maximum Utilisation 50-95%	20.59	532	10.43
Random Choice 50-95%	20.33	476	10.81
Minimum MigrationOverhead	20.23	447	10.99

From the above results it becomes apparent that by optimizing the allocation of VMs according to the current CPU utilisation, the energy consumption could be significantly reduced only at the cost of some SLA violations. Also, when using the hybrid technique is applied, the performance degradation due to migration could be reducedwithout much compromise in the energy consumption. This therefore makes the proposed techniques suitable in scenarios where energy savings, reduction of carbon emissions and hence profit maximisation is a greater concern and SLA violations are acceptable to some extent.

6. Limitations and future directions

A limitation in the proposed solution is that the requirements should be known in advance for initial allocation to take place. This information may not be known in certain scenarios. Hence, we aim to design such algorithms that offer a flexibility of not knowing the requirements in advance. The results are obtained with the help of simulation and it would be interesting to analyse the results if the algorithm is implemented in a real scenario.We would also like to explore some other optimization techniques steepest descent and then compare the results with those from the existing techniques.

References

[1] AkshatDhingra, Sanchita Paul, “Green Cloud Computing: Towards Optimizing Data Centre Resource Allocation”, in International Journal Engineering Research and Technology, Volume 3, Issue 2, February 2014, pp 1038-1042
[2] E. Farnworth and J.C. Castilla---rubio, "SMART 2020: Enabling the low carbon economy in the information age," Group.
[3] R. Buyya, A. Beloglazov, and J. Abawajy, ―Energy-Efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges,‖ in Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2010), Las Vegas, USA, July 12-15, 2010.
[4] X. Fan, W. D. Weber, and L. A. Barroso, ―Power provisioning for a warehouse-sized computer,‖ in Proceedings of the 34th Annual International Symposium on Computer Architecture (ISCA 2007). ACM New York, NY, USA, 2007, pp. 13–23.
[5] Dimitris Bertsimasand John Tsitsiklis, Simulated Annealing, in Statistical Science Volume 8, Number 1 (1993), 10-15.
[6] R. Buyya, A. Beloglazov, and J. Abawajy, ―Energy-Efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges,‖ in Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2010), Las Vegas, USA, July 12-15, 2010.
[7] Anton Beloglazov, and RajkumarBuyya, "Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers", Concurrency and Computation: Practice and Experience (CCPE), Volume 24, Issue 13, Pages: 1397-1420, John Wiley & Sons, Ltd, New York, USA, 2012.
[8] Minas L, Ellison B. Energy Efficiency for Information Technology: How to Reduce Power Consumption in Servers and Data Centers. Intel Press, 2009.

[9] http://www.specpowerbenchmark.org
[10] Cost of Virtual Machine Live Migration in Clouds: A Performance Evaluation
[11] SherifAkoush, RipdumanSohan, Andrew Rice, Andrew W. Moore and Andy Hopper. Predicting the Performance of Virtual Machine Migration, the 18th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'10), Miami, FL, USA, August 2010.
[12] http://en.wikipedia.org/wiki/Simulated_annealing.
[13] AkshatDhingra, Sanchita Paul, "A Survey of Energy Efficient Data Centres in a Cloud Computing Environment", in International Journal of Advanced Research in Computer and Communication Engineering, Volume 2, Issue 10, October 2013, pp. 4033-4041.