

Energy efficiency in virtual machines allocation for cloud data centers with lottery algorithm

Mehran Tarahomi¹, Mohammad Izadi²

¹Kish International Campus, Sharif University of Technology, Iran

²Department of Computer Engineering, Sharif University of Technology, Iran

Article Info

Article history:

Received Aug 3, 2018

Revised Aug 30, 2018

Accepted Sep 16, 2018

Keywords:

Cloud data centers

Energy

Virtual machins

ABSTRACT

Energy usage of data centers is a challenging and complex issue because computing applications and data are growing so quickly that increasingly larger servers and disks are needed to process them fast enough within the required time period. In the past few years, many approaches to virtual machine placement have been proposed. This study proposes a new approach for virtual machine allocation to physical hosts. Either minimizes the physical hosts and avoids the SLA violation. The proposed method in comparison to the other algorithms achieves better results.

*Copyright © 2019 Institute of Advanced Engineering and Science.
All rights reserved.*

Corresponding Author:

Mehran Tarahomi,
Kish International Campus,
Sharif University of Technology,
Tehran, Iran.
Email: tarahomi@ce.sharif.edu

1. INTRODUCTION

A cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers [1]. Lowering the energy usage of data centers is a challenging and complex issue because computing applications and data are growing so quickly that increasingly larger servers and disks are needed to process them fast enough within the required time period. Green Cloud computing is envisioned to achieve not only the efficient processing and utilization of a computing infrastructure, but also to minimize energy consumption [2].

Improving energy efficiency has become increasingly important in data centers in recent years to cut down the tremendous amounts of electricity consumption. The power dissipation of the physical servers is the root cause of power usage of other systems, such as the cooling systems [3]. Currently, resource allocation in a Cloud data center aims to provide high performance while meeting SLAs, without focusing on allocating VMs to minimize energy consumption. To explore both performance and energy efficiency, three crucial issues must be addressed. First, excessive power cycling of a server could reduce its reliability. Second, turning resources off in a dynamic environment is risky from the QoS perspective. Due to the variability of the workload and aggressive consolidation, some VMs may not obtain required resources under peak load, and fail to meet the desired QoS. Third, ensuring SLAs brings challenges to accurate application performance management in virtualized environments. All these issues require effective consolidation policies that can minimize energy consumption without compromising the user-specified QoS requirements [4].

To fine the solution to the virtual machine allocation to physical host, three sub-issue should be addressed. The [5] study has divided the main challenges of this problem to three sub-issues.

- a. When a virtual machine should be migrated?

There are two conditions to migration. When a physical host is over-loaded or under-loaded. For this purpose, various algorithms have been introduced.

- b. Which virtual machine should be migrated?

When one physical host is under-loaded, some of its virtual machined should be selected for migration.

- c. Where virtual machine should be migrated?

The destination should be chosen for second's virtual machines.

The virtual machine allocation to physical hosts or the third problem is similar to classic bin packing that is a NP-hard problem. Heuristic algorithms are one of the first methods that attempt to minimize the energy consumption. One of this algorithm's major problem is their time and are not suitable for big problems, because of their nature these algorithms are not able to search extendable space. One of the other methods for minimizing the energy consumption in data centers is using evolutionary algorithms. Evolutionary algorithms can search better the problem space so that ensures QoS and also reduces the energy consumption.

In this paper new approach based on lottery algorithm is proposed for virtual machine allocation to physical hosts. The results show decreasing 31.25 percent in energy consumption in comparison to PSO and genetic algorithms. The purpose of this study is achieving a pattern for virtual machine allocation to physical hosts by lottery algorithm. In other words a new approach for solving the third-issue has been proposed in this paper to minimize the switch on physical hosts and minimize the energy consumption.

In the next section the related works has reviewed, the third section describes the problem in detail. The proposed algorithms is proposed in the four section. The evaluation parameters and simulation and the setting for simulation is described in section 5. The analyzing the performance of proposed algorithm is described in section 6, section 7 is conclusion of this study.

2. THE RELATED WORKS

The allocation of virtual machines to physical hosts problem is divided to three sub-issues. This section discuss about the previous works for each sub-issues.

2.1. When a virtual machine should migrate?

The first issue is related to the migration time of virtual machine to physical host. There are two conditions for this placement. The first condition is when the physical host is over-loaded. In other words when the load of physical host exceeds the determined threshold, to avoid the risk of SLA's violation because of lake of physical host's resources, the virtual machine should migrate to other physical host. The second condition is for migrating the virtual machine is when a physical host is under-loaded. When a load of virtual machine decreases its total processing is moved to one switched on physical hosts and switch it off. Existing algorithms for detecting over-load physical host are as follows:

2.1.1. Local regression algorithm

The next heuristic is based on the Loess method (from the German l'oss-short for local regression) proposed by Cleveland [6]. The main idea of the local regression method is fitting simple models to localized subsets of data to build up a curve that approximates the original data. The observations (x_i, y_i) are assigned neighborhood weights using the tricube weight function shown in (1)

$$T(u) = \begin{cases} (1 - |u|^3)^3 & \text{if } |u| < 1, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

2.1.2. Median absolute deviation algorithm [7]

The MAD is a robust statistic, being more resilient to outliers in a data set than the standard deviation. In standard deviation, the distances from the mean are squared leading to large deviations being on average weighted more heavily. This means that outliers may significantly influence the value of standard deviation. In the MAD, the magnitude of the distances of a small number of outliers is irrelevant. For a univariate data set X_1, X_2, \dots, X_n , the MAD is defined as the median of the absolute deviations from the median of the data set:

$$MAD = \text{median}_i(|X_i - \text{median}_i(X_i)|), \quad (2)$$

2.1.3. Local regression robust algorithm

The version of Loess described in Section 4.3.2 is vulnerable to outliers that can be caused by leptokurtic or heavy-tailed distributions. To make Loess robust, Cleveland proposed the addition of the robust estimation method bisquare to the least-squares method for fitting a parametric family [6]. This modification transforms Loess into an iterative method. The initial fit is carried out with weights defined using the tricube weight function. The fit is evaluated at the x_i to get the fitted values \hat{y}_i , and the residuals $\hat{e}_i = y_i - \hat{y}_i$. At the next step, each observation (x_i, y_i) is assigned an additional robustness weight r_i , whose value depends on the magnitude of \hat{e}_i . Each observation is assigned the weight $r_i w(x)$, where r_i is defined as in (3).

$$r_i = B\left(\frac{\hat{e}_i}{6s}\right), \quad (3)$$

2.1.4. Interquartile range algorithm [7]

In descriptive statistics, the Interquartile Range (IQR), also called the midspread or middle fifty, is a measure of statistical dispersion. It is equal to the difference between the third and first quartiles: $IQR = Q_3 - Q_1$. Unlike the (total) range, the interquartile range is a robust statistic, having a breakdown point of 25%, and thus, is often preferred to the total range. For a symmetric distribution (i.e., such that the median equals the average of the first and third quartiles), half of the IQR equals the MAD. Using IQR, similarly to (3) the CPU utilization threshold is defined in (4).

$$T_u = 1 - s \cdot IQR, \quad (4)$$

The known algorithm for detecting under-load physical hosts is single-threshold algorithm [8].

2.2. Which virtual machine should migrate?

The second sub-issue, after determining the migration time, if the physical host is under loaded, the total virtual machines should be migrate until physical host is switched off, and if the physical host is over loaded, it should be identify which virtual machine from physical host should be migrate?. Three policy for selecting the virtual machines to migration [7] in over-load condition is RS, MMT and MC algorithms.

2.2.1. Minimal migration time algorithm [9]

The minimum migration time [9] policy migrates a VM v that requires the minimum time to complete a migration relatively to the other VMs allocated to the host. The migration time is estimated as the amount of RAM utilized by the VM divided by the sparse network bandwidth available for the host j [9]. Since the virtual machine with minimum Memory and CPU could migrate faster, so in this policy the small virtual machine are chosen to migration. This policy makes if the more amount of CPU or memory is needed to be free, so a lot of the virtual machines could be migrated.

2.2.2. The random selection policy

The random selection [10] policy selects a VM to be migrated according to a uniformly distributed discrete variable. This policy is suitable for the data centers with large number of virtual machines or in other words can be a good job for a public cloud computing center.

2.2.3. The maximum correlation policy [MC]

The idea is that the higher the correlation between the resource usage by applications running on an oversubscribed server, the higher the probability of the server overloading [9]. This policy is in contrary to the view point of MMT method. In fact, in this policy, instead of migrating multiple small virtual machines, one large virtual machine is migrated. This causes saving time in packing the virtual machines.

2.3. Where virtual machines should be migrated?

The third issue is about where virtual machines should be migrated? After diagnosis migration time and choosing witch virtual machine should be migrated? The third issue determines the destination of each virtual machine. The algorithms are called virtual machine placement algorithm. The large number of virtual

machines and physical makes the idea of using evolutionary algorithms for virtual machine allocation to physical host problem.

Improving energy efficiency has become increasingly important in data centers in recent years. The paper [11] proposed a simulated annealing virtual machine placement algorithm, which is based on simulated annealing theory. Experimental results show that this SA algorithm can generate better results, saving up 25 percentage more energy than First fit decreasing in acceptable time frame.

The paper [12] proposes novel self adaptive particle swarm optimization SAPSO algorithm to solve the intractable nature of the mapping the a set of VM instances onto a set of servers from dynamic resource pool so that the total incremental power drawn upon the mapping is minimal and does not compromise the performance objectives. The experimental results of SAPSO was compared with multi-strategy MEPSO and the result show that SAPSO outperforms the latter for power aware adaptive VM provisioning in a large scale, heterogeneous and dynamic cloud environment.

3. RESULTS AND ANALYSIS

The problem is mapping the virtual machines to physical hosts, so that each virtual machines is allocated to only one physical host and the minimum number of physical hosts are switched on. In other words, consider the number of virtual machines is M and the number of physical hosts is N ($M > N$). V is set of virtual machine which V_i is a sample of virtual machine. Also P is set of physical hosts and P_j represents sample of physical host.

$$V = \{v_1, v_2, \dots, v_m\}$$

$$P = \{p_1, p_2, \dots, p_n\}$$

Lets define:

V_i^{cpu} : the CPU requirement of V_i

V_i^{mem} : the memory requirement of V_i

P_j : a physical machine in P

P_j^{cpu} : the cpu capacity of p_j

P_j^{mem} : the memory capacity of p_j

P_j^{wcpu} : the total CPU workload on p_j

P_j^{wmem} : the total memory workload on p_j

V_{p_j} : the set of virtual machines assigned to physical machine p_j

$$V_{p_j} = \{p_{j1}, p_{j2}, \dots, p_{jm}\}$$

The utilization rate of the CPU in physical server p_j is :

$$\mu_j = P_j^{wcpu} / P_j^{cpu}$$

The energy consumption of physical server p_j when its CPU usage μ_j is:

$$E(p_j) = k_j \cdot e_j^{max} + (1 - k_j) \cdot e_j^{max} \cdot \mu_j$$

When k_j is the fraction of energy consumed when p_j is idle; e_j^{max} is the energy consumption of physical server p_j when it is fully utilized; and μ_j is the CPU utilization of p_j . The purpose of this study allocating physical hosts to each virtual machine according to above Equations, so that the energy consumption is reduced.

4. PROPOSED METHOD

In this section the proposed method is described so that at the first the preliminary description of lottery algorithm is given, then the method for virtual machines to physical hosts with proposed method is proposed.

4.1. Introduction to lottery algorithm

In computing, scheduling is the method by which work specified by some means is assigned to resources that complete the work. The work may be virtual computation elements such as threads, processes or data flows, which are in turn scheduled onto hardware resources such as processors, network links or expansion cards. A scheduler is what carries out the scheduling activity. Schedulers are often implemented so they keep all computer resources busy (as in load balancing), allow multiple users to share system resources effectively, or to achieve a target quality of service. Scheduling is fundamental to computation itself, and an

intrinsic part of the execution model of a computer system; the concept of scheduling makes it possible to have computer multitasking with a single central processing unit (CPU).

A scheduler may aim at one of many goals, for example, maximizing throughput (the total amount of work completed per time unit), minimizing response time (time from work becoming enabled until the first point it begins execution on resources), or minimizing latency (the time between work becoming enabled and its subsequent completion) maximizing fairness (equal CPU time to each process, or more generally appropriate times according to the priority and workload of each process). In practice, these goals often conflict (e.g. throughput versus latency), thus a scheduler will implement a suitable compromise. Preference is given to any one of the concerns mentioned above, depending upon the user's needs and objectives. In real-time environments, such as embedded systems for automatic control in industry (for example robotics), the scheduler also must ensure that processes can meet deadlines; this is crucial for keeping the system stable. Scheduled tasks can also be distributed to remote devices across a network and managed through an administrative back end.

4.2. The proposed method for virtual machine allocation with lottery algorithm

In the proposed method a new method based on lottery algorithm has been proposed for virtual machine allocation to physical hosts. The advantage of proposed algorithm in comparison to previous algorithm is more agility and high speed. In this research a new method based on lottery algorithm and with evolutionary vision has been proposed.

The proposed method steps:

- a. First step: producing N different solutions. The functions of producing initialize solutions have proposed in the following.
- b. Second step: The fitness function is calculated for every single solutions.
- c. Third step: for every solution a ticket is assigned based on the fitness function.
- d. Forth step: one parameter for win rate is used in this algorithm, determines what percentages of solutions moved to the next step. The lottery operation is done in this step and solutions with more tickets has more chance to go to the next step. In this step with notice to win rate, the lottery algorithm repeats and some solutions has been selected for next step. For example if the win rate equals to 70 percent, 70 percent of current solutions are selected to move to the next step and 30 percent of initialize solutions new solutions are created.
- e. Fifth step: The end condition or the number of iterations of algorithm is checked. If the condition is fulfilled the best solution will be chosen otherwise go to the second step.

The problem formulation and production of initialize solutions

As described in the previous sections, the virtual machine allocation is an optimization solution for decreasing energy consumption. The set of virtual machine is as follows:

$$V = \{v_1, v_2, \dots, v_m\}$$

[m presents the total number of virtual machines]

The set of physical hosts is as follows:

$$H = \{H_1, H_2, \dots, H_n\}$$

[n presents the total number of physical hosts]

Some of the restrictions are as follows:

- 1 A virtual machine can only assigned to one physical host.
- 2 For solving the virtual machine allocation to physical hosts, each answer is assumed as a participation in lottery algorithm. As shown in Figure 1, the array index represents of virtual machine's number, and the input number represents the physical host's number which the mentioned virtual machine to be placed on this physical host. In other word if the input number if index i equals to j, means virtual machine[i] is placed on physical host[j]. Sample of solution for proposed algorithm as shown in Table 1.

Table 1. Sample of Solution for Proposed Algorithm

VM1	VM2	VM3	VM _n
Host1	Host2	Host3	Host _n

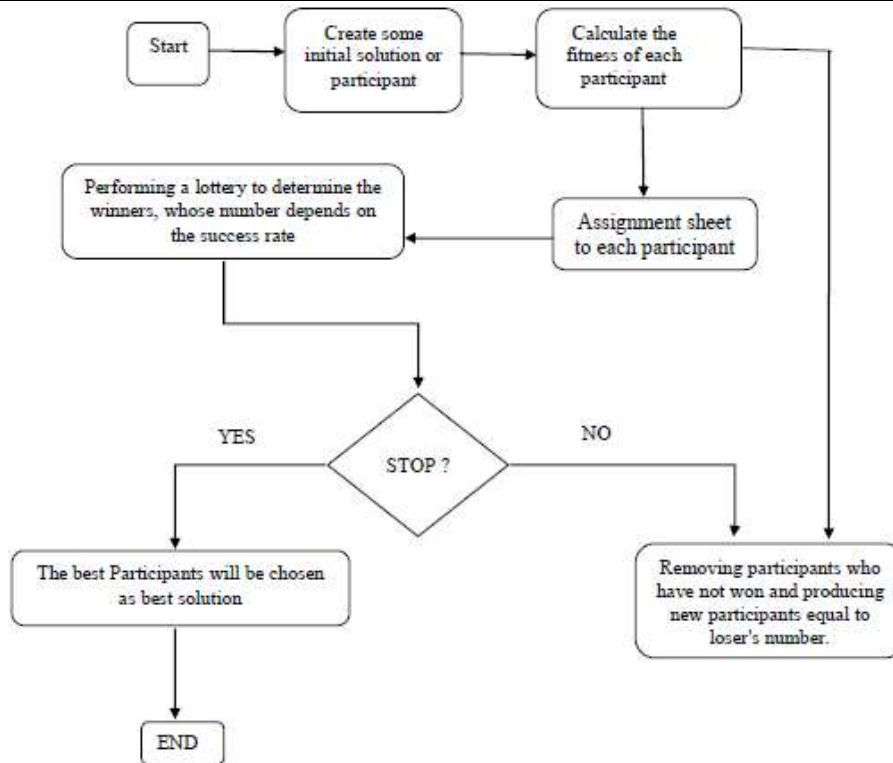


Figure 1. The proposed algorithm's flowchart

5. SIMULATION

The Cloudsim is used to evaluate and analyzing the proposed algorithm's performance. This simulator is a toolkit in java language which is used to simulate cloud environment. The toolbox contains set of several classes, designed by A. Belogazov et al in 2013 [5]. The following scenarios are used for simulating the proposed algorithm.

The simulated data center comprised 800 heterogeneous physical nodes, half of which were HP ProLiant ML110 G4 servers, and the other half consisted of HP ProLiant ML110 G5 servers. The characteristics of the servers and data on their power consumption are given in Section 4.2.2. The frequencies of the servers' CPUs were mapped onto MIPS ratings: 1860 MIPS each core of the HP ProLiant ML110 G5 server, and 2660 MIPS each core of the HP ProLiant ML110 G5 server. Each server had 1 GB/s network bandwidth. In this paper, the proposed method is studied in terms of energy efficiency and the violation of SLA

6. PERFORMANCE EVALUATION

In this study a method for virtual machine allocation to physical hosts has been proposed. As mentioned in the previous sections, there are three sub-issues in a cloud data center, also affect to each other. Four common methods MAD, IQR, LR, LRR on the question of "When a migration should be done?" and three most widely used method MMT, RS, MC for the sub-issue on "which virtual machine should be select for migration" are simulated). The proposed algorithm as a solution for third sub-issue on "where virtual machine should be migrate?" has been proposed. Reducing energy consumption requires to best solutions for each sub-issues. Actually the solution for every sub-issues effects on the final solutions, and it is important that the proposed algorithms is with following of which algorithms. This study analyses the performance of the proposed algorithm with over-load detection algorithms and the virtual machine selecting algorithm. In order to obtain the best solution for optimization of energy consumption the combination of algorithms [the best algorithms for each sub-issues] is important factor.

The Figure 2 shows energy consumption in different combination of algorithms. The vertical axis of diagram shows the energy consumption in w/h and the horizontal axis shows different combinations of methods for each sub-issue. As shows in Figure 1, the minimum amount of energy consumption is for Proposed/LR/MC with 11 w/h. The behavior of three algorithm in combination of VM selection/Host overload detection/Host under-load detection algorithms are a little similar, for example the maximum

amount of energy consumption is related to LRR/RS. In the 12 different combinations, the proposed algorithm performs better than GA and PABFD algorithm. The Figure 2 shows the four points of the most minimum of energy consumptions in Figure 2.

The Figure 3 shows that the proposed algorithm has the minimum amount of energy consumption and the policy of using MC algorithm in virtual machine selection makes the results better. In addition the MC policy has better performance in reducing energy consumption in comparison to RS or MMT policy.

Figure 4 shows the performance of proposed algorithm. GA, PABFD during 10 rounds, in term of the violation of the SLA. The vertical axis shows the violation of SLA in percentage and the horizontal axis shows the algorithms of each sub-issues. As shown in Figure 2 the minimum number of violation of SLA is for LRR/MC for proposed, GA and PABFD algorithms. The proposed algorithm with 0.37 number is in the third place. Among 12 different states, the proposed algorithm performed best in 5 states in comparison on the other algorithms.

Figure 5 shows the violation of SLA for the most minimum numbers of energy consumption methods. As shown in Figure 4 and Figure 5, the violation of SLA decrease by MC policy. As a result of the comparison of Figures 2 and Figure 3, the energy consumption and the violation of SLA are related inversely. The violation of SLA for proposed algorithm is more than the other algorithms but the difference is 0.09 percentage and is very little and can be ignored.

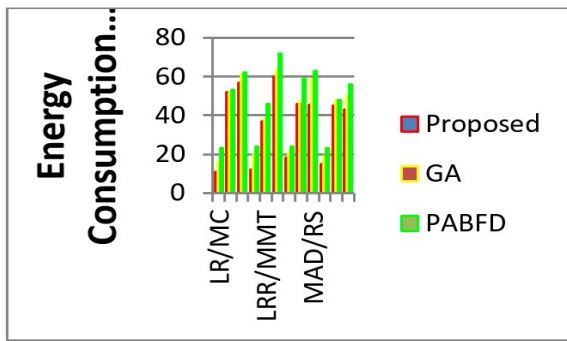


Figure 2. The energy consumption

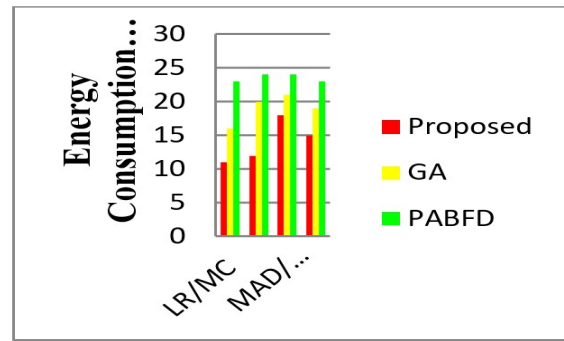


Figure 3. The four minimum points of energy consumption

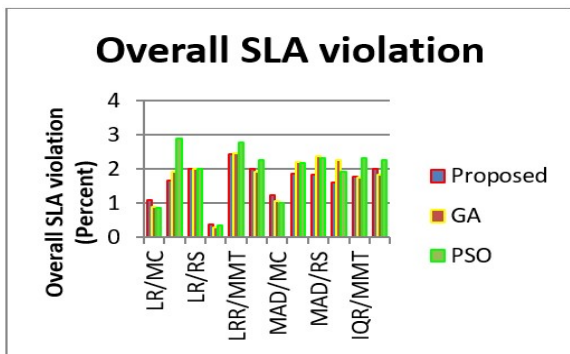


Figure 4 Overall SLA violation

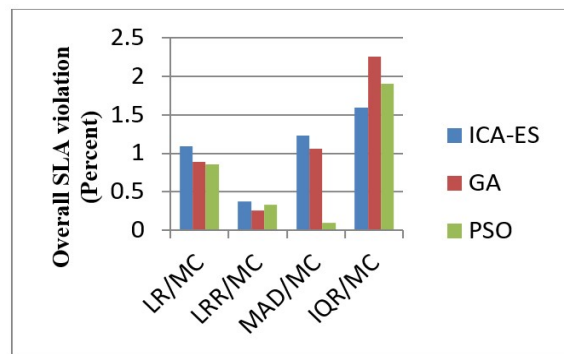


Figure 5. The violation of SLA for the most minimum energy consumption for Scenario C

7. CONCLUSION

As shown in Figures 1-4 the proposed algorithm has the minimum amount of energy consumption in collaborative with LR algorithm for over-load detection algorithm and MC algorithm for selecting the virtual machine, but the minimum amount for the violation of SLA is with collaborative with LRR algorithm for over-load detection algorithm and MC algorithm for selecting the virtual machines. As is clear, the best policies are LR/MC. The proposed algorithm has improved the energy consumption about 31.25 percent.

REFERENCES

- [1] Buyya, Rajkumar, Chee Shin Yeo, and Srikumar Venugopal, "Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering It Services As Computing Utilities," *High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on. IEEE*, 2008.
- [2] Buyya, Rajkumar, Anton Beloglazov, and Jemal Abawajy. "Energy-efficient Management of Data Center Resources for Cloud Computing: A Vision, Architectural Elements, and Open Challenges." arXiv preprint arXiv:1006.0308, 2010.
- [3] Wu, Yongqiang. "Energy Efficient Virtual Machine Placement In Data Centers", 2013.
- [4] Energy-aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud computing Anton Beloglazov, Jemal Abawajy, Rajkumar Buyya.
- [5] Beloglazov, A., "Energy-efficient Management Of Virtual Machines In Data Centers for Cloud Computing," Submitted in Total Fulfilment of the Requirements of the Degree of Doctor of Philosophy, Department of Computing and Information Systems The University Of Melbourne, 2013.
- [6] Guenter, B., Jain, N., & Williams, C., "Managing Cost, Performance, and Reliability Tradeoffs for Energy-aware Server Provisioning," *In INFOCOM, 2011 Proceedings IEEE*, pp. 1332-1340, IEEE, April, 2011.
- [7] Beloglazov, A., & Buyya, R., "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*, 24(13), 1397-1420, 2012.
- [8] Beloglazov, Anton, and Rajkumar Buyya. "Managing Overloaded Hosts for Dynamic Consolidation of Virtual Machines in Cloud Data Centers under Quality of Service Constraints." *IEEE Transactions on Parallel and Distributed System*, vol. 24, no. 7, 1366-1379, 2013.
- [9] Beloglazov, Anton, and Rajkumar Buyya. "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*, vol. 24, no. 13 pp. 1397-1420, 2012.
- [10] Beloglazov, Anton, and Rajkumar Buyya, "Adaptive Threshold-Based Approach for Energy-Efficient Consolidation of Virtual Machines in Cloud Data Centers," *Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science*. Vol. 4, ACM, 2010.
- [11] Wu, Yongqiang, Maolin Tang, and Warren Fraser. "A Simulated Annealing Algorithm for Energy Efficient Virtual Machine Placement." *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2012.
- [12] Jeyarani, Rajarathinam, N. Nagaveni, and R. Vasanth Ram. "Design and Implementation of Adaptive Power-Aware Virtual Machine Provisioner (APA-VMP) Using Swarm Intelligence." *Future Generation Computer Systems*, vol. 28, no. 5, 811-821, 2012.