Research Journal of Applied Sciences, Engineering and Technology 7(14): 2954-2960, 2014
DOI:10.19026/rjaset.7.626
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: August 24, 2013 Accepted: September 12, 2013 Published: April 12, 2014

**Research Article**

# Reservation Resource Technique for Virtual Machine Placement in Cloud Data Centre

Ajith Singh. N and M. Hemalatha
Department of Computer Science, Karpagam University, Coimbatore, Tamil Nadu, India

**Abstract:** Migrations of Virtual Machine directly influence on energy consumption and QoS, to avoid migration of virtual machine when a host is overloaded a good placement technique need to be applied. Virtual Machine Placement is vital in cloud computing to utilize the resources in an efficient manner. Migration of a VM instance when a host is overloaded is familiar in cloud computing. VM selection policy finds a suitable VM to migrate from overloaded host and place to an under loaded host or turn on a new host. While migration there is small downtime of the service, even thou down time is small there is a huge change in energy consumption. Energy consumption in data centre has lead to emission of carbon dioxide to the environment. Frequent VM migration may cause the services to high latency in the network and may disturb the network environment. These works focus to reduce the VM migration, improve SLA and energy consumption. Therefore, a reservation method known as RTBBE (RTBBE (Reservation Technique Bin BECK Entropy) proposed in the study that is by allocating and assigning double upper threshold with entropy method with new overload detection PR (Polynomial Regression) and a VM selection policy MUR (Minimum Utilization Rank) had proposed in this study. The result shows that the proposed technique reduces the energy consumption, SLA and VM migration. Experimental shows that the proposed method reduce the energy up to 21.30 kWh when the overload detection PR combines with MUR, SLA of 0.00029% with IQR with MUR and 775 VM were migrated with LRR and MC.

**Keywords:** Bin BECK entropy, cloud computing, energy, migration, RTBBE, SLA, virtual machine

## INTRODUCTION

Virtualization is a core technology of cloud computing. The problem of Virtual Machine (VM) placement has become a hot topic recently; it is an important approach to improve power efficiency and resource utilization in cloud infrastructures. Virtual Machine placement is a process to allocate VM on the host (physical Machine). Allocating and de-allocating of VM by using different overload detection, which is either over utilized or underutilized? If the hosts are over utilized or underutilized, a VMs selection policy should trigger to select which VMs need to migrated. VM placement requires three parts, first, a proper VM allocation method, second to detect overload of the host and the last part which VMs to be migrated.

Energy issue and emission of carbon dioxide by data centres is also a big issue due to increase of carbon footprint. Roytman *et al*. (2013) pointed out that; average server utilization in traditional data centres is low, estimated between 5 and 15%. An idle server often consumes more than 50% of its peak power (EPA ENERGY STAR Program, 2007), implying that servers at high utilization consume less compared to low utilization consuming significantly more energy. The server may be running low utilization VMs that can be grouped together and put the server in sleep mode.

As the server utilization is very low in which many apps and service run, use of cloud and running multiple OS by using Virtualization, can save wastages on server space and energy (Roytman *et al.*, 2013). A report published in Data Centre Energy Usage Slowing Down (2011) said that after using cloud computing data centre energy usage is slowing down, thanks to cloud computing and Virtualization. In a white paper published by CEET (Centre for Energy-Efferent Telecommunications, 2013) by 2015 data centres, energy consumption will be only 9% where wireless networks will have 90% (The Power of Wireless Cloud, 2013), the increment is due increase of mobile device in which they can exploit cloud services by wireless.

Figure 1, shows different levels of Virtualization, on each Physical Machine (PM) there is a Virtual Machine Monitor (VMM), also called hypervisor that allows many Virtual Machines to share the physical resources (Omar *et al*., 2010). In cloud computing, VM is viewed as boxes and physical server is viewed as a bin. The problem posed is placing the boxes in bins. VM and physical server have their own properties, which are of mainly CPU, memory and bandwidth capacities as properties of the bin.

**Corresponding Author:** M. Hemalatha, Department of Computer Science, Karpagam University, Coimbatore, Tamil Nadu, India

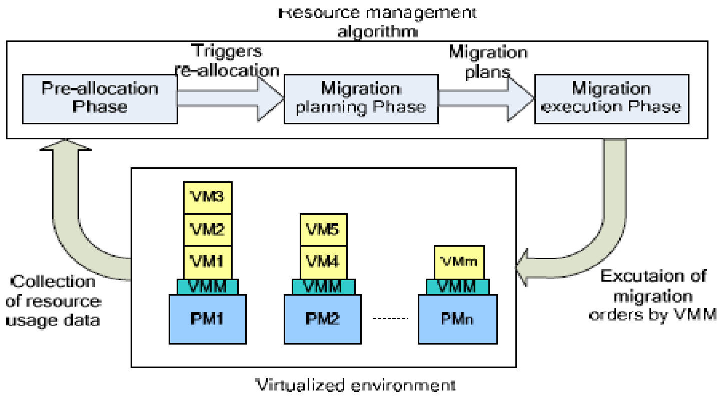


Fig. 1: Architecture of a general live migration based resource manager (Calheiros *et al*., 2009)

The problem with Virtual Machine placement in the data centres is defined as: given a set of virtual machines $VM = \{vm_1, vm_2,..., vm_n\}$ and a set of physical machines $PM = \{pm_1, pm_2,..., pm_m\}$, where each *vmi* is a triplet $vm_i = (cpu_i, ram_i, bw_i)$, $1 \leq i \leq n$ denoted cpu, memory and bandwidth requirements of Virtual Machine respectively. Each *pmj* is also a triplet $pm_j = (cpu_j, ram_j, bw_j)$, $1 \leq j \leq m$ denoted resource capacity of a physical machine. In addition, $x_{ij}$, $1 \leq i \leq m$, $1 \leq j \leq n$ and $y_i$, $1 \leq i \leq m$ are decision variables, $x_{ij} = 1$ if and only if $vm_j$ is mapped onto $pm_i$, $y_i = 1$ if $pm_i$ is used to host virtual machine. The objective is to minimize $\sum_{i=1}^{m} yi$ while finding all values of $x_{ij}$.

There are several implicit constraints in the above definition:

- Each VM can be hosted on only one physical machine.
- For each type of resource, the amounts of resource requests of VM sharing the same physical machine are smaller or equal to the capacity of the physical machine hosting them.
- The number of physical machines that host virtual machines are not more than $m$, $\sum_{j=1}^{m} yi \leq m$ (Yu and Gao, 2012).

The key idea is to reserve the resources while allocating the VM to the host and when the host overloaded with some threshold it can utilize the reserve resources and wait until it reaches another threshold.

## LITERATURE REVIEW

A study by using the game-theoretic method to solve the optimization problem of resource allocation in network systems from the viewpoint of cloud providers done by Wei *et al*. (2010). Cloud computing is based on QoS and cost which is consider by both the provider and user, Author used game theory to solved the problem in which author first used binary integer programming method to obtain initial independent optimization and based on the result an evolutionary mechanism is designed to achieve the final optimal and fair solution (Wei *et al*., 2010).

Wang *et al*. (2010) works on three-level cloud computing network highlight the changes of network bandwidth and hardware technology and uses of low-power host to achieve the high reliability. By keeping the system busy, a process to maintain a load balance by using OLB scheduling algorithm had done by the author. Algorithm plan so that all the host will be working state apart from that LBMM scheduling algorithm is also use to achieve the minimum execution time. Author was mainly focused on task scheduling consider that network in the cloud is in a dynamic nature (Wang *et al*., 2010). A rule based resource manager is proposed by (Rajkamal and Pushpendra, 2012) author highlight of resource provisioning in private cloud which is limited in it, author describe that when need of more resources is needed in private it can be extend with rule based and can be used in the public cloud (Rajkamal and Pushpendra, 2012). Bobroff *et al*. (2007) proposed the algorithm by allocating minimum resources to VMs; by doing this, it could predict the future resource requirement. The algorithm remaps the VM to PM for future resource demand.

Verma *et al*. (2009) propose a $90^{th}$ percentile based provisioning approach. They propose two algorithms:

- Correlation Based Placement (CBP) in which each VM is sized at $90^{th}$ percentile of its peak resource demand. For each VM to placed, it checked first whether it has a positive correlation with any of the VMs that placed in a particular machine.
- **Peak Clustering based Placement (PCP):** With PCP approach each VM is provisioned with the $90^{th}$ percentile utilization value and a peak buffer of capacity equal to the maximum of peak size of all the VMs with considerably low correlation among their peaks of resource demand is kept reserved for all those VMs.

CBP has an obvious disadvantage of ending up using many servers when there are many correlated VMs or applications. Although PCP fixes up this problem but still provisioning resource for each VM individually and presence of a peak buffer, leave much scope for resource wastage.

Chaisiri *et al*. (2009) look on the service provided by cloud provider and proposed an algorithm, which could minimize the cost factor. Algorithm which is based on the stochastic integer programming that works on different stages possibly two stages. On one of it calculates the demand of VMs in reservation phase and another is to calculate the numbers of VMs allocated in both the utilization and on demand phases. Authors try to combine and try to put and design a new algorithm for VM placement that is on Integer Linear Programming Problem (ILP) and could solve the NP-Hard problem (Chaisiri *et al*., 2009).

Optimization for VMs consolidation can be reduced to the Multi Dimensional Bin Packing Problem (MDBPP), MDBPP is a problem of subset selection in which a set object that have different volumes must but packed into the minimum amount of bins of different capacity without exceeding the total capacity of each bin, author try to solve the problem of VM placement which is an NP-hard problem and thus optimal algorithms are not scalable in terms of a number of nodes (due to exponential cost in time and space of finding an optimal solution) which is compulsory in large scale system such as cloud (Armel, 2012; Shaw, 2004). Live migration of Virtual machine which is consider and support only on cloud computing is deeply studied by Clark *et al*. (2005) live migration of virtual machine happen when a host is overload and to balance it live migration happen, author use Xen hypervisor for the experiment and proof that downtime of service is below discernable thresholds. Travostino *et al*. (2006) virtual machines migrate on a WAN area with just 1-2 sec of application downtime through light path. Migration techniques through Remote Direct Memory Access (RDMA) further reduce migration time and application downtime (Lagar-Cavilla *et al*., 2009; Wood *et al*., 2007) proposes two approaches for dynamically map VMs on PMs: a black box approach that relies on system-level metrics only and a grey box approach that takes into account application-level metrics along with a queuing model. VM packing performed through a heuristic that iteratively places the highest-loaded VM on the least-load PM. Some of these mechanisms, for instance prediction mechanisms that can be integrated in our architecture within application-specific local decision modules. Regarding the VM packing problem, we argue that a Constraint Programming approach has many advantages over placement heuristics. Such heuristics are brittle and must return with care if new criteria for VM-to-PM assignment are introduced. Moreover, these heuristics cannot guarantee an optimal.

**Allocation policies:** Allocation of resource in cloud computing specially deals with resource of host the resource can be anything either CPU, Memory, bandwidth etc. According to Anton and Rajkumar (2011) VM allocation divided into two parts:

- Placing the VM to a host based on its requirement which is VM placement or VM consolidation.
- Optimization of current allocations of VMs, which is further divided into 2 part:
    - Select the VM which need to be migrated from overloaded or under loaded host.
    - Placing those VM which is migrated which part 1.

In general, problem of dynamic VM consolidation can be split into 4 sub-problems:

- Deciding when a host is considered to be under loaded, so that all the VMs should be migrated from it and the host should be switched to a low power mode, such as the sleep mode.
- Deciding when a host is considered to be overloaded, so that some VMs should be migrated from the host to other hosts to avoid performance degradation.
- Selecting VMs to migrate from an overloaded host out of the full set of the VMs currently served by the host.
- Placing VMs selected for migration to other active or re-activated hosts (Anton and Rajkumar, 2011; Open Stack, 2011).

## PROPOSED METHODOLOGY

The approach to dynamic VM consolidation proposed in this chapter follows a distributed model, where the problem is divided into 4 sub-problems:

- Host under load detection
- Host overload detection
- VM selection
- VM placement

The proposed method improved the BIN packing with reservation technique and entropy method.

**RTBBE pseudo-code:**

**Step 1:** Placement of VMs to a single host is done by keeping the host reserve with 30% of total resources

**Step 2:** Allocation of VMs to Host is done by using the BIN BECK Entropy Method

**Step 3:** Once the host overloads it scale using reserve resource until it reach 90% to avoid migration

**Step 4:** Applies Overload Detection (Polynomial Regression)
**Step 5:** VM Selection (Minimum Utilization Rank)
**Step 6:** Repeat Step 1

**BIN beck's entropy method:**

- The entropy-based argument which guarantees the existence of proper half-colorings χ is widely termed "Beck's Entropy Method" from the field of discrepancy theory. The discrepancy of a set system $s \subseteq 2^{[n]}$ is defined as:

$$\text{disc}(s) = \min_{\chi:[n]\to\{\pm1\}} \max_{S\subseteq s} |\chi(S)| \quad (1)$$

- Where a bin should be packed exactly with the items in s
- Let us sort the items according to their sizes (i.e., $2/k \geq s1 \geq ... \geq s_n \geq 1k$) and partition the items into groups *I*1,..., It such that the number of incidences in *A* is of order 100 *k* for each group
- In other words, if we abbreviate $vI_j = \sum_{i\in I_j}^{j} Ai$ as the sum of the row vectors in *I j*, then $\|vI j\| \approx 100k$
- Since each column of *A* sums up to at most *k* and each group consumes 100 *k* incidences, we have only $t \leq m/100$ many groups. Now, we can obtain a suitable *y* with at most half the fractional entries by either computing a basic solution to the system:

$$vI_j(x - y) = 0 \; \forall j \in [t],\; 1T\,y = 1T\,x,\; 0 \leq y \leq 1 \quad (2)$$

By applying the Constructive Partial Coloring Lemma to $v_{I1}... v_{It}$ and $v_{obj}$: = (1,...,1)

- With a uniform parameter of $\lambda$: = 0
- In fact, since $(t+1) \cdot e\text{-}02/16 \leq m\ 100 + 1 \leq m/16$
- The meaning of the constraint $vI(x - y) = 0$ is that *y* still contains the right number of slots for items in group *I*. But the constraint does not distinguish between different items within *I*; so maybe *y* covers the smaller items in *I* more often than needed and leaves the larger ones uncovered
- Allocate VM if size <70%
- If overload detection == true allocate 100-30%
- Else allocation size <90
- Migrate VM with selection policy

**Polynomial regression models:**

- In general, we can model the expected value of y as an nth order polynomial, yielding the polynomial regression model:

$$y = a_0 + a_1x + a_2x^2 + \ldots\ldots + a_nx^n + \varepsilon \quad (3)$$

- "Resource scheduler" data table named Vmtable consisting of the variables CPU, Memory, Costs and Power
- To run a polynomial regression model on one or more predictor variables, it is advisable to first centre the variables by subtracting the corresponding mean of each, in order to reduce the inter correlation among the variables
- Suppose we wish to use a second order polynomial model involving the response variable CPU and the predictor variables Memory and Costs. To centre them, the R commands would be:
  - x1 <- VM$ Memory - mean (VM$ Memory)
  - x2 <- VM$ Costs - mean (VM$ Costs)
- Note that we have named the centered variables x1 and x2. We also will need the second order terms for the model:
  - x1sq <- x1^2
  - x2sq <- x2^2
  - x1x2 <- x1 * x2
- The names chosen are, of course, arbitrary. Obviously we could continue with a third order terms and so forth as needed.
- Next, we need to add these new variables to our data table:

VM <- (VM, x1, x2, x1sq, x2sq, x1x2) (4)

Then we can obtain a second order regression model named Poly for these three variables in the usual manner:

VM <- CPU (x1 + x2 + x1sq + x2sq + x1x2) (5)

- x3 <- VM$ Power - mean (VM$ Power)

VM <- Power (x1 + x2 + x1sq + x2sq + x1x2) (6)

- if VM <- COST > Threshold
  - then Overload = True
  - else Overload = false

**Minimum utilization rank:**

- **Failure interval:** This declares a VM failure if there is no heartbeat received for a specified number of seconds (default is 30).
- **Minimum uptime:** After a VM has been powered on, its heart beats are allowed to stabilize for the specified number of seconds. This time should include the guest OS boot time (default is 120).
- **Maximum per-VM resets:** This is the maximum number of failures and automated resets allowed within the time that the maximum resets the time

window specifies. If no window is specified, then once the maximum is reached, automated reset is discontinued for the repeatedly failing VM and further investigation is necessary (default is 3).

- **Maximum resets time window:** This is the amount of time for the specified maximum per-VM resets occur before automated restarts stop (default is one hour):

$$VM's_{rank} = \frac{Maximun\ per - VM\ Resets}{Failure\ Interval * Minimum\ Uptime * Maximum\ Resets\ Time\ Window} \quad (7)$$

## RESULTS AND DISCUSSION

The proposed method RTBBE, the reservation technique with a Bin BECK entropy method to balance the load in cloud data centres, improved the VM placement and reduces the energy. The algorithm proposed in this work reserve the resources in VM placement that reduce the VM migration and Energy. The proposed method first reserve the resource while VM allocation by 30% as reserve resource on the Host, when the host reach 70% of its total resource it allows

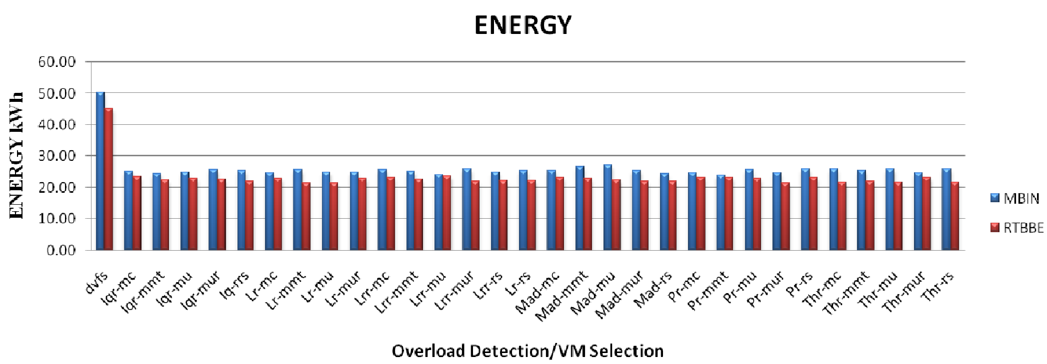


Fig. 2: Energy chart
Overload detection: DVFS- Dynamic voltage frequency scaling, IQR- Inter-quartile range, LR- Local regression, LRR- Robust local regression, MAD- Median absolute deviation, THR- CPU utilization threshold, PR- Polynomial regression; VM selection policy: MC- Maximum correlation, MMT- Minimum migration time, RS- Random selection, MUR- Minimum utilization rank

Table 1: RTBBE with overload detection and VM selection

| Overload detection/ VM selection | Energy kWh | | SLA (%) | | Migration | |
|---|---|---|---|---|---|---|
| VM placement | MBIN | RTBBE | MBIN | RTBBE | MBIN | RTBBE |
| DVFS | 50.34 | 45.15 | 0.00000 | 0.00000 | 0 | 0 |
| IQR-MC | 25.18 | 23.33 | 0.00033 | 0.00033 | 840 | 777 |
| IQR-MMT | 24.28 | 22.41 | 0.00034 | 0.00034 | 847 | 823 |
| IQR-MU | 24.83 | 22.88 | 0.00034 | 0.00031 | 842 | 810 |
| IQR-MUR | 25.58 | 22.59 | 0.00030 | 0.00029 | 813 | 803 |
| IQ-RRS | 25.37 | 21.88 | 0.00035 | 0.00035 | 854 | 811 |
| LR-MC | 24.68 | 22.80 | 0.00038 | 0.00033 | 919 | 837 |
| LR-MMT | 25.65 | 21.44 | 0.00037 | 0.00036 | 844 | 790 |
| LR-MU | 24.84 | 21.48 | 0.00035 | 0.00037 | 814 | 820 |
| LR-MUR | 24.92 | 22.98 | 0.00036 | 0.00030 | 775 | 791 |
| LRR-MC | 25.51 | 23.12 | 0.00034 | 0.00031 | 828 | 775 |
| LRR-MMT | 25.19 | 22.58 | 0.00034 | 0.00030 | 872 | 790 |
| LRR-MU | 24.07 | 23.52 | 0.00037 | 0.00030 | 820 | 787 |
| LRR-MUR | 25.74 | 21.90 | 0.00034 | 0.00038 | 807 | 825 |
| LRR-RS | 24.78 | 22.05 | 0.00036 | 0.00035 | 842 | 849 |
| LR-RS | 25.29 | 22.22 | 0.00037 | 0.00033 | 811 | 851 |
| MAD-MC | 25.28 | 23.03 | 0.00036 | 0.00031 | 840 | 791 |
| MAD-MMT | 26.56 | 22.91 | 0.00031 | 0.00030 | 777 | 766 |
| MAD-MU | 27.15 | 22.34 | 0.00030 | 0.00035 | 786 | 793 |
| MAD-MUR | 25.47 | 21.86 | 0.00032 | 0.00034 | 820 | 812 |
| MAD-RS | 24.37 | 21.93 | 0.00035 | 0.00034 | 840 | 819 |
| PR-MC | 24.68 | 23.24 | 0.00038 | 0.00033 | 882 | 808 |
| PR-MMT | 23.83 | 23.15 | 0.00038 | 0.00032 | 833 | 788 |
| PR-MU | 25.60 | 22.88 | 0.00032 | 0.00032 | 832 | 798 |
| PR-MUR | 24.72 | 21.30 | 0.00036 | 0.00040 | 827 | 848 |
| PR-RS | 25.90 | 23.14 | 0.00035 | 0.00033 | 859 | 861 |
| THR-MC | 25.73 | 21.57 | 0.00033 | 0.00040 | 807 | 879 |
| THR-MMT | 25.36 | 21.93 | 0.00035 | 0.00034 | 821 | 863 |
| THR-MU | 25.94 | 21.56 | 0.00031 | 0.00037 | 810 | 835 |
| THR-MUR | 24.49 | 23.05 | 0.00038 | 0.00033 | 890 | 821 |
| THR-RS | 25.89 | 21.62 | 0.00032 | 0.00037 | 825 | 837 |

Table 2: Comparison value of MBIN and RTBBE

| | Energy kWh | | SLA (%) | | Migration | |
|---|---|---|---|---|---|---|
| RTBBE | PR-MUR | 21.30 | IQR-MUR | 0.00029 | LRR-MC | 775 |
| MBIN | LRR-MU | 24.07 | MAD-MU<br>IQR-MUR | 0.00030 | LR-MUR | 775 |

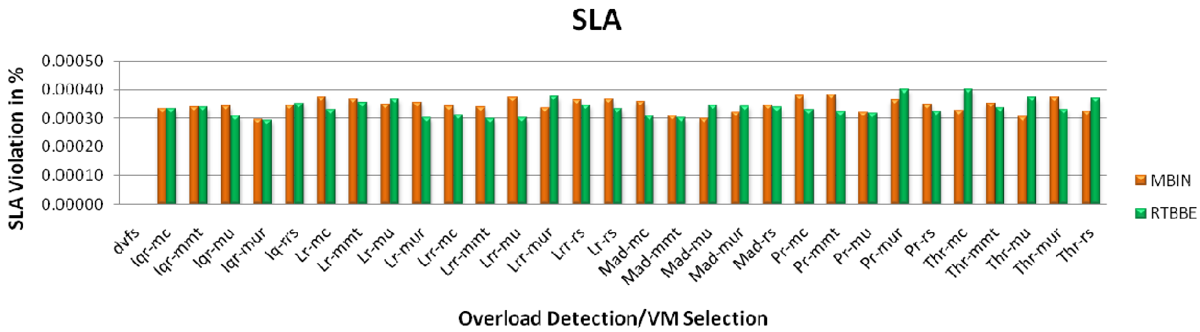


Fig. 3: SLA chart

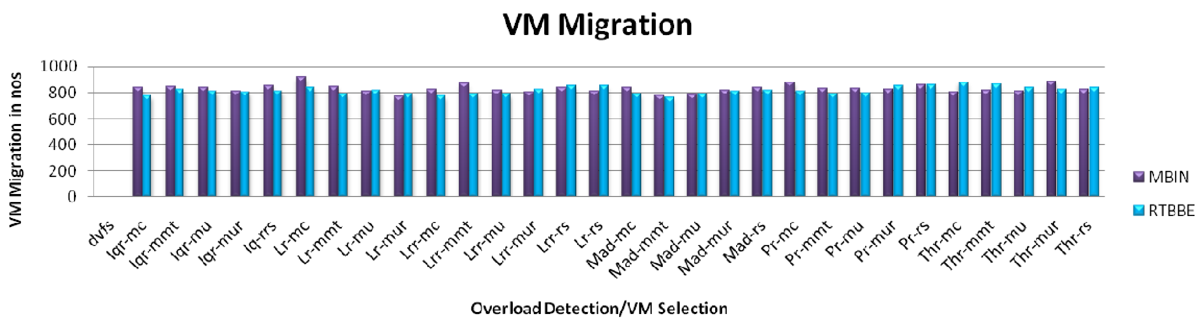


Fig. 4: VM migration chart

the VM to use the free resource of 30% with a threshold of 90%. When the hosts reach 90%, it triggers the overload detection algorithm. Below is the final output of the proposed RTBBE (Fig. 2):

- Energy: RTBBE * PR * MUR = 21.30 kWh
- SLA: RTBBE * IQR * MUR = 0.00029%
- Migration: RTBBE * LRR * MC = 775 VM migrates

Using Planet Workload (Calheiros *et al*., 2009) simulation has been run for 8 h with proposed VM placement, overload detection and VM selection. Table 1 gives a summary of the experimental result of the proposed RTBBE with various existing overload detection and VM selection. Table 2 pointed out the least energy consumption, SLA violation and VM migration. The proposed RTBBE when simulation revealed that PR and MUR is reliable for energy consumption of 21.30 kWh where as when used MBIN 23.83 kWh energy was consumed. SLA of 0.00029 was violated when use of RTBBE and 0.00030 with MBIN SLA violation chart for both the technique is presented in Fig. 3. Seven hundred and seventy five VM was migrated both with RTBBE and MBIN. Further implementing reservation technique with BIN BECK entropy method balance the load, improved the VM placement. The proposed Polynomial Regression (PR) for overload detection and Minimum Utilization Rank (MUR) for VM selection has enchanted the proposed VM placement. Our key idea of reducing energy, SLA and VM as pointed out in this study, strongly recommend deployment and further study of the proposed algorithm towards attaining high QoS in cloud computing and curbing $CO^2$ emission all cost effectively (Fig. 4).

## CONCLUSION

The proposed method RTBBE reduced the energy consumption in data centers, the idea of implementing reservation techniques with entropy method balance the load, improved the VM placement and reduces the energy. The proposed Polynomial Regression (PR) for overload detection and Minimum Utilization Rank (MUR) for VM selection has enchanted the proposed VM placement. The proposed VM placement RTBBE is simulated and compare with existing overload detection and VM selection, the proposed work is simulated with new proposed overload detection and

VM selection. Virtual Machine presents a great opportunity for cloud. Cloud provider has to consider minimizing the cost and the factor related to that is processor, storage, memory, network and power.

## REFERENCES


Anton, B. and B. Rajkumar, 2011. Optimal online deterministic algorithm and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Software: Practice Exp., 41(1): 23-50.

Armel, E., 2012. Energy-aware Distributed Ant Colony Based Virtual Machine Consolidation in IaaS Clouds. Retrieved form: http://dumas.ccsd.cnrs.fr/dumas-00725215.

Bobroff, N., A. Kochut and K. Beaty, 2007. Dynamic placement of virtual machines for managing SLA violations. Proceeding of 10th IEEE International Symposium on Integrated Network Management, pp: 119-128.

Calheiros, R.N., R. Ranjan, C.A.F. De Rose and R. Buyya, 2009. Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services. Technical Report No. GRIDS-TR-2009-1, Grid Computing and Distributed Systems Laboratory, the University of Melbourne, Australia.

CEET White Paper: The Power of Wireless Cloud, 2013. Retrieved form: http://www.ceet.unimelb.edu.au/pdfs/ceet_white_paper_wireless_cloud.pdf.

Chaisiri, S., B.S. Lee and D. Niyato, 2009. Optimal virtual machine placement across multiple cloud providers. Proceeding of IEEE Asia-Pacific Services Computing Conference, pp: 103-110.

Clark, C., K. Fraser, S. Hand, J.G. Hansen, E. Jul, C. Limpach, I. Pratt and A. Warfield, 2005. Live migration of virtual machines. Proceedings of the 2nd Conference on Symposium on Networked Systems Design and Implementation, pp: 273-286.

Data Centre Energy Usage Slowing Down, 2011. Retrieved form: http://www. information-age.com/technology/data-centre–and-it-infrastructure/1644738/data- centre-energy-usage-slowing-down-report.

Lagar-Cavilla, H.A., J. Whitney, A. Scannell, P. Patchin, S. Rumble, E. De Lara, M. Brudno and M. Satyanarayanan, 2009. Snow flock: Rapid virtual machine cloning for cloud computing. Proceedings of Euro Sys09. Nuremberg, Germany.

Omar, A.R., M. Masaharu and A. Kiyoshi, 2010. Live migration-based resource managers for virtualized environments: A survey. Proceeding of 1st International Conference on Cloud Computing, GRIDs and Virtualization, pp: 32-40.

Open Stack, 2011. Open Source Cloud Computing Software. Retrieved form: http:// www. openstack.org/.

PA ENERGY STAR Program, 2007. U.S. environmental protection agency. Report to Congress on a Server and Data Centre Energy Efficiency Public Law Technical Report, pp: 109-431.

Rajkamal, K.G. and K.P. Pushpendra, 2012. A rule-based approach for effective resource provisioning in hybrid cloud environment. Int. J. Comput. Sci. Inform., 1(4): 101-106.

Roytman, A., A. Kansal, S. Govindan, J. Liu and S. Nath, 2013. Algorithm design for performance aware VM consolidation. Technical Report, Microsoft Research.

Shaw, P., 2004. A constraint for bin packing. Proceedings of 10th International Conference on the Principles and Practice of Constraint Programming 2004 (CP-2004), Lecture Notes in Computer Science, 3258: 648-662.

Travostino, F., P. Daspit, L. Gommans, C. Jog, C.D. Laat, J. Mambretti, I. Monga, B.V. Oudenaarde, S. Raghunath and P.Y. Wang, 2006. Seamless live migration of virtual machines over the MAN/WAN. Future Gener. Comp. Sy., 22(8): 901-907.

Verma, A., G. Dasgupta, T.K. Nayak, P. De and R. Kothari, 2009. Server workload analysis for power minimization using consolidation. Proceedings of the 2009 Conference on USENIX Annual Technical Conference (USENIX'09), pp: 28-28.

Wang, S., K. Yan, W. Liao and S. Wang, 2010. Towards a load balancing in a three-level cloud-computing network. Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT), pp: 108-113.

Wei, G., A.V. Vasilakos, Y. Zheng and N. Xiong, 2010, A game-theoretic method of fair resource allocation for cloud computing services. J. Supercomput., 54(2): 252-269.

Wood, T., P. Shenoy, A. Venkataramani and M. Yousif, 2007. Black-box and gray-box: strategies for virtual machine migration. Proceeding of 4th USENIX Symposium on Networked Systems Design and Implementation.

Yu, Y. and Y. Gao, 2012. Constraint programming-based virtual machines placement algorithm in Datacenter. Int. Fed. Info. Proc., 385: 295-304.