International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS)
Volume VII, Issue IX, September 2018 | ISSN 2278-2540

Genetic Algorithm based Optimized Energy Consumption Model for Virtual Machine Consolidation of a Data Center

Francis Okoye, Harmony Nwobodo Nzeribe and Godwin Ozor
Computer Engineering Department, Enugu State University of Science and Technology, Enugu, Nigeria

***Abstract*: - High energy consumption of the cloud computing system of a data center is of great disadvantage to both the cloud service provider and the users or clients. To reduce the energy consumption of the data center, virtualization of the application machines is required to improve server utilization without compromising the quality of service. The paper considered multi objective approach: the migration of the virtual machine, utilization of server resources, distance between virtual machine in the physical machine, sources and destinations characteristics of the server for successful operation with minimal energy. The simulation of the model reveals significant reduction of energy consumption and improved quality of service.**

***Index Terms*: virtual machine, energy consumption, genetic algorithm, distance, cloud computing**

I. INTRODUCTION

Cloud computing has become interesting and popular becauseof its ability to offer utility-oriented IT services over theInternet to world populace. Cloud computing is a paradigm todevelop scalable on-demand virtualized resources based ona pay-as-you-go model [1]. Different types of applications, ranging from conventional to fairly complex scientific, can utilize cloud-based servicesin various forms, including software, hardware, and data. Cloud data centers ideallyallocate resources to users in a way that satisfies the requiredQuality of Service (QoS) determined by the cloud subscribers.Because of the rapid growth of cloud services and their corresponding technologies, cloud infrastructures have become more complicated and complex. Hence, resource management is one of the most prominent issues in modern cloud environments, directly affecting the efficient deployment of cloud services. Modern data centers provide a high level of performance and optimization; however, a new concern is energy consumption.

Balancing the amount of energyconsumption and the performance of a cloud represents a critical issue in the IT industry. Lack of viable optimizing energy model is one of the most significant sources of ever increasing power consumption in these systems. This means that more resources are used in executing applications than the sufficient number of resources and then more power is consumed [2].In cloud computing environments, greater power savingcan be achieved by using consolidation or efficient scheduling of customer applications. In consolidation, the cloud tries to use fewer servers or physical machines in order to execute applications without breaking the Service Level Agreement (SLA) [3]. So, the cloud selects some servers to be turned off and their virtual machines or applications are migrated or transferred to other servers where they can complete execution.In addition, scheduling of applications to virtual machines and scheduling of virtual machines to servers can play important role in reducing energy consumedin cloud environments [4]. The propose model considered the maximum distance or range between the source server and the destination server for effective communication but less energy dissipated.

II. RELATED LITERATURE

Mohamed Amoon (2018) in his research suggested that the source and destination awareness will mitigate the energy sapping of the system. The algorithm was robust for utilization but could not address the distance between the source and destination which also takes much energy.

Based on set points for checking CPU utilization, Beloglazov*et al.* [6] proposed a consolidation mechanism. Inthe case that CPU utilization increases more than the upperthreshold or drops lower than the lower threshold, some VMswill be selected and migrated over to other hosts. For thisreason, the authors performed an experiment to determinethe best set of upper and lower thresholds to reduce powerconsumption while keeping SLA violations low. Also, theauthors suggested using three VM selection policies. The first policy is Minimization of Migration (MM). The MMpolicy consolidates the least number of VMs to other hostsso that CPU utilization goes below the upper threshold. TheRandom Choice policy (RC) selects the VMs randomly. TheHighest Potential Growth policy (HPG) selects the VMs thathave the lowest CPU utilization relative to their total requiredCPU capacity. The results showed that the MM policy witha lower threshold set to 30% and an upper threshold setto 70% provides the best results. This study showed goodresults in both energy consumption and SLA violation rate; however, it has two weaknesses. First, the power consumptionmodel only considers CPU power usage. Second, the consolidation mechanism is static because the thresholdsare defined as fixed values, and this issue reduces the scalability of the approach for various workloads. However, ourproposed mechanism considers the power consumption of allthe server's components.

Dynamic and fuzzy operations of systems could to work with Boolean of fixed point, hence, Beloglazov and Buyya [7] proposed a dynamic VM consolidation mechanism forreducing both energy consumption and SLA violations incloud data centers. This mechanism uses historical data ofresource utilization for determining the adaptive thresholdsfor each server. The Median Absolute Deviation (MAD), Local Regression (LR) and Interquartile Range (IQR) policies are introduced for determining the dynamic upper thresholds. The LR policy is based on the Loess method andaims to determine the upper threshold by finding a regression curve that approximates future data. Moreover, theauthors presented two VM selection policies. The Mini-mum Migration Time (MMT) policy selects the VMs withthe least time needed for migration, and the MaximumCorrelation policy (MC) selects a VM that have the highest correlation of the CPU utilization with other VMs.This mechanism considers the power consumptionof all server components; however, the CPU is the onlyconsidered factor for consolidating the VMs, which is aweakness.

Taheri and Zamanifar [8] introduced a two-phase VMconsolidation mechanism to cope with the problem of incomplete migrations. Perplex VMs are the VMs that should beconsolidated but that have no place in other hosts. Therefore,the system terminates the migration and replaces the VMs tothe prior place. This issue leads to a waste of CPU capacityand power and increases the network's overhead. Based onthe proposed framework, in the first phase, VMs from the over-utilized hosts migrate to other hosts and then in thesecond phase, VMs from underutilized hosts are sent to otherhosts.

All the reviewed work above contributed to a large extent to minimize energy consumption in data centers but areas like topology and control switch strength were not adequately worked on. The process architecture of the data center is as shown in Fig 1.

III. OPTIMIZED ENERGY CONSUMPTION (OEC) MODEL

The optimized energy consumption (OEC) model considered the communication between virtual machines of both homogeneous and heterogeneous in nature. The model also considered the communication between physical machine (PM) and virtual machine (VM). The process that led to migration of VMs from source to destination was also investigated to underpin the utilization of the server resources. The OEC model was structured as outlined below.

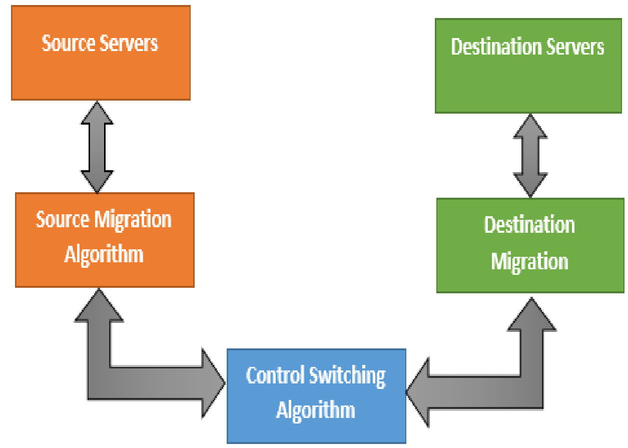


Fig 1: Data Center Migration Process

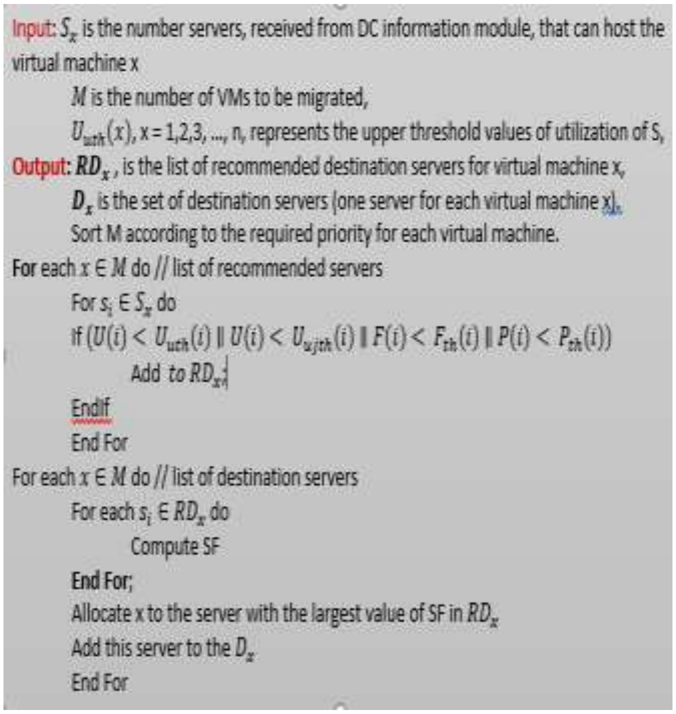
A. *Source and Destination Host Performance Evaluation.*

A mechanism for accessing or evaluating the working characteristic of the source servers and the destination servers was designed and the algorithm as developed in fig 2 for source and fig 3 for destination.

Input: S is the number of turned on servers in data center z,
Q is the number of VMs in the server in the data center,
$U(x)$, x = 1,2,3 ..., n represents the utilizations of S,
$U_{ith}(x)$, x = 1,2,3, ..., n represents the lower threshold values of utilization of S,
$U_{jth}(x)$, x = 1, 2, 3, ..., n represents the distance between VMs,
$F(x)$, x = 1,2,3 ..., n represents the failure probabilities of S,
$F_{th}(x)$, x = 1,2 3, ..., n, represents the threshold values of the failure probability of S,
$P(x)$ P(x), x = 1,2,3 ..., n represents the average power consumption of S,
$P_{th}(x)$, x = 1,2,3, ..., n, represents the threshold values of average power consumption of S,
$T(x)$, x = 1,2,3 ..., n, represents the response time of s,
$T_{th}(x)$, x = 1,2,3..., n, represents the threshold values of the response time of S,
Output: $SS \subset S$, is the set of source servers,
M, is the set of VMs to be migrated from source servers of d,
For each $s_i \in S$ do
If $(U(i) < U_{ith}(i) \parallel U(i) < U_{jth}(i) \parallel F(i) > F_{th}(i) \parallel P(i) > P_{th}(i) \parallel T(i) > T_{th}(i))$
{
Add s_i to SS;// add the server to the set of source servers
For each $v_{ij} \in V_i$ do // add the set of VMs to the migration list;
Add v_{ij} to M;
End For;
}
End IF
Shut down $\boldsymbol{s_i}$
Remove $\boldsymbol{s_i}$ from S;
End For

Fig 2: Source algorithm

According to the performance requirement, the Monitoring module prepares a set of source servers to be turned off and a migration list that contains virtual machines to be migrated. The migration list will be handed over to the control switching unit. The control unit has embedded logics or interactive interface that communicate both the source and destination servers. The source algorithm selects virtual machines from servers that have performance under threshold values of failure probability, utilization and power consumption.



Input: S_x is the number servers, received from DC information module, that can host the virtual machine x
M is the number of VMs to be migrated,
$U_{uth}(x)$, x = 1,2,3, ..., n, represents the upper threshold values of utilization of S,
Output: RD_x , is the list of recommended destination servers for virtual machine x,
D_x is the set of destination servers (one server for each virtual machine x).
Sort M according to the required priority for each virtual machine.
For each $x \in M$ do // list of recommended servers
For $s_i \in S_x$ do
If $(U(i) < U_{uth}(i) \parallel U(i) < U_{ujth}(i) \parallel F(i) < F_{th}(i) \parallel P(i) < P_{th}(i))$
Add $to\ RD_x$;
EndIf
End For
For each $x \in M$ do // list of destination servers
For each $s_i \in RD_x$ do
Compute SF
End For;
Allocate x to the server with the largest value of SF in RD_x
Add this server to the D_x
End For

Fig 3: Destination algorithm

For each virtual machine, the Migration module asks the control module for a set of servers that can host the virtual machine according to the customer's requirements. The control module supplies the migration module. The destination algorithm with a list of servers for each virtual machine in the migration list. Based on this servers list, the Migration module determines the list of recommended servers for each virtual machine. The set of servers in the recommended list should satisfy the threshold values of power consumption rate, failure probability, distance and utilization. Then, the most suitable destination server is selected from the recommended list according to the policy defined by the destination algorithm. The selected server is added to the list of destination servers.

In order to achieve that, the destination algorithm computes the value of Selection Factor (SF) for each server in the set of recommended servers and uses this value to select the most suitable destination server for each virtual machine in the migration list. The Selection Factor (SF) of a server r is defined by:

$$SF_r = \sum_{i=0}^{n} A_{ri} W_{ri} \tag{1}$$

where A_{ri} represents a criterion partially used to determine the value of the Selection Factor of server r. The criteria used may include server utilization, failure probability, response time, power consumption rate, monetary cost rate, load balancing. In accordance to their needs, the provider and the customer determine the value of each related criterion.

For example, the provider determines the values of failure probability and utilization criteria, while the customer determines the values of response time and monetary cost criteria. The value W_{ri} is the corresponding weight for eachcriterion.

B. The Distance Between The Virtual Machines

The distance between the source and destination of the virtual machines host is a parameter to consider. Every transmission of signals involves different levels of energy dissipation, hence shorter the length the better the energy consumption of the system. Fig 4 x-rayed the distance between virtual machines which is energy dependent. Take for instance, the energy expelled when VM4 is to be migrated to the host of VM2 will be quite significantly different as compared to migration from VM2 to VM1.

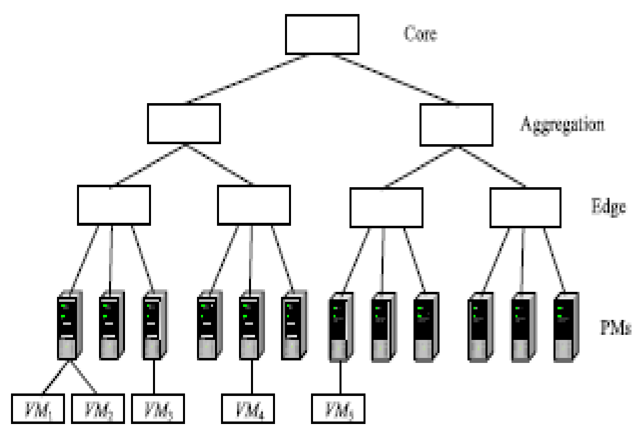


Fig 4: Virtual Machines (VMs) distance with reference to Physical Machines (PMs)

IV. GENETIC ALGORITHM

Developed from the evolution law in the ecological world, the Genetic algorithm is a random searching method. After the first population is generated, it evolves better and better approximate solutions using the law of survival of the fittest from the generations. An individual is chosen in every generation based on the fitness of different individuals in certain problem domains. A new population representing a new solution set is produced when different individuals combine, cross, and vary by genetic operators in natural genetics. The paper proposed an optimized energy model through the genetic algorithm based on the real situation of cloud computing. In an ordinary world, the productivity of any individual depends on the fitness value which shows the number of decedents it will have. The fitness function is the measure for the superiority of an individual in the entire population. The fitness value shows the performance ofan individual. If the fitness is large, then the performance of an individual is better. Depending on the fitness function value, the individuals are determined to survive or die out. Hence, the fitness function is the motivating factor in the genetic algorithm. The fitness function of this work considered the minimum distance between the source and destination host, the utilization set points and bandwidth of the channel.

$$F = \frac{1}{Dn + Y} \quad (2)$$

Where *F* is the fitness function of the system, *D* is the distance between the source and destination, *n* number of virtual machines that are to be migrated and *Y* is the utilization index.

V. RESULTS AND ANALYSIS

The VMs allocation in the data center with a homogeneous configuration of the PMs was considered and genetic algorithm was proposed to solve this problem. The VM allocation on the physical servers of the IT infrastructure relates to the problem of consolidation of computing resources. The case of using homogeneous PM configurations is chosen because it can be implemented within a single cluster, which may be considered as a unit of control in a data center. The mathematical model of the VMs allocation on the PMs is represented as follows.

$$r_{ji} = \begin{cases} 1 & if\ VM\ Kj\ is\ allocated\ \ on\ PM\ Ni \\ 0 & otherwise \end{cases} \quad (3)$$

The matrix *R* is a solution to the problem and determines the allocation of *K* VMs on the set *N* of PMs. The authors consider that all PMs in set *N* have identical specifications and, consequently, the same computing resources.

Then the optimum criterion for solving the problem of VM placement on PMs will be

$$min \sum_{i=1}^{n} yi \quad (4)$$

that is the PMs should be filled with VMs so that the minimum number of PMs are involved. When the criterion (4) is satisfied the total cost S of the data center and PMs maintenance and energy supply will be minimized.

The objective function can be represented as follows:

$$S = \sum_{i=1}^{n} s_i y_i \quad (5)$$

where *si* is the maintenance and energy supply costs for the *i* th PM. In the case when the PMs in the data center have identical specifications (i.e., homogeneous), the expression (5) becomes

$$S = s \sum_{i=1}^{n} y_i \quad (6)$$

where *s* is the maintenance and energy costs per PM.

Considering the previous description, the problem of *K* VMs allocation can be summarized as follows: it is necessary to place the VMs on data center PMs so that either the expression (5) or (6) reaches a minimum value. The authors consider two cases, namely, the initial placement of the VMs and also their change in placement during the execution. The algorithm restarts when unused resources are detected in PM. If the number of unused resources on a PM is greater than the threshold, then that server is added to a consolidation list. The algorithm restarts if the total number of unused resources on the physical servers, included in the consolidation list, exceeds the resources of one PM. As a result, it is proposed to run the genetic algorithm not for all PMs, but for PMs in the consolidation list. All migrations initiated by the genetic algorithm at the previous stage must be completed. Simulation studies were performed on the data center resource allocation problem solution using two optimization tools: the genetic algorithm (GA), the particle swarm optimization (PSO) and conventional data center setup.

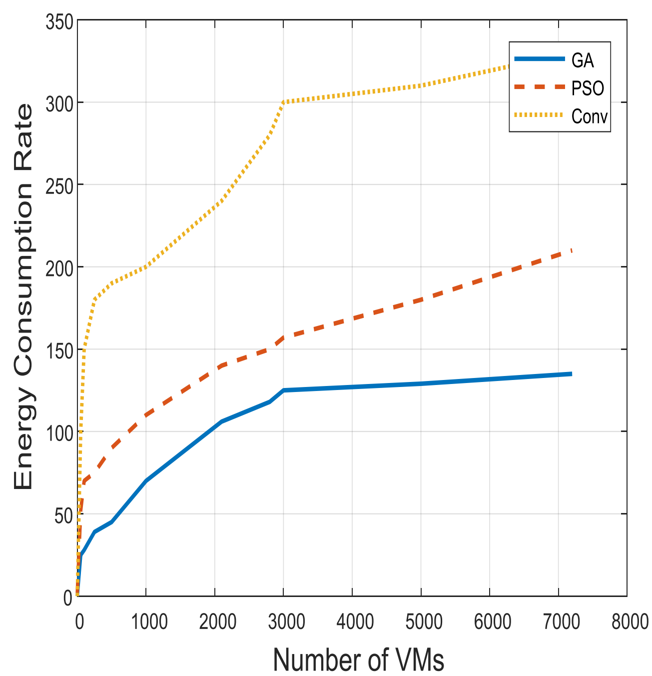


Fig 5: Virtual machines vs its energy consumption rate

It is now established that the genetic algorithm performs better than other optimization tested in this paper. it can be inferred from fig 5 that energy consumption rate stays fairly constant irrespective of the increase in the number of virtual machines.

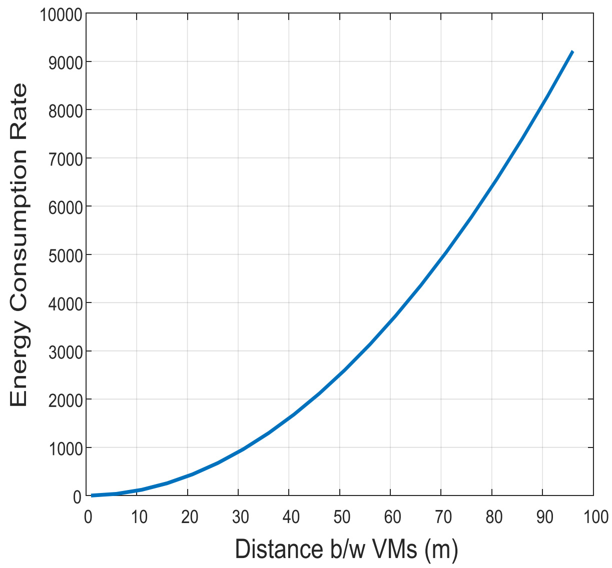


Fig 6: Distance of the VMs and energy consumption effect.

Upon implementation of the algorithm, energy consumption of the system was minimized based the distance of the virtual machine as shown in fig 6.

VI. CONCLUSION AND RECOMMENDATION

Allocation of virtual machines on physical servers is a huge challenge in the cloud computing systems. Considering the large number of types and virtual machine settings that modern service providers offer, as well as the high density of virtual machines per physical server the solution for this problem is complicated. In this paper, the virtual machine allocation problem is solved by using the genetic algorithm. The energy consumption of the system was also minimized to without compromising quality of service. Distance or position of the physical machines that host the virtual machine was a key to plan and optimize the energy consumption. The area of bandwidth and signal relay was not discussed here, hence a vital area to explore.

REFERENCES

[1]. Amoon A. “A multi-criteria-based approach to save electrical power in cloud data centers.” DOI 10/1109/ACCESS.2018.2830183, IEEE, Vol 6, pp 24110-24117, May 2018.

[2]. W. Zhu, Y. Zhuang, and L. Zhang, ``A three-dimensional virtual resourcescheduling method for energy saving in cloud computing," *Future Generat. Comput. Syst.*, vol. 69, pp. 66_74, Apr. 2017.

[3]. Y. Gao, H. Guan, Z. Qi, T. Song, F. Huan, and L. Liu, ``Service level agreementbased energy-efficient resource management in cloud data centers,"*Comput. Elect. Eng.*, vol. 40, pp. 1621_1633, Jul. 2013.

[4]. D. Buia, Y. Yoon, E.-N. Huh, S. Jun, and S. Lee, ``Energy efficiency forcloud computing system based on predictive optimization," *J. ParallelDistrib. Comput.*, vol. 102, pp. 103_114, Apr. 2017

[5]. A. Beloglazov, J. Abawajy, and R. Buyya, ``Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing," *Future Generat. Comput. Syst.*, vol. 28, no. 5, pp. 755_768, 2012.

[6]. A. Beloglazov and R. Buyya, ``Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," *ConcurrencyComput., Pract. Exper.*, vol. 24, no. 13, pp. 1397_1420, 2012.

[7]. M. M. Taheri and K. Zamanifar, ``2-phase optimization method for energy aware scheduling of virtual machines in cloud data centers," in *Proc.IEEE Int. Conf. Internet Technol. Secured Trans. (ICITST)*, Dec. 2011, pp. 525_530.