International Conference on
RESEARCH IN ELECTRICAL AND COMPUTER ENGINEERING
15 December 2016 Singapore
www.receconf.com

Optimizing Energy and parallel tasks using task consolidation in clouds

Arezo kalhor
Azad university of Qazvin
arezo_kalhor@yahoo.ccom

Esmaeil zeinali khosraghi
Azad university of Qazvin
zeinali_es@yahoo.com

Abstract

In the future, cloud computing will become an important technology in the development of Internet services. Due to the growing demands of cloud infrastructure, energy consumption in data centers has increased that this is a significant problem. High energy consumption not only leads to increase operating costs and decrease profits providers but emissions of carbon lead to not be good for the environment. Much research has suggested data centers with high energy efficiency and techniques use such as virtualization and task consolidation. This approach leads to significantly reduced energy consumption of data centers.

Task consolidation way to increase the use of cloud computing resources. Increasing use provide benefits such as construction IT service provides and quality assurance services. The main purpose remains to maximize resource utilization and ultimately to minimize energy consumption. In this study, we tried to integrate task consolidation algorithms of ETC and ECTC energy consumption and total execution tasks. Simulation results show that energy consumption of EPTC algorithm rather than Bi-objective algorithm 30% and time to complete tasks 28% improved.

Keywords: cloud computing; task consolidation; resource utilization; energy consumption; EPTC;

Introduction

The cloud computing is created in 2007 and it is still a young subject because of its quality in flexible dynamic IT infrastructures, Quality of Service (QoS) and the ability of configuring software services[1]. Resources in these systems can be widely distributed and the scale of resources involved can range from several servers to an entire data center [2]. As well as, Cloud led to the establishment of large data centers. These data center consumed energy and consequently the carbon emission publish in the world [3].

To integrate and make good use of resources at various scales, cloud computing needs efficient methods to manage them. Consequently, the focus of much research in recent years has been on how to utilize resources and how to reduce power consumption. [2].

When the resources in cloud environment are bigger than before, it will lead greater energy consumption in data center. An average data center consumes as much energy as 25,000 households. As energy costs are increasing while availability shrinks, there is a need to shift the focus from optimizing data center resource management for pure performance to optimizing them for energy conservation, while maintaining high service level performance [4].

Recent studies identified that server energy consumption scales linearly with (processor) resource utilization. This encouraging fact further advocates the significant contribution of task consolidation to the reduction in energy consumption [5]. The task consolidation is also known as workload consolidation problem which is the process of assigning set of tasks to set of resources without violating time constraints [6]. There are many algorithms in task consolidation. These algorithms contribute to reduce energy consumption in cloud data centers. So we decided to work this subject.

The remainder of the paper is organized as follows. Section II describes energy models, the task consolidation problem in this paper. Section III overviews related work about task consolidation algorithms. Proposed algorithm has presented in Section IV. We tried to show evaluation of proposed algorithm in section V. finally, the conclusion and future work have proposed in section VI.

MODELS

cloud model

The target system consists of a set R of r resources/processors. These are fully interconnected in the sense that a route exists between any two individual resources (Fig. 1). The resources are assumed homogeneous in terms of computing capability and capacity. The aforementioned is achieved through the virtualization technologies. Nowadays, as many core processors and virtualization tools (e.g., Linux KVM, VMware Workstation & VMware Fusion, Xen, Parallels Desktop for Mac, Virtual Box) are commonplace. The number of concurrent tasks on a single physical resource is loosely bounded. In general, a cloud computing can span across multiple geographical locations [6] (i.e., distributed), but for simplicity, the cloud model in our study is assumed to be confined to a particular physical location. The inter-processor communications are assumed to perform with the same speed on all links without substantial contentions. In this model, a message can be transmitted from one resource to another while a task is being executed on the recipient resource, which is possible in many systems [5, 6, 7].

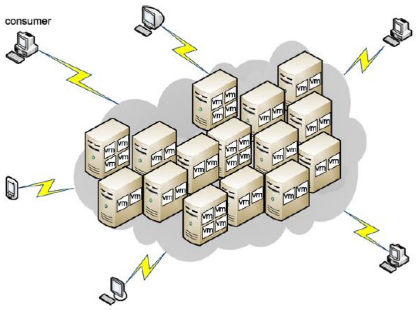

Figure 1: Cloud Model [7]

Application Model

Services offered by cloud providers can be classified into software as a service (SaaS), platform as a service (PaaS) and infrastructure as a service (IaaS). Note that, when instances of these services are running, they can be regarded as computational tasks or simply tasks. IaaS requests are typically tied with predetermined time frames (e.g., pay-per-hour), but requests of SaaS and PaaS are often not strongly tied with a fixed amount of time (e.g., pay-per-use). Therefore, service requests for SaaS and PaaS can be estimated based on historical data and/or consumer supplied service information. In our study, Service requests arrive in a Poisson process and the requested processing time follows exponential distribution. CPU utilization of each service request can be identifiable and disk and memory use correlates with processor utilization. Hereafter, application, task and service are used interchangeably [5, 6, and 7].

Energy model

Our energy model is assumed that processor utilization has a linear relationship with energy consumption. In other words, for a particular task, the information on its processing time and processor utilization is sufficient to measure the energy consumption for that task. The utilization Ui in resource ri *at* any given time is defined as*:*

$$U_i = \sum_{j=0}^{n-1} u_{i,j} \tag{1}$$

Where *n* is the number of tasks running at that time and ui, j is the resource usage of a task tj.
At any given time, the energy consumption Ei *of* a resource *ri* is defined as:

$$E_i = (p_{max} - p_{min}) \times U_i + p_{min} \tag{2}$$

pmin is the minimum power consumption in the active mode (or as low as 1% utilization) and pmax is the power consumption at the peak load (or 100% utilization).
The total utilization (UR) the total energy consumption (ER) of the system at any given time are defined as

$$U_R = \sum_{i=0}^{m-1} U_i \text{ and } E_R = \sum_{i=0}^{m-1} E_i \tag{3}$$

In this equation, m is the number of resources used. The resources in the underlying system are assumed to be incorporated with an effective power-saving mechanism for idle time slots. The mechanism results from the significant difference in energy consumption, between active and idle resources states. Specifically, at any given time, the energy consumption of an idle resource is set to 10% of pmin. Because the overhead to turn off and back on a resource takes a no negligible amount of time, the option for idle [5, 6, 7].

Task consolidation problem

The task consolidation is also known as workload consolidation problem. It is assumed that a set T= {t0**…** tn-1} of n tasks to a set R = {r0**…** rm-1} computing resources is assigned, without violating time constraints. This idea have two purposes such as maximize resource utilization and ultimately to minimize energy consumption. Time constraints are directly related to the resource usage associated with the tasks [2, 7]. Task consolidation aims at effective usage of cloud resources by consolidating a set of tasks into a small number of virtual machines. Effective resource usage has advantage for reduction monetary cost by reducing: (a) amount of virtual machines; (b) labors required to maintain virtual machines; (c) floor space; and (d) energy consumption [8].

RELATED WORK

In this section, we want to introduce several algorithms of task consolidation. One the algorithm is ECTC. This approach computes the energy consumption of a given task on a selected resource. The cost function of this algorithm is designed to encourage resource sharing [8]. The cost function of ECTC, computes the actual energy consumption of the current task by subtracting the minimum energy consumption (pmin) required to run a task, if other tasks would be running in parallel with that task. In other words, the energy consumption of the overlapping time period among the running tasks and the current task (tj) is explicitly taken into account. The cost function tends to discriminate the task being executed in a standalone mode[1,2].

ECTC has a cost function called fi,j. For a task tj on a resource ri, we can calculate value fi,j as follows:

$$f_{i,j} = [(p\Delta \times u_j + p_{min}) \times \tau_0] - [(p\Delta \times u_j + p_{min}) \times \tau_1 + (p\Delta \times u_j \times \tau_2)] \tag{4}$$

$p\Delta$ is the difference between pmax and pmin, uj is the utilization rate of tj. $\tau 0$, $\tau 1$ and $\tau 2$ is total processing time of tj. $\tau 1$ and $\tau 2$ are The time period tj is running stand alone and that tj is running in parallel with one or more tasks, respectively[6,7].

However, this algorithm does not take into consideration other resources which have influence in energy consumption such as disk utilization, memory utilization, network bandwidth usage, application software. It tends to increase number of virtual machines within the cluster. As well as, it does not consider the migration of task from one virtual cluster to another [8].

The authors in [6, 7], the cloud computing system utilizes virtualization technologies where tasks can be easily consolidated which is an effective method to increase resource utilization and in turn reduces energy consumption. They proposed task consolidation Maximum rate Utilization (MaxUtil). MaxUtil looks after the more energy efficient resources in terms of resource utilization. The energy model is devised on the basis that processor utilization has a linear relationship with energy consumption[8].

The task consolidation problem is modeled as a bin packing problem where virtual machines represent bins, tasks are objects to be packed in the bins and CPU resource utilization represents bin dimension. They proposed 100% resource utilization rule for virtual machines and makes task consolidation decision based on the resource utilization which is a key indicator for energy efficiency. The advantage of this method is to reduce number of virtual machines in the cluster. So helped to minimize energy consumption in the cloud computing system [8].

The MaxUtil cost function is derived with the average utilization during the processing time of the current task, as core component. The cost function aims to increase consolidation density and has a double benefit[6,7]:

1. Implicit reduction of the energy consumption is directly related
2. Decreased number of active resources

In others words, MaxUtil tends to intensify the utilization of a small number of resources. MaxUtil has a cost function called fi,j. For a task tj on a resource ri, we can calculate value fi,j as follows:

$$f_{i,j} = \frac{\sum_{\tau=1}^{\tau_0} U_i}{\tau_0} \tag{5}$$

Which is the utilization of a resource ri, divided by total execution time (t0) of task tj [1, 2].
However, the deficiencies and disadvantages of the MaxUtil algorithm are: (a) it assumes energy consumption is linear to resource utilization focused on CPU usage without considering other resources utilization such as disk, network bandwidth, memory etc. (b) the incorporation of task migration increased energy consumption because migrated tasks tend to be with short remaining processing time and these tasks are most likely to hinder the consolidation of a new arriving task (c) the migration of task from one virtual cluster to another is not modeled[8].
Energy-aware task consolidation (ETC) technique optimize energy consumption of virtual clusters. The CPU and network utilization are taken into account when dealing with task migration in the virtual clusters. The authors proposed 70% CPU utilization principle to manage task consolidation among virtual clusters in the cloud computing system. When the CPU utilization in a virtual cluster is above

70%, the task is migrated to another cluster. If there are multiple virtual machines in the clusters available to receive the tasks, the one with minimal energy cost is chosen to perform task consolidation[8].

ETC has a cost function called costi,j. For a task tj on a resource ri, we can calculate value costi,j as follows:

$$\mathbf{Cost_{ij}} = \sum_{t=T_{1j}}^{T_{1j}+T_{pj}} E_t(V_i) + \frac{DS}{BW_{P0}} \times 2\beta\ w/s \quad (6)$$

The main idea of ETC is to consolidate tasks and to keep the CPU utilization of virtual machines under the specified CPU Utilization Threshold (CUT). Given a cloud system composed of multiple virtual clusters (VC), for example, three VCs A, B and C (denoted by VCA, VCB and VCC), the task consolidation strategy within a virtual cluster (for example, VCA) can be described as follows[2]:

1. The scheduler of VCA dispatches task tj to a VM. If more than one VM is available, an appropriate VM is selected based on the best-fit strategy.
2. If there is no VM available, and Vi is below the specified CPU utilization threshold, VCA asks for resource support from other VCs, e.g., VCB or VCC.
3. If both VCB and VCC can provide VMs that run below the specified CPU utilization threshold, then Eq. (6) is used to select the VM from the VC that consumes the least amount of energy when transmitting and executing the task.
4. If none of the VCs can provide a VM below the specified CPU utilization threshold, then tj is assigned to the Vi that consumes the least amount of energy locally (i.e., VCA).

The task consolidation strategy uses the best-fit strategy to optimize resource utilization. The best-fit strategy achieves this by migrating tasks to whichever VM will most closely approach the target CPU utilization threshold. The CPU utilization threshold depends on hardware architecture and may differ on different cloud systems.

Their experiment results show that energy-aware task consolidation technique can reduce power consumption in managing task consolidation for cloud systems. The simulation results show that ETC can significantly reduce power consumption in a cloud system, with 17% improvement over MaxUtil. However, by using 70% principle for CPU utilization, the virtual machine still consumes energy because it is not fully utilized. ETC increases number of virtual machines during the task consolidation as compared to MaxUtil technique [2, 8].

The idea behind the bi-objective model is to combine the two cost functions to only benefit from ECTC and MaxUtil advantages. The algorithm will then provide the more energy efficient resource based on both of the considered aspects. We must note that ECTC computes the energy consumption of a given task on a selected resource, while MaxUtil looks after the more energy-efficient resource in terms of resource utilization. The ECTC cost function is designed to encourage resource sharing; the energy consumption of two tasks running in parallel is slightly superior than the energy consumption of a task ran alone [6, 7].

Therefore, the approach uses the two cost functions of ECTC and MaxUtil. The respective results are combined to build a point in a two-dimensional search space where ECTC and MaxUtil gives the x and y coordinate, respectively. Originally, Equation returns a value greater than zero only when applied on a resource allowing task consolidation. Among the collected results, the highest value identify the most energy- efficient resource (if $\tau1 \neq \tau0$), as well as the null value identifies empty resources (if $\tau1 = \tau0$). The two cost functions (ECTC and MaxUtil) have to be slightly modified because we want to construct the point in the search space [2, 6]. Energy consumption ej of a task (tj) on a given resource (ri) is defined as:

$$e_j = (p\Delta \times u_j + p_{min}) \quad (7)$$

ECTC has a cost function called fi,j. For a task tj on a resource ri, we can calculate value fi,j as follows:

$$f_{i,j} = \begin{cases} (e_j \times \tau_0) & ; \text{if } \tau_1 = \tau_0 \\ ((e_j \times \tau_1) + (p\Delta \times u_j \times \tau_2)) & ; \text{otherwise} \end{cases} \quad (8)$$

MaxUtil cost function called **fi, j** is defined as:

$$f_{i,j} = \sum_{a_j}^{d_j} U_i \quad (9)$$

aj ,dj are the arrival time and the due date of the current task tj, respectively.

However, BTC algorithm decrease energy consumption rather than ECTC and MaxUtil. The result of their study should not only contribute on the reduction of electricity bills of cloud computing infrastructure providers, but also promote the combinations of existing techniques toward optimized models for energy efficient use, without performance degradation [7].
As noted, Bi-objective algorithm by combining ECTC and MaxUtil costs dramatically reduced energy consumption. As well as, ETC can significantly reduce power consumption in a cloud system, with 17% improvement over MaxUtil. Our proposed algorithm utilizes the advantages of two algorithms to improve energy efficiency and the completion of tasks. In the following, our proposed algorithm will implement with Cloudsim simulator and we will prove to achieve these objectives.

EPTC(ENERGY AND PARALLEL TASKS WITH TASK CONSOLIDATION)

In this section, we want to present proposed algorithm. This approach uses the two functions of described in equation 6, 8. The respective results are combined to build a point in a two-dimensional search space where ECTC gives the x coordinate and ETC the y coordinate. As well, we assume that there are three virtual clusters in the cloud systems. The strategy can be defined as follows:

1. We consider an optimum resource called r^* and set optimum=$\emptyset$.
2. The scheduler of VC_A dispatches task tj to a VM. If more than one VM is available, an appropriate VM is selected based on the best-fit strategy and calculate equation 6 and compare with optimum resource. If that is larger than optimum then it puts in value r^* as optimum resource and updates optimum value.
3. If there is no VM available, and Vi is below the specified CPU utilization threshold, VCA asks for resource support from other VCs, e.g., VCB or VCC.
4. If both VCB and VCC can provide VMs that run below the specified CPU utilization threshold, then Eq. (8) is used to select the VM from the VC that consumes the least amount of energy when transmitting and executing the task and the amount of calculation from equation 8 put in variable f_y. As well, equation 6 calculates and the amount of calculation from equation consider as variable f_x and these two values compare with optimum value. . If that is larger than optimum then it puts in value r^* as optimum resource and updates optimum value.
5. The scheduler of VC_B dispatches task tj to a VM. If more than one VM is available, an appropriate VM is selected based on the best-fit strategy and calculate equation 6 and compare with optimum resource. If that is larger than optimum then it puts in value r^* as optimum resource and updates optimum value.
6. The scheduler of VC_C dispatches task tj to a VM. If more than one VM is available, an appropriate VM is selected based on the best-fit strategy and calculate equation 6 and compare with optimum resource. If that is larger than optimum then it puts in value r^* as optimum resource and updates optimum value.
7. If none of the VCs can provide a VM below the specified CPU utilization threshold, then tj is assigned to the Vi that consumes the least amount of energy locally (i.e., VCA).

EPTC algorithm:

Set r^* and optimum=$\emptyset$
For all tasks in the job queue do
{
If $r_i \in$ cluster (A) and cpu Utilization Threshold
//Check the CPU Utilization Threshold for all tasks in the job queue of a virtual cluster VC_A
    If f_x>optimum then

{ $r^* = r_i$
optimum $= f_x$
}
If none of the resources in VC_A can execute task tj without surpassing the CPU Utilization Threshold
{
Confirm_CUT (VC_B, tj)
Confirm_CUT (VC_C, tj)
If more than one resource can execute task tj without surpassing the CPU Utilization Threshold
{
Calculate f_x And f_y
Put them in result
If result>optimum then
{ $r^* = r_i$
Optimum=result
}
}
If $r_i \in$ cluster (B) and cpu Utilization Threshold
//Check the CPU Utilization Threshold for all tasks in the job queue of a virtual cluster VC_B
If f_x>optimum then
{ $r^* = r_i$
optimum $= f_x$
}
If $r_i \in$ cluster (C) and cpu Utilization Threshold
//Check the CPU Utilization Threshold for all tasks in the job queue of a virtual cluster VC_C
If f_x>optimum then
{ $r^* = r_i$
optimum $= f_x$
}
Otherwise
Consolidate task tj to the VM in VC_A that consumes the least amount of energy.
}
Dispatch task tj to the r^*
}

This approach compared with previous algorithms dramatically reduces energy consumption and time to complete of tasks. In next section, we will show the evaluation of our approach in Homogeneous data center.

EVALUATION OF PERFORMANCE

Experiment configuration

We want to evaluate six scenarios. We assume that workload of virtual clusters of VCB and VCC are low, medium and high. We used pmax and pmin of 30 and 20, respectively. These values can be seen as rough estimates in actual resources and can be referenced as 300 watt and 200 watt, respectively. For simplicity, we should consider CPU utilization threshold 70% in our algorithm. As well as the simulation of the 53 tasks that task is between 10 and 100,000, have been used. We tried to indicate results of simulation in two datacenters. A small datacenter has 10 resources and a big datacenter has 20 resources. These results in small datacenters show as follows:

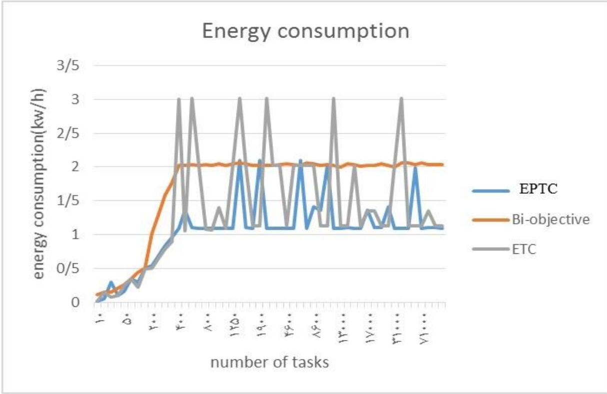

Figure 2: energy consumption with low workload in VC_B and VC_C

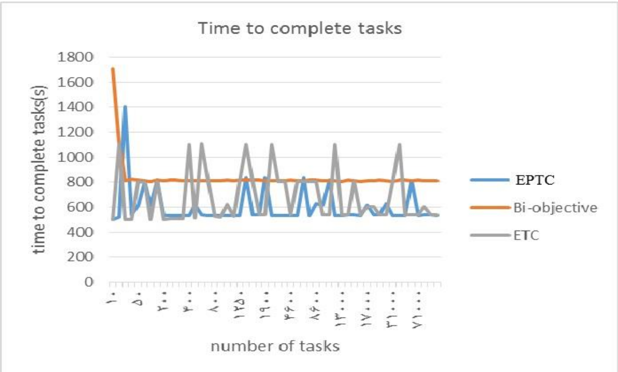

Figure 3: Time to complete with low workload in VC_B and VC_C

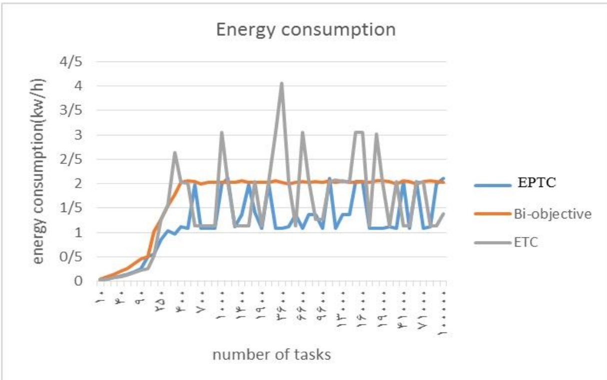

Figure 4: energy consumption with medium workload in VC_B and VC_C

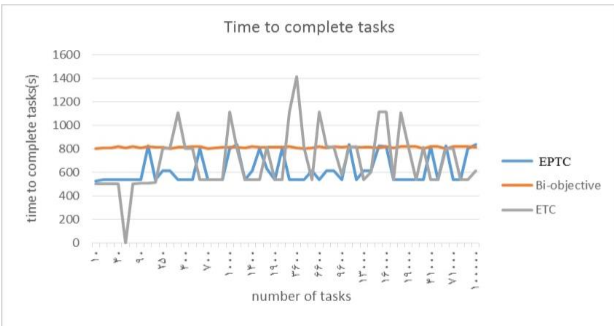

Figure 5: Time to complete tasks with medium workload in VC_B and VC_C

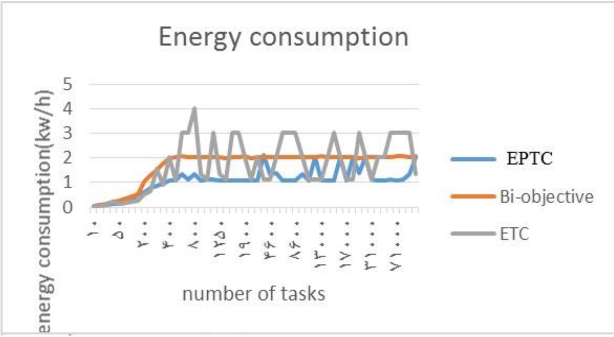

Figure 6: energy consumption with high workload in VC_B *and* VC_C

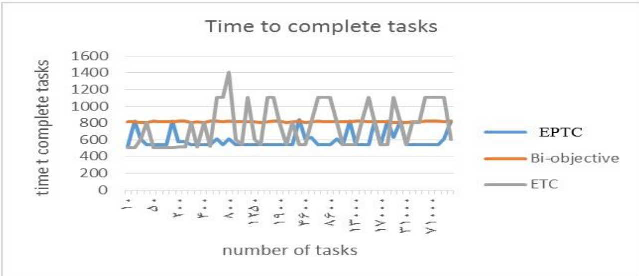

Figure 7: Time to complete tasks with high workload in VC_B *and* VC_C

The simulation in big datacenters show as follows:

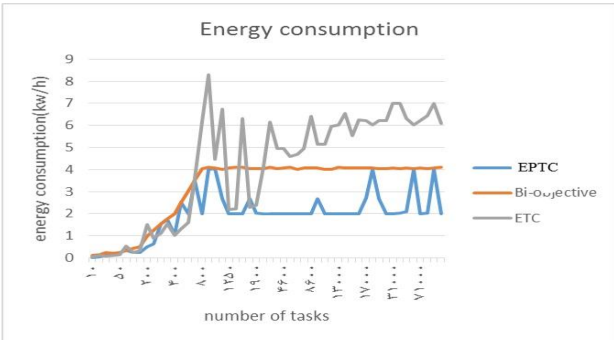

Figure 8: energy consumption with low workload in VC_B *and* VC_C

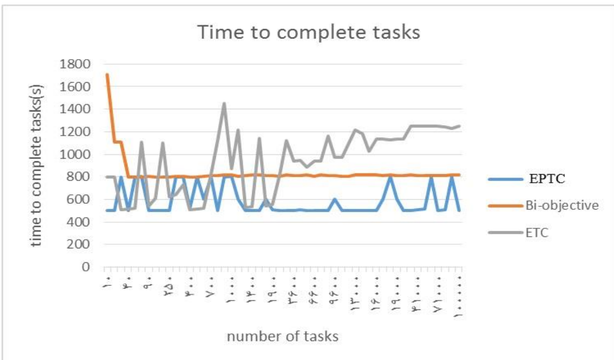

Figure 9: Time to complete with low workload in VC_B *and* VC_C

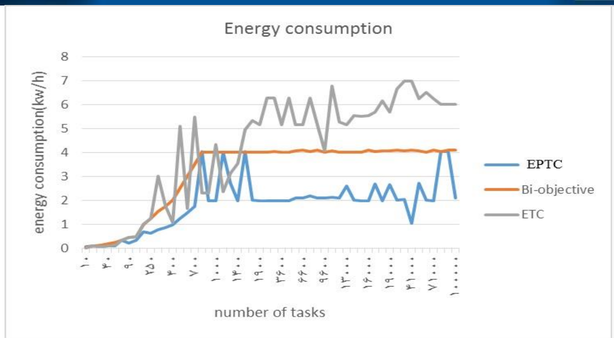

Figure 10: energy consumption with medium workload in VC_B and VC_C

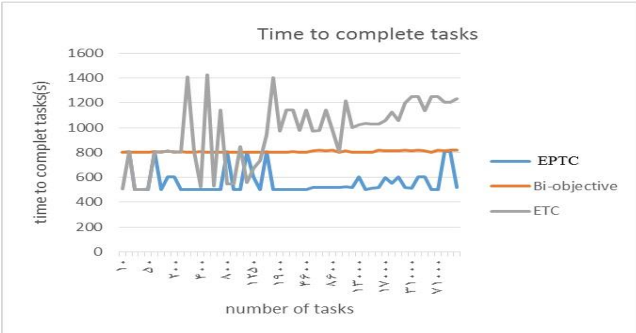

Figure 11: Time to complete with medium workload in VC_B and VC_C

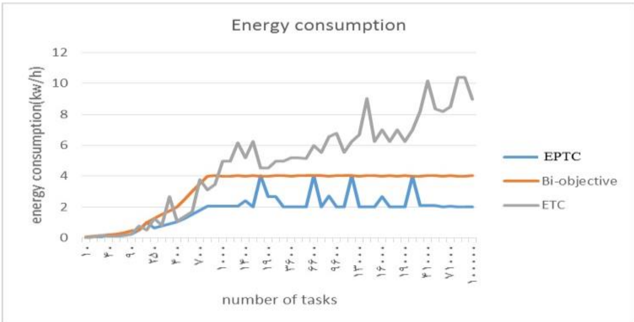

Figure 12: energy consumption with high workload in VC_B and VC_C

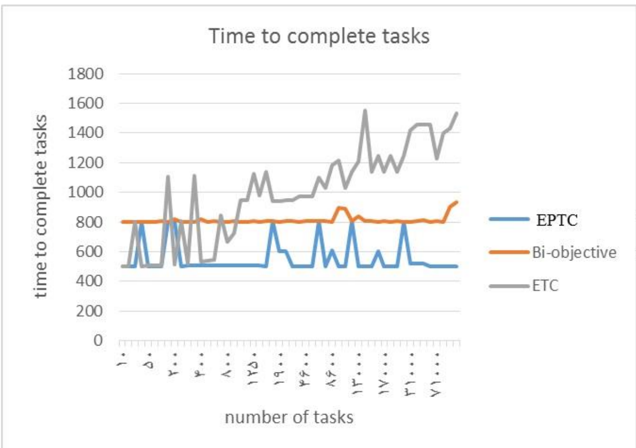

Figure 13: Time to complete with high workload in VC_B and VC_C

Experiment results

As you can see in the output, energy consumption algorithm rather than Bi-objective and ETC algorithm has been improved. As well, time to complete tasks rather than previous algorithms significantly has reduced (table I).

Table I: Percent improvement in energy consumed by the proposed method compared to other methods.

Percent improvement in energy consumption and the completion of tasks with the proposed method	Time to complete tasks	Energy consumption
Proposed rather than Bi-objective	28%	30%
Proposed rather than ETC	30%	49%

CONCLUSIONS AND FUTUR WORK

As cloud and green computing paradigms are closely related and they are gaining their momentum, the energy efficiency of clouds has become one of most crucial research issues. Task consolidation particularly in clouds has become an important approach to streamline resource usage and in turn improve energy efficiency [7]. Task consolidation is an effective technique greatly enabled by virtualization technologies, which facilitate the concurrent execution of several tasks in virtual machines and in turn reduce the energy consumption. In this paper we wanted to introduce proposed algorithm with task consolidation. This algorithm leads to reduce energy consumption in clouds. . Simulation results shows that energy consumption of the proposed algorithm rather than Bi-objective algorithm 30% and time to complete tasks 28% improved. Bi-objective algorithm has used ECTC and MaxUtil algorithms and these two algorithms are non-cluster. Therefore, the migration task from one cluster to another cluster are not considered. On the other hand, awareness of resource in each cluster doesn't exist because of the lack of clusters in these algorithms. But proposed algorithm can solve these problems.

A computational algorithm requires time calculations and implementation algorithm leads to consume energy and time. However, by using 70% principle for CPU utilization, the virtual machine still consumes energy because it is not fully utilized. ETC increases number of virtual machines during the task consolidation as compared to MaxUtil technique [3,4]. It is also one of the disadvantages of the proposed approach. Therefore, ETC increases number of virtual machines during the task consolidation as compared to MaxUtil technique.

In ETC algorithm, network bandwidth between the clusters is an important issue. Therefore, this issue should also be considered in the proposed method and it is considered in the future work.

References

[1] I. Ghani, N. Niknejad and S. Jeong. (2015). energy saving in green cloud computing data centers: a review. Journal of Theoretical and Applied Information Technology. Vol.74 No.1.

[2] C, Hsu. K, Slagter. S, Chen and Y, Chung. (2014). Optimizing energy consumption with task consolidation in clouds. Information Sciences. Vol 258. pp. 452–462.

[3] S, Sohrabi. I, Moser. (2015). A survey on Energy-Aware Cloud. European Journal of Advances in Engineering and Technology. vol .2. No. 2.

[4] A, Bharathi. R.S Mohana and A, Ushapriya. (2014). Reducing Energy Consumption and Increasing Profit with Task Consolidation in Clouds. International Journal of Engineering Science and Innovative Technology (IJESIT). Vol 3. Issue 1.

[5] Y. Choon Lee and A. Y. Zomaya. (2010). Energy efficient utilization of resources in cloud computing systems. Springer Science. DOI 10.1007/s11227-010-0421-3.

[6] A, Kaur. R, kaur and P, Jain. (2013). Algorithms for Task Consolidation Problem in a Cloud Computing Environment. International Journal of Computer Applications. vol .75. No. 4.

[7] G,Valentini. S, Khan and P, Bouvry.(2013). Large scale network-centric distributed system. Published in wiley Energy-efficient Resource Utilization in Cloud Computing.

[8] E, Sylvester Mkoba and M. A. Abdo Saif R. Nicole. (2014). A survey on energy efficient with task consolidation in the virtualized cloud computing enviroment. International Journal of Research in Engineering and Technology. Volume: 03 Issue: 03.

[9] L, Hao. G, Cui. M, Qu and W, Ke. (2014). Resource Scheduling Optimization Algorithm of Energy Consumption for Cloud Computing Based on Task Tolerance. JOURNAL OF SOFTWARE. Vol. 9. No. 4.

[10] H, Chen. X, Zhu. H, Guo. J, Zhu. X, Qin and J,Wu,.(2014). Towards Energy-Efficient Scheduling for Real-Time Tasks under Uncertain Cloud Computing Environment. to Journal of Systems and Software.

[11] F,Chen. J,Grundy. Y,Yang. J,Schneider and Q,He. (2013). Experimental Analysis of Task-based Energy consumption in cloud computing systems. ACM ICPE. pp.295-306.

[12] P.P Pandya and H.A, Bheda. (2014). Dynamic Resource Allocation Techniques in Cloud Computing. International Journal of Advance Research in Computer Science and Management Studies. Vol 2. Issue 1.

[13] A, Hameed. A, Khoshkbarforoushha. R, Ranjan. P, Prakash Jayaraman. J, Kolodziej. P, Balaji. S, Zeadally. Q, Marwan Malluhi. N, Tziritas. A, Vishnu. S. U, Khan and A, Zomaya. (2014). A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems. Springer. DOI 10.1007/s00607-014-0407-8.