International Journal of Scientific Research in Computer Science, Engineering and Information Technology
 | Volume 2 | Issue 2 | ISSN : 2456-3307

A Survey on Cloud Applications

J. Preethi[1], N. Aswathy[2]

[1]Assistant Professor, Department of CSE, Anna University Regional Campus, Coimbatore, Tamilnadu, India

[3]PG Student, Department of CSE, Anna University Regional Campus, Coimbatore, Tamilnadu, India

ABSTRACT

The numbers of users accessing the cloud datas are rising day by day. Clouds are based on data centers, which are powerful to handle large number of users, who can access anytime and anywhere. Data centers consumes huge amount of energy leads to increase the cost and carbon emission. Large number of data centers is easy to built, but not good for environment. The business community has begun to embrace cloud computing as a viable option to reduce the costs and to improve IT and business agility. Many techniques had been proposed in order to reduce the environmental impact of cloud application. In an existing system presented an approach to minimize the environmental impact of cloud-based application considering of its entire life cycle. An adaption mechanism derived by an adaption controller that reduces the CO_2 emission. Application Controller decides when to apply an adaptation strategy and decides the strategy most suitable for the given context is called as adaptation strategy selection which reduces the environmental impact and computation time and increases the performance in cloud applications. So we can prevent the air pollution by minimize the amount of CO_2 in air. In this paper we discussed various existing work related to reduce the CO_2 emission in cloud applications. By using high performance cloud environment co2 emission can be reduced and the performance also improved without an environmental impact.

Keywords : CO_2 Emission, Adaptation Strategy, Virtual Machine, Cloud Computing

I. INTRODUCTION

Commercial cloud Infrastructure-as-a-Service (IaaS) providers, such as Amazon EC2, offer several types of virtual machines (VMs) that differ in their amount of resources based on the pay-as-you-go model [1]. This allows cloud users to run their applications on the most appropriate virtual machine instances and pay for the actual resources that are used [2, 3]. However, the resources supplied by cloud providers can vary over time due to highly dynamic workloads that require resizing, creating and (or) terminating VMs. Furthermore, such resources consist of multiple types (or dimensions) including CPU, memory, disk and network bandwidth. As a consequence, if the owners of cloud data centers cannot effectively schedule and reallocate heterogeneous costs [4–6]. Conversely, increasing the workload of some VMs may cause the corresponding physical servers to be overloaded, possibly affecting the quality of service (QoS) experienced by the hosted applications. In fact, the QoS level offered to cloud users needs to fulfill the service level agreement (SLA) of the cloud provider [7, 8]. When a server is overloaded, it is challenging to determine which and how many VMs should be selected for migration to suitable hosts. As migration is expensive, VM selection plays an important role to limit the number of VMs migrations. Additionally, the target physical server also should be correctly selected for placing a VM under migration. For instance, the target host should not be overloaded in both the current and the future period of time after allocating the migrated VM. During the migration process, if there is no active physical server with sufficient resources available, an inactive server is started and the selected VMs are allocated on such a machine. On the other hand, when a host is underutilized, all VMs from such a host are selected for migration if they can be consolidated into other hosts without causing overutilization. Idle servers are then switched to a low-power state to save energy [9, 10].In this paper a study is made on the existing approaches and their limitation and the proposed techniques to solve above mention issues.

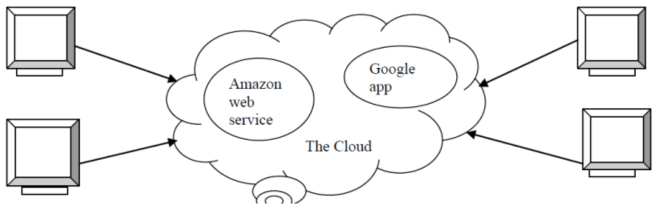

Figure 1. Cloud computing model

II. METHODS AND MATERIAL

Related Work

C. Peoples (2011) introduced an algorithmic mechanism designed in order to develop an automate selection of a data centre in response to application requests, the Data Centre (DC) Energy-Efficient Context-Aware Broker (e-CAB). The author achieves the reduction of carbon emission and balancing of other performance-related attributes including delay and the financial cost.

Beloglazov et al (2012) considered the resource allocation algorithm where resources are allocated taking into account total energy consumption, the number of violations of the Service Level Agreement (SLA), and the number of migrations. Resources allocation is related to both new and old VMs, which can be moved to improve system state. The approach uses bin packing for placing new Virtual machines and an algorithm lead by CPU usage for deciding when to migrate a VM.

DragosDiaconescuet al (2013) proposes a framework that aims to reduce the energy consumption of a data center. In many case resources are not used at their maximum capacity or even worst, even in some cases there are many resources that are not used at all. This research objective is to migrate the some resources between different data centers in order to turn off the some hosts that are not used.

Review of Energy Efficient and CO_2 Aware Cloud Computing

Cloud radio access network (C-RAN) has emerged as a promising solution to support exponentially increasing demand in the data rate. The attractive capacity enhancement mainly comes from the coordinated processing and , which poses great challenges on computing capability in the baseband unit pool. This section presents the various technologies to provide energy efficient and Co2 aware Cloud environment.

Virtual machine consolidation for energy efficient Cloud

Nguyen et al (2016) proposed a Virtual machine consolidation aims at reducing the number of active physical servers in the data center so as to decrease the total power consumption. In this context, most of existing solutions are rely on aggressive virtual machine migration, thus resulting in unnecessary energy wastage and overhead. Besides, virtual machine consolidation should take into account multiple resource types at the same time, since CPU is not the only critical resource in the cloud data centers. In fact, also network bandwidth and energy can become a bottleneck, possibly causing violations in the service level agreement. This article presents a virtual machine consolidation algorithm with the multiple usage prediction (VMCUP-M) to improve the energy efficiency of cloud data centers.

Usage prediction scheme is given in Algorithm 1.Such a method is especially attractive for consolidating VMs in cloud data centers with millions of heterogeneous machines and resource types due to its time complexity.

On Energy-Efficient Offloading in Mobile Cloud

Guoet al (2016) introduced a distributed eDors algorithm computation offloading selection which is computed by workload of a task. “Green Data Centers” refers to, energy aware, energy efficient and CO2 emission minimizing the designs, protocols, devices, infrastructures and algorithms for data centers. Reduce the EEC by optimally adjusting CPU clock frequency of SMDs based on the dynamic voltage and frequency scaling (DVFS) technique in local computing. This approach minimizes the energy consumption with delay constraints. The following algorithm is proposed by the author for computation offloading selection Cinzia Cappiello et al (2016) proposed a novel approach to reduce the environmental impact of CO2 emissions on cloud based application. The author aim to exploit adaptivity with the help of Application Controller which, enacting the right adaptation strategy for a given context. This approach provides the improvement of the trade off between QoS and CO2 emission reduction.

Diarmuid Grimes et al (2016) examine the impact of using predicted resource usage for optimal server consolidation. We investigate the occurrences of over-utilized resources on servers due to under-predicted resource usage. We propose methods to reduce the likelihood of such occurrences, both through the enforcement of safety capacities on the server side, and through biasing towards over-prediction on the VM side. The results indicate that an appropriate balance can be found between energy savings and non-violation of SLAs.

S.no	Author	Title	Methodology	Benefits	Limitation	Ref.no
1	Goyal, Sudhir, SeemaBawa, and Bhupinder Singh	Green Service Level Agreement (GSLA) framework for cloud computing	Green SLA aware cloud resource reservation (GSLACRR) algorithm.	offers cloud resource services in an energy efficient manner	High operational costs.	[18]
2	Karpowicz, Michał, EwaNiewiadomska-Szynkiewicz, PiotrArabas, and AndrzejSikora	Energy and Power Efficiency in Cloud	Power consumption models and energy-aware task scheduling. Resource allocation algorithms	ACPI-compliant low power idle	Performance of energy-aware server-level and network-level control mechanisms is poor	[19]
3	Wajid, Usman, Barbara Pernici, and Gareth Francis	Energy Efficient and CO2 Aware Cloud Computing: Requirements and Case Study	energy efficient and CO_2 aware cloud computing	energy efficient cloud sourcing	CO2 emission minimizing designs	[20]
4	Arroba, Patricia, and RajkumarBuyya	DVFS-Aware Consolidation for Energy-Efficient Clouds	consolidation algorithm, DVFS policy	provides substantial energy savings of up to 39.14% for scenarios under dynamic workload conditions	Imitated Consolidation performance	[21]
5	Yamagiwa, Motoi, and Minoru Uehara	A Proposal for Development of Cloud Platform Using Solar Power Generation	solar power and low consumption electricity PC	Provides the energy efficient cloud through solar power	Required more amount of sealed lead acid battery	[22]
6	Yamagiwa, Motoi, and Minoru Uehara	A Study on Constructing an Energy Saving Cloud System Powered by Photovoltaic Generation	Solar power	Provides cloud resource services in an energy efficient manner	small core server using the ARM processor used which result in low performance	[23]

S.no	Author	Title	Methodology	Benefits	Limitation	Ref.no
7	Cappiello, Cinzia, PierluigiPlebani, and Monica Vitali	Energy-Aware Process Design Optimization	proposes a method to support the process design by optimizing the configuration	Continuously guarantee good performance and energy efficiency.	there is still room for improvements	[24]

3. Work flow

1) Workload rearrangement
2) Time shifting
3) Configuration analysis
4) Adaptive strategy selection

3.1) Workload rearrangement

Flow rearrangement affects the structure of the application flow that can be modified by the Application Controller in different ways.

Rearranging the workload assigned to the tasks composing the application and switching off a VM if no longer needed.

Skipping tasks if they are defined as 'optional' in the Application Profile under critical timing conditions. In Workload rearrangement, the vm parameters (estimate power (p), estimate response time (rt), estimate energy mix (em) were calculated. The total CO_2 emission of a particular vm is calculated by the following formula.

$$VM\ [CO_2] = p*rt*em \quad (1)$$

After that, the tasks are rearranged iteratively, and the best of the vm in the vm list is selected. The tasks are allocated to selected virtual machines.

Work rearrangement algorithm

```
INPUT: tasks [no task]: tasks to be executed
INPUT: vmset [no_vm]: list of vm nodes
OUTPUT: worklist [no_vm]: task assigned to nodes
        If no_vm==1 then
           add All(worklist[0],tasks)
           exit
        end if
        for vm: vmset do
             p ← ESTIMATE_POWER
             rt ← ESTIMATE_RESPONSE
TIME(vm)
             em ← ESTIMATE_ENERGYMIX(vm)
             V M CO2[vm] ← p* rt* em
        end for
        tot_CO2 ← SUM(V MCO2)
        j ← no_tasks
          for vm: vmset do
             no_assigned_tasks ← no tasks*
```

$$*round(\frac{1}{V\ M\ CO2[vm]} * \frac{1}{\sum 1/tot_{CO2}})$$

```
If no_assigned_tasks>0 then

For  i: no_assigned_tasks  do
          j - -
          if j >=0 then
add(worklist[vm],task[no_tasks-j])
          end if
               end for
        else
SWITCHOFF(vm)
               no_vm - -
        end if
       end for
       bestvm= indexO f(min(VM CO2))
       if j > 0 then
           add(worklist[bestvm],tasks[no_tasks])
       end if
```

3.2) Time shifting

The time shifting work is exploits regular variations of emission factors to delay execution of the application and reschedule it in time intervals in which emission factors are the expected to decrease and CO_2 emissions are lower. For example, if users submit their request during the afternoon, the Application Controller, considering the current site response time, variations defined in the pattern, emission factors and the estimated energy consumption of the application, can calculate the quantity of CO_2 emissions at different times and decide to delay the execution to most suitable time (e.g., in the night) for minimizing CO_2 emissions.

Time Shifting algorithm

INPUT: site [no_{sites}]: emission factor profiles
INPUT: e f pattern [time, variations]: emission factor pattern
INPUT: app: application to be executed
INPUT: site: site in which application has to be executed
OUTPUT: starttime: proposed start time

```
rt ← ESTIMATE_RESPONSETIME(app)
en ← ESTIMATE_ENERGY(app)
starttime ← CURRENTTIME
e f_current ← ESTIMATE_EFAVG(site,time,rt)
emissions ← en* e f
for time:efpattern do
    e f_est ← ESTIMATE_EFAVG (site,time,rt)
    emissions_est = en* e f_est
    if emissions_est < emissions_est then
        emissions ← emissions_est
        start time ← time
    end if
end for
```

3.3) Configuration analysis

In this section, we shift our focus to the achieving energy efficiency by modifying the application profile, in particular the resource metadata. We propose an analytical approach to compare several configurations that differ in how the resources are using and how the tasks are executing, particularly related to:

The number of used resources: it refers to the number of used VMs and their size to execute the tasks

- ✓ The execution policy: it refers to the way the tasks are performed
- ✓ The storage access policy: synchronous vs. asynchronous access.

We use queuing theory as a mean to model different configurations to execute a given set of tasks. In the queuing model, computing resources (e.g., the VMs) are represented as a network of stations and the number of executed tasks is presented as the customers. The basic station can be either queue station or delay station (i.e., station has no queue) and it is characterized by service demand (i.e., the time required to serve one job at the station). Other stations are performing the advanced operations such as Fork and Join to simulate synchronous/asynchronous access. Each configuration has a queuing model whose inputs include:

- ✓ The number of tasks to be executed, no tasks.
- ✓ The number of stations and their type either queue or delay station
- ✓ The queue station is used in case of having shared elements
- ✓ The set of service demand D of the stations
- ✓ Some of these inputs can be extracted from the Application Profile.

Configuration analysis algorithm

INPUT: no_tasks: number of tasks to be executed
INPUT: C < m, [stations,station_types, D] >: the set of configurations, each is characterized by its corresponding queueing model m, number of stations, types of stations station_types, the set of required service demand at each station D

OUTPUT: <T, E>: the set of estimated energy consumption and execution time of the configurations

```
for c: configurations do
    compute execution time T_c from m
    compute consumed power P_vm (see Eq.2)
    if T_c and E_c satisfy constraints then
        insert < T_c , E_c > to < T,E >
    end if
end for
return <T,E>
```

3.4) Adaptive strategy selection

In this section a set of techniques that are employed in the strategy selector module of the Application Controller to decide when to apply an adaptation strategy and which strategy is most suitable for given a context. Similarly, an inefficient use of the resources allocated to the VMs can activate the Application Profile refinement strategy to find a better configuration for a future execution of the application, while the flow rearrangement strategy can also be activated by the detection of a modification of the behavior of a VM in terms of response time or power consumption, which can be an input for rearranging the workload between the different active VMs. Finally, a high value for CO2 emissions in a site where the application or one of its tasks is running can enact the time shifting strategy to detect the better starting time. Given this knowledge, the awareness about the relations between variables allows the selector to reason about indirect improvements, knowing how the metrics interfere with each other enables to enact strategies which can directly improve the violated metrics, or indirectly improve them.

Adaptive strategy selection algorithm

INPUT: BN: the BN of relations among indicators
INPUT: A_set: the set of available adaption strategies
INPUT: Q[no_strategies, no_indicators]: the quality matrix
INPUT: C[no_indicators]: the current state (context) of the indicators
OUTPUT: a*: the strategies to be enacted

if I_n is violated then
  $w_n = 1$
else if I_n is near to be violated then
  $w_n = 0.5$
else
  $w_n = 0$
end if
for a: A_set do
  p(a|C) ⟵ $\sum f(Q[a, In] * wn)$ the impact of strategy a over indicator I_n
end for
for p: P do select the parent set P of the violated indicators from B N
  $C' = C \cup$ the state for p maximizing the probability of improving violated indicators
End for
Compute values for w_n according to C'
For a: A_set do
  p(a|C') ⟵ $\sum f(Q[a, In] * wn)$
end for
select a* with the highest livelihood of success given p(a|C) and p(a|C')

III. RESULTS AND DISCUSSION

Table 2. Energy Consumption (KWh) HPC

TASK/method	CO2-aware Adaptation Strategies for Cloud Applications
5	19
10	29
15	48
20	60
25	69

If the task method given in the table 2 as 15, then the virtual machine are assigned automatically and executed. The co2 aware adaptation strategy for cloud application results in 48 KWh.

Table 3.Execution time (Sec) HPC

TASK/method	CO2-aware Adaptation Strategies for Cloud Applications
5	21
10	27
15	33
20	42
25	45

If the task method given in the table 3 as 10, then the virtual machine are assigned automatically and executed. The co_2 aware adaptation strategy for cloud application results in 27 Sec.

Table 4. Emissions HPC

TASK/method	CO2-aware Adaptation Strategies for Cloud Applications
5	235
10	251
15	261
20	277
25	290

If the task method given in the table 4 as 20, then the virtual machine are assigned automatically and executed. The co_2 aware adaptation strategy for cloud application results in 277.

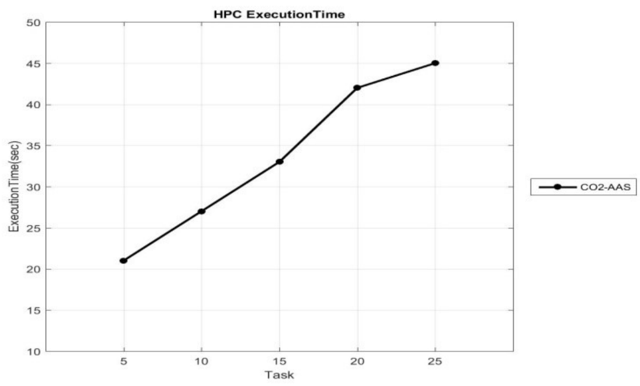

Figure 3. Execution time high performance cloud

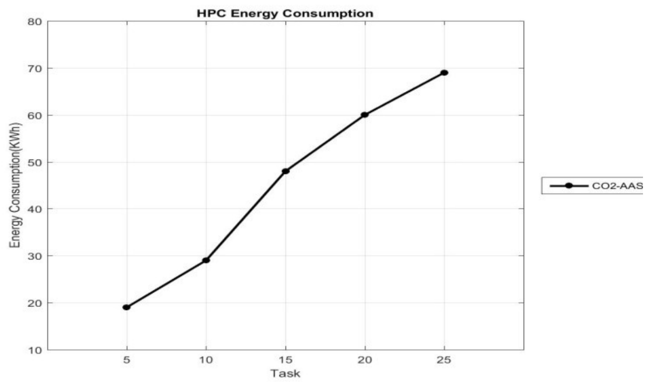

Figure 4. Energy consumption high performance cloud

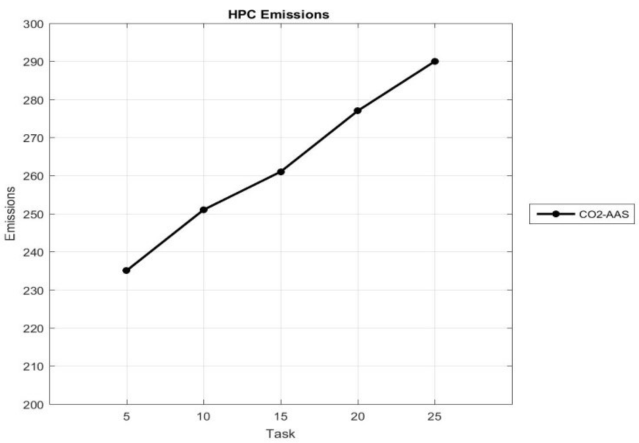

Figure 5. Emission high performance cloud

IV. CONCLUSION AND FUTURE WORK

The utilization of cloud resources increases several issues related to their environmental impact. Recently, most of the contributions have focused on energy efficiency achieved through a better physical and virtual resource management. In an existing system presented an approach to minimize the environmental impact of cloud based application considering of its entire life cycle. An adaption mechanism derived by an application controller that reduces the co2 emission. In this paper we have examined the various existing work Co2 emission and reduction. Above Result is obtained by using high performance cloud environment adaptation strategy is selected with the workflow workload rearrangement, time shifting, configuration analysis, adaptation strategy is used to reduce the co2 emission .In future work, multiple adaption strategy is introduced by using an efficient K2-Simulated Annealing algorithm which controls solving process to the optimization direction of minimum, and can escape from local extreme points with accepting inferior solutions at a certain probability. The multiple adaption strategy allows the user to more than one adaptation action at the same time.

V. REFERENCES

[1]. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic,"Cloud computing and emerging it platforms: Vision, hype, andreality for delivering computing as the 5th utility," Journal of Future Generation Computer Systems, vol. 25, pp. 599–616, 2009.

[2]. S. Bhardwaj, L. Jain, and S. Jain, "Cloud computing: A study of infrastructure as a service (iaas)," International Journal of Engineering and Information Technology, vol. 2, pp. 60–63, 2010.

[3]. J. Zhu, Cloud Computing Technologies and Application. Handbook of Cloud Computing, 2010, pp. 21–45

[4]. C.-H. Hsu, K. D. Slagter, S.-C. Chen, and Y.-C. Chung, "Optimizingenergy consumption with task consolidation in clouds," Information Sciences, vol. 258, pp. 452–462, 2014.

[5]. S. Chaisiri, B.-S. Lee, and D. Niyato, "Optimization of resource provisioning cost in cloud computing," IEEE Transactions on Services Computing, vol. 5, pp. 164–177, 2012.

[6]. R. W. Ahmad, A. Gani, S. H. A. Hamid, M. Shiraz, A. Yousafzai,and F. Xia, "A survey on virtual machine migration and serverconsolidation frameworks for cloud data centers," Journal of Networkand Computer Applications, vol. 52, pp. 11–25, 2015.

[7]. A. Beloglazov and R. Buyya, "Managing overload hosts for dynamicconsolidation of virtual machines in cloud data centersunder quality of service constraints," IEEE Transactions of Paralleland Distributed Systems, vol. 24, pp. 1366–1379, 2012.

[8]. S. K. Garg, A. N. Toosi, S. K. Gopalaiyengar, and R. Buyya, "Slabasedvirtual machine

management for heterogeneous workloadsin a cloud datacenter," Journal of Network and Computer Applications,vol. 45, pp. 108–120, 2014.
[9]. M. Blackburn, "Five ways to reduce data center server powerconsumption," The Green Grid, Tech. Rep., 2008. Online Available: http://www.thegreengrid.org//media/WhitePapers/White Paper 7 - Five Ways to Save Power.pdf?lang=en
[10]. V. Mathew, R. K. Sitar man, and P. Shenoy, "Energy-aware load balancing in content delivery networks," in 2012 IEEE INFOCOM,2012, pp. 954–962.
[11]. Nguyen, TrungHieu, Mario Di Francesco, and AnttiYla-Jaaski. "Virtual Machine Consolidation with Multiple Usage Prediction for Energy-Efficient Cloud Data Centers." IEEE Transactions on Services Computing (2017).
[12]. Guo, Songtao, Bin Xiao, Yuanyuan Yang, and Yang Yang. "Energy-efficient dynamic offloading and resource scheduling in mobile cloud computing." In Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on, pp. 1-9. IEEE, 2016.
[13]. Cappiello, Cinzia, Nguyen Ho, Barbara Pernici, PierluigiPlebani, and Monica Vitali. "Co2-aware adaptation strategies for cloud applications." (2015).
[14]. Peoples, Cathryn, Gerard Parr, and Sally McClean. "Energy-aware data centre management." In Communications (NCC), 2011 National Conference on, pp. 1-5. IEEE, 2011.
[15]. A. Beloglazov, R. Buyya, Y. C. Lee, and A. Zomaya, "A Taxonomyand Survey of Energy-Efficient Data Centers and Cloud ComputingSystems," Advances in Computers, vol. 82(2), pp. 47–111, 2011.
[16]. Diaconescu, Dragos, Florin Pop, and ValentinCristea. "Energy-aware Placement of VMs in a Datacenter." In Intelligent Computer Communication and Processing (ICCP), 2013 IEEE International Conference on, pp. 313-318. IEEE, 2013.
[17]. Grimes, Diarmuid, Deepak Mehta, Barry O'Sullivan, Robert Birke, Lydia Chen, Thomas Scherer, and Ignacio Castineiras. "Robust Server Consolidation: Coping with Peak Demand Underestimation." In Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), 2016 IEEE 24th International Symposium on, pp. 271-276. IEEE, 2016.
[18]. Goyal, Sudhir, SeemaBawa, and Bhupinder Singh. "Green Service Level Agreement (GSLA) framework for cloud computing." Computing 98, no. 9 (2016): 949-963.
[19]. Karpowicz, Michał, EwaNiewiadomska-Szynkiewicz, PiotrArabas, and AndrzejSikora. "Energy and Power Efficiency in Cloud." In Resource Management for Big Data Platforms, pp. 97-127. Springer International Publishing, 2016.
[20]. Wajid, Usman, Barbara Pernici, and Gareth Francis. "Energy efficient and CO2 aware cloud computing: Requirements and case study." In 2013 IEEE International Conference on Systems, Man, and Cybernetics, pp. 121-126. IEEE, 2013.
[21]. Arroba, Patricia, and RajkumarBuyya. "DVFS-Aware Consolidation for Energy-Efficient Clouds." In 2015 International Conference on Parallel Architecture and Compilation (PACT), pp. 494-495. IEEE, 2015.
[22]. Yamagiwa, Motoi, and Minoru Uehara. "A proposal for development of cloud platform using solar power generation." In Complex, Intelligent and Software Intensive Systems (CISIS), 2012 Sixth International Conference on, pp. 263-268. IEEE, 2012.
[23]. Yamagiwa, Motoi, and Minoru Uehara. "A study on constructing an energy saving cloud system powered by photovoltaic generation." In 2012 15th International Conference on Network-Based Information Systems, pp. 844-848. IEEE, 2012.
[24]. Cappiello, Cinzia, PierluigiPlebani, and Monica Vitali. "Energy-aware process design optimization." In Cloud and Green Computing (CGC), 2013 Third International Conference on, pp. 451-458. IEEE, 2013.