International Journal of Global Engineering (IJGE) E- ISSN: 2456-3099

VOL 2 ISSUE 3 (2018) PAGES 100-108

Received: 4/02/2018. Published: 26/02/2018

Attaining Energy and Storage Adaptability via Energy efficient Computational offloading with CO2 reduction

S.Sashikumar, G.VijayBabu, T.Johnpeter

Department of Computer Science and Engineering

Dhanalakshmi Srinivasan Institute of Technology, Samayapuram, Tamil Nadu, India

Abstract

Achieving strength effectivity throughout computational offloading is the indispensable and is the high goal of this work. Energy efficient Computational offloading with CO2 discount (EECOC) is proposed for accomplishing strength efficiency. Load balancing and CO2 minimization are the manipulate parameters considered all through the work. Green computing is accomplished when CO2 is minimized. Hence load balancing and computational offloading leads to the favored objective. Run time migration is considered to decrease the overhead in terms of idle machines. CO2 is emitted by using VMs in case they are over burdened. Computation offloading is performed to reduce load on VMs for this reason decreasing ranges of CO2. Load distribution considers the configuration of VMs. Machines with higher configuration is chosen at first vicinity for allocation. Load is similarly decreased through doing away with redundancy in given job. Results point out that Energy efficient Computational offloading with CO2 discount reduces electricity consumption via 23%.

Keywords — Load Balancing, CO2, Green Computing, Redundancy. EECOC, Energy Efficiency.

Introduction

Cloud computing is used to supply consumer with the resources extra than functionality of their bodily machines [1]. Cost is encountered on the foundation of pay per use. [2], [3] as load on VMs in cloud increases, performance degrades. This degradation in performance also motive VMs to emit CO2 . Load on VMs required to be balanced to minimize CO2. Computational offloading is used to limit load and CO2 levels. Energy Efficiency is achieved through thinking about [4]live VM migration in case of deterioration occurs in VMs. [5], [6]Energy environment friendly scheduling is additionally used for reaching optimization during migration. load can be in addition decreased by using thinking about redundancy factor. The jobs containing comparable work can be eliminated from the job to decrease load on the VM. This reduces the load and as a result emission of CO2 levels. [18]Proposes strength efficient method for lowering CO2 out of VMs. The method makes use of temperature as a base mechanism for decreasing CO2 and bettering electricity efficiency. Temperature tiers are viewed as threshold with which temperature of VMs are compared. In case temperature of VMs enhances past threshold values than workload from present day VM is offloaded to next VM in sequence. Proposed methodology is primarily based on improving performance and reducing load, as a result decreasing emission of CO2 levels. Rest of the paper is geared up as under. Section 2 describes scheduling and its targets in superior computing. Section 3 describes the current work which is achieved in order to achieve energy efficiency. Section four describes the proposed work. Section 5 offers the result and performance analysis and subsequent area offers conclusion and future work.

Scheduling and Its Objectives

Scheduling in multi cluster environment is necessary area of research. Legion of scheduling algorithms are enjoying a phase to agenda sources in multi cluster environment. Task can arrive from geographical giant area and this existing a task that which task have to be given a useful resource at first place. In multi cluster environment Computer, data, and other resources are shared[7]. The service company and customers must agree upon what to be shared within given environment. The international nature of useful resource sharing is unique facility supplied inside multi cluster environment. The useful resource may also be existing within one of a kind administrative domain and demanded through node belonging to some other domain [8]. Scheduling henceforth turns into essential for choicest and deadlock free system. Scheduling is the system of assigning assets to jobs primarily based on goal features defined. Type of scheduling relies upon upon the objective characteristic associated with the resource. Scheduling assets has following phases related with it.

- Resource Discovery
- Resource Filtering
- Resource Selection
- Resource Scheduling Policy

Before allocation inside multi cluster environment, resources must be discovered. Resource may or may additionally no longer be available[7]. Hence this segment turns into critical for monitoring of useful resource inside the system[9]. The reachable resources need to be checked to decide whether they satisfy the necessities or not. Hence filtering is compulsory. Resource choice out of handy resources is next phase. This segment is integral seeing that out of reachable resources of identical kind aid with highest quality circumstance is selected for allocation. This is required so that job can be carried out nicely within time. Healthier useful resource choice is the goal of this phase. Scheduling policy decides the useful resource allocation is primitive or not. Resource allocation is said to be primitive if once allocation aid can be triggered from assignment even if it is not but fully completed.

Study of Literature

Energy efficiency is fundamental while assets are allotted to VMs. Work has been executed in the direction of this aspect. This part describes the techniques used to achieve strength effectivity with the aid of lowering load using computational offloading mechanism. [10] Propose strength efficient cellular cloud computing using wireless electricity transfer. The technique combines cellular cloud computing and microwave strength transfer technique. Using this technique it is possible to function computation in wearable devices. Set of insurance policies are formulated for controlling CPU cycles in case of local computing and offloading for different mode of computing. [11] Suggests power constraint mechanism to make certain job execution efficiently. Code migration is cautioned to optimize energy efficiency. Pre-copy with remote execution takes place. With faraway execution, job executes from the faraway server. In case of deterioration, job is migrated through code and consequently progress of job is saved and it is done once more from the location it is stopped on preceding machine. Results show full-size enchancment in phrases of downtime and migration time. [12] Researched a challenge computing and fee of file offloading to reduce power consumption. Radio aid allocation is specially viewed in this literature. Energy environment friendly computational offloading (EECO) on 5G network is proposed in this paper. Uplink and Downlink transmission fee is viewed via the following equations.

Uplink Transmission Rate 2, = 2(1 + + 2)

Equation 1: Uplink Transmission Rate

Where _P‘ is the energy of mobile device, _I‘ denotes the interference, _g‘ indicate the channel gain, _σ‘ is the noise.
Downlink Transmission Rate = 2(1 + + 2)

Equation 2: Downlink Transmission Rate

Channel for having access to used is M. Cost beneath the extend constraint is reduced considerably.

[13] Proposes a decentralized method for cell computational offloading. Decentralized approach follows more than one digital machines on which load is distributed. The computation is significantly reduced on person machine. The power efficiency is completed considering the fact that precedence whilst allocation is considered. Results point out expanded performance.

[14] Proposes obligation biking mechanism to attain strength effectivity in scheduling of resources in wi-fi sensor network. Duty biking is divided into electricity management and topology manipulate mechanisms. Node redundancy is regarded in topology manipulate and strength management is considered in case of sensor allocation. Sensors have limited strength and electricity related with them. This work correctly manages each strength and power and for this reason a result bought is higher in terms of energy efficiency. Minimum load a node can tackle is given via the following equation

(( )) ≤ ( ) ≤ ( ) ≤ minf()(degf()( )

Equation 3: Load equation for nodes

_G‘ suggests the design of the structure G={V,E}, _V‘ is the set of vertex and _E‘ is the set of edges. _n‘ suggests total range of nodes.

[15]consider each dynamic energy as properly as leakage power for strength effectivity throughout scheduling. Precedence constraint is employed in this case. Jobs consequently are performed in phrases of priority instead than sequential. The execution time is calculated in phrases of following equations.
= (1 −) ∗ + ∗

Equation 4: Execution time calculation.

Jobs are achieved on 1, 2 and four cores for checking the strength consumption. Results exhibit higher scheduling as in contrast to other scheduling mechanisms.

Next section describes the element proposed methodology which decorate Green computing by means of reducing the CO2 stage and reduces strength consumption also.

Proposed System

The proposed device reflect onconsideration on parameters Load, Energy and scheduling on man or woman machines inside cloud environment. The normal company of proposed methodology is listed as follows

Algorithm EECOC

Initialization area
Set i=0,
Max=Maximum_VM,
Threshold=Capacity_of_vm,
MaxJ=maximum_no_jobs_in_queue
1. Input Jobs indicated via Cloudlet(Jobsi).

2. Assign Priority with every Job(Cloudlet). Maintain Jobs according to precedence in P_Queue.

3. Specify Threshold cost associated with every VMi. Threshold is the most limit of load associated with each VMi

4. Repeat while i<maxj <br=""></maxj>
    4.a Assign job to VMi
    4.b if Loadi>Threshold then
    4.c I=i+1(Migrate the Load to other VM in sequence)
    End of if
    4.d Load_on_vmi=Loadi End of Loop

5. Load on Individual Machine=Load_on_vmi

6. CO2_level=Consumed_Loadi

7. Storage_Utilizationi=S_used/S_Total

The proposed methodology is designed to reduce energy consumption and in consequence minimize CO2 levels. The proposed machine is described in element as under.

A. Priority Job Queue

[16], [17] As jobs arrive inside the system, priority is assigned to them. This precedence in proposed machine is allocated with the aid of identifying sorts of jobs. The jobs arriving inside the system are divided into categories as

System Processes3

- User Processes
- System tactics are usually given absolute best preferences and ser tactics are give n least preferences. The precedence is assigned on the basis of following elements
- Type of operation is identified and priority is assigned as operation number.
- Client laptop performing the operation.
- The type of agent through which job is originated.
- Priority of customer pc in references to job order.

After precedence is assigned, jobs are grouped collectively within the queue.
[,] = { 1, 2, – – – – }
Equation 5: x signify jobs organized in the structure of a queue.

B. Assigning Threshold Load To Virtual Machines
Threshold load is assigned to VMs on the groundwork of abilities of Host machines. Host machines if has configuration of 10GB RAM, 2000GHz processor and generated VMs are 10 then configuration of each VM is listed as follows.

VM	RAM	Processor
1	1 GB	200GHz
2	1GB	200GHz
------	------	-----
------	------	-----
-----	------	-----
-----	-----	-----
-----	-----	------
10	1GB	200 GHz

Table 1: Configuration of VMs

As the capabilities of VMs are equal consequently identical threshold value can be allotted to every VM. Threshold fee much less than 1 GB(in phrases of memory) and less than 200GHz(in phrases of processing requirement) are assigned to every virtual machine.

C. Job Migration for Energy

Efficiency
The abilities of VMs are checked earlier than allocation of job. In case, job necessities are extra than threshold value, subsequent VM in sequence is checked. This technique continues till all the jobs are allotted and executed. The distinctive variable ‗expire' is introduced. This variable suggests VM capability expires or not. If, VM capability is now not two expired then job can proceed on contemporary machine. Since laptop capability is now not used beyond its capability as a result electricity consumption is reduced considerably. After executing the jobs parameter calculation procedure begins. This is described as under

D.Parameter Evaluation

The job execution is accompanied by parameter evaluation. The parameter which how well worth of find out about are

- Load
- Energy
- CPU utilization
- Storage Utilization

The CO2 emission radically depends upon load on modern-day machine. In order to calculate CO2 emission through person machine, default fee of CO2 emission per unit load is assumed for every machine. As the load on VM increases, CO2 ranges increases. Hence CO2 greatly relies upon upon the load on modern-day machine.

Load is evaluated through the amount of time job is carried out on present day machine. Let = { 1, 2 − − −
−, } suggests the jobs and V={ 1, 2, − − −, } are the set of VM. Load on V1 is according to burst time of Jobs. J1 if has burst time of 50 and is executed on V1 than load on V1 is 50.

Consumed power is calculated by way of the use of following method
= * one hundred
Equation 6: Energy estimation mechanism
_Load‘ shows the amount of load on modern-day computer per unit time. It is given thru following equation.
= ( ) * ( )
Equation 7: Load estimation equation
Poweri is predefined power associated with each VM. Load define burst time of job on cutting-edge vm. Storage in phrases of share is given as S(%)=(storagei/Total)*100

Equation 8: Storage in p.c shape
Storage indicates job storage requirement and complete suggests complete job storage requirements. The CO2 emitted by using VM is calculated the usage of the equation.

em_factori=(em_factori*Total_Load)/100;

Equation 8:

CO2 calculation em_factor indicates the CO2 stages emitted via VM. Total_Load is the sum of burst time possessed with the aid of every job.

Next section describes overall performance evaluation and results.

Performance Analysis and Result

Comparison
Performance of EECOC is evaluated the usage of the simulation conducted in CloudSim. Obtained results in phrases of a number of parameters are listed as follows.

Table 2: Results in terms of Load and Storage.
Load is calculated the use of equation 7 and Storage (%)is calculated in phrases of equation 8

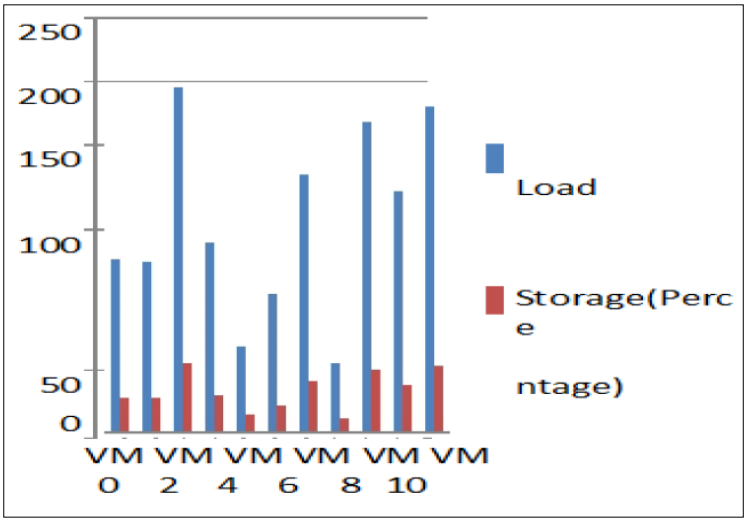


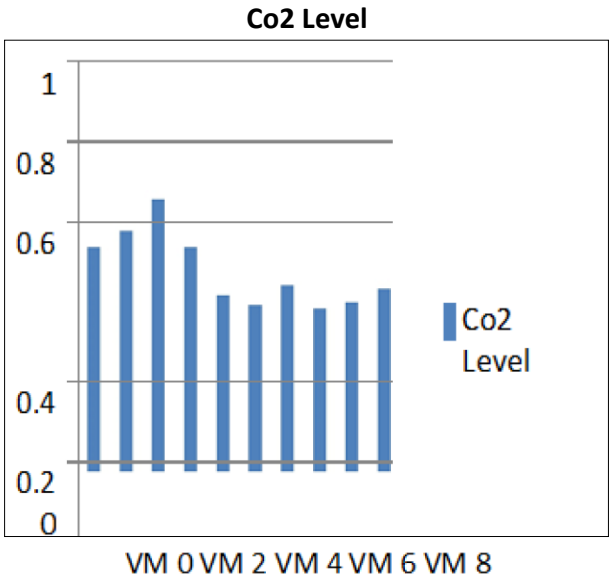
Figure 1: Plot of Load and storage present on VM.

The performance of EECOC is also evaluated in terms of CO2 levels. This is given as follows

Virtual machine	Co2 Level
VM 0	0.70
VM 3	0.70
VM 4	0.55
VM 5	0.52
VM 6	0.58
VM 7	0.51
VM 8	0.53
VM 9	0.57
VM 10	0.54

Table 3: CO2 levels associated with VMs in EECOC.

Plots corresponding to Table 3 are as follows



Virtual machine	Load	Storage (%)
VM 0	100	20.0
VM 1	99	19.8
VM 2	200	40.0
VM 3	110	21.0
VM 4	50	10.0
VM 5	80	15.5
VM 6	150	30.0
VM 7	40	8.0
VM 8	180	36.0
VM 9	140	28.0
VM 10	190	38.0

Figure 4: Comparison of EECOC with existing literature [18]

As indicated through obtained result, EECOC is performing higher and result is multiplied through 33%.

VI. CONCLUSION AND FUTURE SCOPE

The EECOC reduces CO2 degrees and bettering performance. Job allocation approach is formed via thinking about priority of the job submitted. Once job is submitted, VMs are analyzed for deterioration. As the VM is deteriorating, load is migrated from current laptop to subsequent desktop in sequence. Load is reduced notably on current computing device and load is balanced. Threshold fee of CO2 emitted by awesome machines is maintained. As the load is assigned on man or woman machine, CO2 stages emitted by using VM increases. However in EECOC these stages are minimal due to migration of load. N the future, different allocation techniques like shortest jobs first and round robin can be used along with EECOC to decide mechanism to lower CO2 tiers further.

References

[1] Shirmohammadi, Shervin, et al. "Introduction to the special section on visual computing in the cloud: Cloud gaming and virtualization." IEEE Transactions on Circuits and Systems for Video Technology 25.12 (2015): 1955-1959.

[2] Cui, Xiaolong, et al. "Shadow replication: An energy-aware, fault-tolerant computational model for green cloud computing." Energies 7.8 (2014): 5151-5176.

[3] Yadav, Vikram, et al. "Energy Efficient Data Center in Cloud Computing." Cloud Computing in Emerging Markets (CCEM), 2015 IEEE International Conference on. IEEE, 2015.

[4] Nagafuchi, Yukio, et al. "Routing optimization for live VM migration between datacenters." Information and Telecommunication Technologies (APSITT), 2015 10th Asia-Pacific Symposium on. IEEE, 2015.

[5] Dai, Xiangming, Jason Min Wang, and Brahim Bensaou. "Energy-efficient virtual machines scheduling in multi-tenant data centers." IEEE Transactions on Cloud Computing 4.2 (2016): 210-221.

[6] Devasvaran, V., NM Abdul Latiff, and NN Nik Abdul Malik. "Energy efficient protocol in wireless sensor networks using mobile base station." Telecommunication Technologies (ISTT), 2014 IEEE 2nd International Symposium on. IEEE, 2014.

[7] Kokilavani, T., and DI George Amalarethinam. "Reduced Makespan Task Scheduling Algorithm for Grid Computing." vol 9: 71-76.

[8] Hamscher, Volker, et al. "Evaluation of job-scheduling strategies for grid computing." International Workshop on Grid Computing. Springer, Berlin, Heidelberg, 2000.

[9] Wu, Chenggang, et al. "FPS: A fair-progress process scheduling policy on shared-memory multiprocessors." IEEE Transactions on Parallel and Distributed Systems 26.2 (2015): 444-454.

[10] You, Changsheng, Kaibin Huang, and Hyukjin Chae. "Energy efficient mobile cloud computing powered by wireless energy transfer." IEEE Journal on Selected Areas in Communications 34.5 (2016): 1757-1771.

[11] Dhuria, Ritika, and Kiranbir Kaur. "A HYBRID APPROACH FOR REDUCING DOWNTIME, MIGRATION TIME AND ENERGY CONSUMPTION OF LIVE VM MIGRATION." International Journal of Advanced Research in Computer Science 8.8 (2017).

[12] Zhang, Ke, et al. "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks." IEEE access 4 (2016): 5896-5907.

[13] Chen, Xu. "Decentralized computation offloading game for mobile cloud computing." IEEE Transactions on Parallel and Distributed Systems 26.4 (2015): 974-983.

[14] Aslanyan, Levon, Hakob Aslanyan, and Hamid Khosravi. "Optimal node scheduling for integrated connected-coverage in wireless sensor networks." 2013 Computer Science and Information Technologies (CSIT). IEEE, 2013.
[15] Wei, Tongquan, Xiaodao Chen, and Shiyan Hu. "Reliability-driven energy-efficient task scheduling for multiprocessor real-time systems." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 30.10 (2011): 1569-1573.
[16] Jiang, Bangjie, Junmin Wu, and Xiaodong Zhu. "Priority-based live migration of virtual machine." International Conference on Grid and Pervasive Computing. Springer, Berlin, Heidelberg, 2013.
[17] Wei, Tongquan, et al. "Fixed-priority allocation and scheduling for energy-efficient fault tolerance in hard real-time multiprocessor systems." IEEE Transactions on Parallel & Distributed Systems 11 (2008): 1511-1526.
[18] Wajid, Usman, et al. "On achieving energy efficiency and reducing CO 2 footprint in cloud computing." IEEE transactions on cloud computing 4.2 (2016): 138-151.