ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

INTERNATIONAL JOURNAL OF RESEARCH – GRANTHAALAYAH
A knowledge Repository

Science

THE ADVANTAGE OF GENETIC ALGORITHM IN ENERGY-EFFICIENT SCHEDULING FOR HETEROGENEOUS CLOUD COMPUTING

Hang Zhou [*1], Samina Kausar [2], Ningning Dong [3]
[*1, 2, 3] School of Computer Engineering and Science, Shanghai University, CHINA

DOI: https://doi.org/10.29121/granthaalayah.v4.i7.2016.2590

ABSTRACT

Nowadays Energy Consumption has been a heavy burden on the enterprise cloud computing infrastructure. This paper focuses on the hardware factors in energy consumption. Inspired by DVFS, it proposes a new energy-efficient (EE) model. This paper formulates the scheduling problem and genetic algorithm is applied to obtain higher efficiency value. Simulations are implemented to verify the advantage of genetic algorithm. In addition, the robustness of our strategy is validated by modifying the relevant parameters of the experiment.

Keywords:
Energy conservation; cloud computing; heterogeneous; genetic algorithm.

Cite This Article: Hang Zhou, Samina Kausar, Ningning Dong, "THE ADVANTAGE OF GENETIC ALGORITHM IN ENERGY-EFFICIENT SCHEDULING FOR HETEROGENEOUS CLOUD COMPUTING" International Journal of Research – Granthaalayah, Vol. 4, No. 7 (2016): 1-9.

1. INTRODUCTION

Recent years, with the demand for improvement of commercial and scientific computing capacity, cloud computing is developing constantly and has played an important role in social production and daily life. Although the power of a single computer is not high, the power consumption of cloud computing data center with large cluster is quite amazing. Some big enterprises, for example, Google consumed more than 260 million watt energy in 2011 which equals to the total civic consumption of Irvine city in California [1]. With the launch of Amazon EC2, Windows Azure and Google AppEngine, the energy consumption of these large-scale cloud computing infrastructures are growing year by year. On one hand, substantial growth in energy consumption will emit large amount of CO2 which intensifies the greenhouse effect; on the other hand, the price of hardware is coming down which leads the expenditure of growing energy consumption to be more significant. The CEMS project in Amazon shows that the cost of energy consumption has reached to 41.62% of total investment expenditure [2]. Therefore, the

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

study of how to reduce energy consumption and how to build green cloud computing has been a hotspot and a potential area.

2. RELATED WORKS

There are many researches on the algorithms and strategies in energy conservation. These studies can be grouped into three categories. The first method is dynamic voltage and frequency scaling (DVFS) technique. Based on the model: $P=CV^2F$, energy conservation can be achieved if we decrease corresponding frequency and voltage of server in idle state [3]. The second approach is virtual bin-packing algorithm. M. Cardosa and A. Singh achieve energy conservation by tradeoff between time and space (CPU, memory and disk I/O). The third way is virtual machine migration. When the VM's utilization of resources is not high or the number of active VMs is below a threshold, the migration of VMs can reduce energy consumption by increasing the utilization of resources in physical machine.

Although the above-mentioned strategies make some progress in this area, there are still some deficiencies. First of all, these previous models seem quite simple but lacking accuracy especially in heterogeneous scenarios. Studies in [3-4] hold the homogeneous precondition that all the server nodes have the same configuration. [5] assumed there is no energy consumption in the phase of turning on and off. [6] quantified the energy consumption by 3 key stages but there is shortage of precise calculation. The second, the performance of these strategies always depends on the specific platform thus having weaknesses in operation. In large heterogeneous cluster, it is difficult to make a unique metric to adjust voltage and frequency. The third, the performance fluctuation in terms of different configurations and different platforms makes it impossible to quantify the cost of VM migration with accuracy.

From the perspective of job scheduling, this paper tries to deal with energy conservation by scheduling strategy in heterogeneous cloud cluster scenarios. Job scheduling is not a new problem both in grid computing and cloud computing. [7] applies GA algorithm to assign jobs. Compared with FIFO scheduler and Fair scheduler, it shows the advantage of GA in scheduling. [8] illustrates the efficiency of PSO algorithm in job scheduling on CloudSim. [9] takes both energy and execution time into account and designs a dual fitness function for better performance. These above-mentioned algorithms achieve improvement than heuristic. However, some drawbacks are obvious. An intuitive and efficient model is needed to quantify the effectiveness of energy conservation. Furthermore, the granularity of most algorithms and strategies is 'job' rather than 'task'. In fact, there are three granularities which are Queue, Job and Task respectively [10]. 'Task' scheduling should be given more attention and this paper brings about a new energy-efficiency model to quantify the energy conservation in cloud computing. In order to make our approach more realistic, we limit the CPU utilization with SLA in heterogeneous cluster and verify the advantage of genetic algorithm in energy conservation.

3. ENERGY EFFICIENT MODEL

A reasonable energy efficiency model is a prerequisite for the evaluation of energy conservation. Some researchers designed model based on specific hardware resource. Using performance counter, Martonosi and Contreras brought about a linear model based on Intel XScale processor.

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

Memory also increases the energy cost and it was quantified by number of L1 misses. Depending on instruction read miss in cache and delay time [11] achieved high accuracy in energy efficiency. Besides above-mentioned method, there are two prevalent system models. One is that energy consumption is quantified by utilization of CPU. [12] performed some experiments which verified the accuracy of linear regression on CPU utilization. The other is DVFS which focuses on the running voltage and master frequency [13]. However, it may not be suitable to attribute energy consumption of a cluster to the system power of a server as decrease in power always accompanies with reduction of CPU utilization. The deadline of workloads will delay due to the low CPU utilization Therefore it may eventually lead to overall higher energy consumption. Compared with the quantity value of energy consumption, energy efficiency which represents the efficiency of using energy is more suitable to evaluate the energy conserving strategies in cloud data center. In general, energy efficiency is the ratio of effective energy and overall consumption [14]. FlOPS (floating-point operations per second) is the metric of efficiency while Watt is the unit of energy consumption of a server in unit time. The energy efficiency of a server can be defined as equation (1).

$$\eta = \frac{\text{FLOPS}}{Watt} = \frac{OPS}{\text{Sec}ond \times Watt} = \frac{OPS}{Joule} \quad (1)$$

where OPS is the number of float calculation per second. OPS is obvious related to the frequency of CPU and it increases with higher CPU frequency. Nevertheless, the value of OPS varies with different number of assembly line of computing and different number of clock cycles in heterogeneous scenarios. In addition, the value of OPS is too high and a workload unit (U) is introduced in this paper to reduce the magnitude of value.

We define U as the Unit of workload which equals to the computation work of a server with 1 Ghz frequency in 1 second. Then the energy efficiency of a server during t seconds is like this $\eta(t) = {L(t)}/{E(t)}$. In current time-sharing operation system, the number of executions per cycle is related with CPU utilization. Therefore the value of energy efficiency has relationship with CPU utilization besides frequency. $L(t) = f(t) \times \omega(t) \times \lambda$ where f(t) means the frequency of CPU at t and ω(t) represents the utilization at t. λ is a parameter of CPU which varies with different processes and standards.

$$\eta(t) = \frac{L(t)}{E(t)} = \frac{f(t) \times \omega(t) \times \lambda}{P(t)} \quad (2)$$

P(t) is the power of a server at time t. Some studies have demonstrated the linear relationship between power and CPU utilization [15] which can be formulated as $p(t) = A\omega(t) + B$ Then the expression of energy efficiency is equation (3).

$$\eta(t) = \frac{f(t) \times \omega(t) \times \lambda}{A\omega(t) + B} \quad (3)$$

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

For better formulation, two factors should be taken into consideration. The first factor is dynamic of CPU utilization. As it is not a fixed value during working, we introduce the concept of integral in expression (4).

$$\eta(t) = \frac{\int_0^T f(t) \times \omega(t) \times \lambda}{\int_0^T A\omega(t) + B} \quad (4)$$

Considering the scenario of multiple servers in cluster (N nodes in a cluster), the energy efficiency of the cluster is defined as equation (5).

$$\eta(\mathrm{T}) = \frac{\sum_{i=1}^{N} \int_0^T f(t)\omega(t)\lambda dt}{\sum_{i=1}^{N} \int_0^T (A\omega(t) + B)dt} \quad (5)$$

Higher value of η(T) means high efficiency of energy thus fewer energy will be consumed. Compared with power (P), efficiency η(T) is a better parameter.

4. CONTEXT AND FORMULATION

Here are some points in real scenarios in cloud data center.

Heterogeneity. The number of nodes in a cluster may reach thousands thus heterogeneity is inevitable. However, servers with same configuration are purchased with bulk to unify management.

Diversity in utilization of CPU. Equation (5) shows that the value of efficiency depends on the utilization of CPU. When assigning jobs to nodes, the utilizations of all CPUs in nodes should be taken into consideration.

Constraints in SLA. Confined by specific SLA, every type of server has its optimal utilization of CPU. If the actual value is lower than it, the capacity of computation will be wasted; otherwise, resource contention will be introduced and leads the incapacity to meet SLA.

Granularity. Fine granularity should be adopted to get higher prediction. In Map-reduce mode, big jobs are divided into tasks which will be assigned to nodes eventually. For instance, data are split into N elements (64M) in Hadoop platform.

Set N to be the number of servers in the cloud dada center DC. There are M types of nodes in DC and the corresponding CPU frequency, CPU utilization and CPU parameter are $f_k(t)$, ω_k and $\lambda(k)$ (k=1, 2, …, M) respectively. Job J is divided into D tasks S (S_1, S_2, …, S_d) in Map-Reduce. The objective is to find an optimal scheduling strategy which maximize the value of η(T). Set ω_{pk} (k=1, 2, …,M) to be the current CPU utilization of M types of servers whose optimal CPU utilization is ω_{ok} (k=1, 2, …, M) respectively. The number of tasks which are assigned to nodes

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

is m_k (k=1, 2, …, M). CPU utilization of a node increases when tasks are assigned to it. To the same task, obviously the increment of M is different with different types of nodes. Set the increment to ω_{TK} (k=1, 2, …, M) respectively, then $\omega_{FK} = \omega_{PK} + m_k\omega_{TK}$ in which ω_{FK} means the CPU utilization of node k after receiving corresponding tasks. Then the overall energy efficiency objective is equation (6).

$$\eta(\mathrm{T}) = \frac{\sum_{i=1}^{N}\int_0^T f_k(t)(\omega_{pk} + m_k\omega_{Tk})\lambda_k dt}{\sum_{i=1}^{N}\int_0^T (A(\omega_{pk} + m_k\omega_{Tk}) + B)dt} \quad (6)$$

The running frequency, CPU utilization and CPU parameter, etc., can be measured directly or experimentally. In addition, the overall value of efficiency is always consistent with narrow band. Therefore, eventually the formulation of optimal scheduling can be described as follows:
Maximize η(T)

$$s.t.\begin{cases} K_1 + K_2 + K_3 + ... + K_M = D \\ \eta(\mathrm{T}) = \dfrac{\sum_{i=1}^{N}\int_0^T f_k(t)(\omega_{pk} + m_k\omega_{Tk})\lambda_k dt}{\sum_{i=1}^{N}\int_0^T (A(\omega_{pk} + m_k\omega_{Tk}) + B)dt} \\ \omega_{Pk} + m_k\omega_{Tk} <= \omega_{Ok}(k = 1, 2, ..., M) \end{cases} \quad (7)$$

This shows that the objective is to seek the maximum value of a nonlinear function with M arguments and some constraints.

5. ALGORITHM AND SIMULATION

5.1.ALGORITHM DESIGN

a. Coding.

Binary encoding ω_{Fk} and the coding length depends on both the difference of ω_{ok} and ω_{pk} and the accuracy of CPU utilization.

$$(v)_B = (\frac{\omega_{Ok} - \omega_{Pk}}{p})_D$$

Here the length of v is just the length of binary coding.

b. Chromosome Design

A complete chromosome should be able to represent all elements in an optimized scheduling K. Equation (7) shows that the value of fitness can be fixed by elimination method. Therefore the binary code of ω_{Fk} of M-1 nodes should be included in the chromosome.

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

c. Population Initialization

M-1 binary numbers are generated randomly and connected into a chromosome. Set the number of gens to length. The number of individual in the population is NP. The population is initialized by a double loop where the number of executions in outer and inner loop is NP and length respectively.

d. Fitness Function Design

As the objective is to get the maximum value of the function, the fitness function is the objective function itself. For simplification, ω_{Fk} is used instead of mk. Elimination is completed to fit the strong constraint in equation (7). Then the fitness function F can be presented like this.

$$F = (D - \sum_{k=1}^{M-1} m_k)\omega_{Tk}\lambda_k) / (\sum_{i=1}^{N-1} f_k(t)(\omega_{Pk} + m_k\omega_{Tk})\lambda_k) + f_k(t)(\omega_{Pk} + A(\omega_{Pk} + m_k\omega_{Tk}) + B) + A(\omega_{Pk} + (D - \sum_{k=1}^{M-1} m_k)\omega_{Tk} + B) \quad (8)$$

e. Crossover

Father and mother chromosomes are derived by the way of roulette wheel selection. Also we choose the position of crossover by random. Note that only the first half of binary code is taken into account as the crossover in the second half of binary code always leads to local optimal value.

f. Differentiation

Firstly we get the position of differentiation by random. Then we do the bitwise operations in the position as indicated previously. The suitable probability of differentiation can be achieved by actual hardware configuration. Too high or too low probability is appropriate.

g. Fitness Computation

After the process of heredity and variation above, a new individual is created. All individuals of next generation will be ready after NP times loop. Then all values of fitness of these individuals are calculated by equation (8). Better individuals are selected and iterations are repeated between step d and step g.

Table 1: Parameters and Configurations

Type	n	f/GHz	A	B	ωPk	ωOk	ωTk
I	50	2.7	42.75	53.75	0.1	0.5	0.0025
II	30	2.2	37.36	52.68	0.2	0.6	0.0030
III	20	3.0	49.80	55.74	0.3	0.8	0.0020

5.2.SIMULATION

The platform for simulation is CloudSim 3.0 in which 100 server nodes with 3 different hardware configurations are created. Set CPU parameter λ to 1 for simplicity. Other parameters are shown in table 1 in which f, A, B and ω_{TK} are from configuration of real servers.

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

Set the number of tasks to 10000, NP=50, the maximum number of generations (NG) to 100, crossover probability Pc=0.9, differentiation probability Pm= 0.1, the accuracy value of CPU metric (eps) is 0.01. Algorithm in section IV is applied in this simulation with the parameter data in table 1. Fig. 1 shows the data after 100 iterations in which the efficiency value of GA scheduling is higher than the value of default scheduling strategy. ω_{F1}=0.2976, ω_{F2}=0.5765, ω_{F3}=0.5282 when we get the maximum value of η(t). Meanwhile, m_1=80, m_2= 125, m_3=114 which means that the number of tasks assigned to 3 types of servers are 80, 125, 114 respectively. Considering the effect of energy conservation, the efficiency value of best scheduling is 16.6 which is 7.5% higher than the default scheduling. If GA scheduling is applied consistently, over 10% energy efficiency can be achieved and this is a significant result.

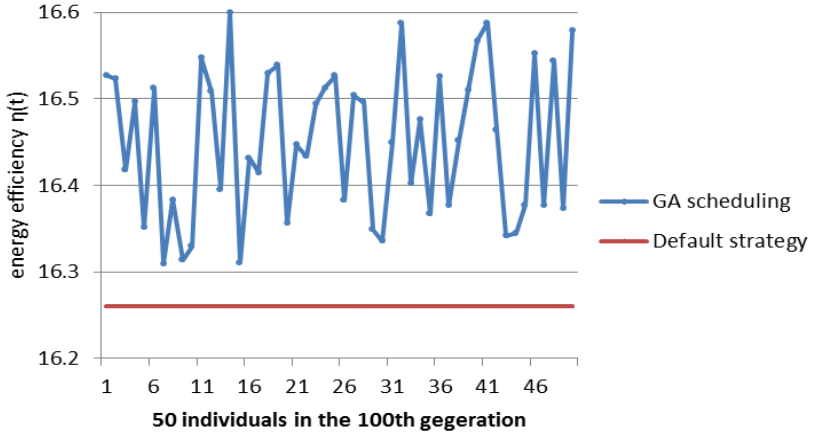

Figure 1: energy efficiency comparison

In order to verify the robustness of GA in scheduling, we test the effect with variance of CPU utilization and hardware configuration. Firstly, we change the current CPU utilization of 3 types of servers. The values of ω_{P1}, ω_{P2}, ω_{P3} are set to 0.3, 0.2 and 0.1 respectively in the simulation. Still we apply GA algorithm with others and keep the parameters unchanged. Fig. 2 shows the maximum efficiency value is 17.8. All 50 efficiency values are higher than the value of default strategy. The average advantage in energy conservation is 7.2% which is also significant. The best scheduling approach is ω_{F1}=0.4937, ω_{F2}=0.3976, ω_{F3}=0.5149. The corresponding assignment is m_1=77, m_2=66, m_3=207. To a specific type of server node, once the running frequency is fixed, then the efficiency value can be derived like this.

$$\eta(t) = \frac{\lambda f(t)\omega(t)}{A\omega(t)+B} = \frac{\lambda f(t)}{A+\frac{B}{\omega(t)}} \quad (9)$$

Equation (9) shows obviously that the value of efficiency will increase with the increment of CPU utilization. If we compare these two above-cited best scheduling approaches, we find that the number of tasks assigned to type III server increases significantly. The reason is that the current CPU utilization drops in the second simulation and thus more tasks should be assigned to obtain higher efficiency value.

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

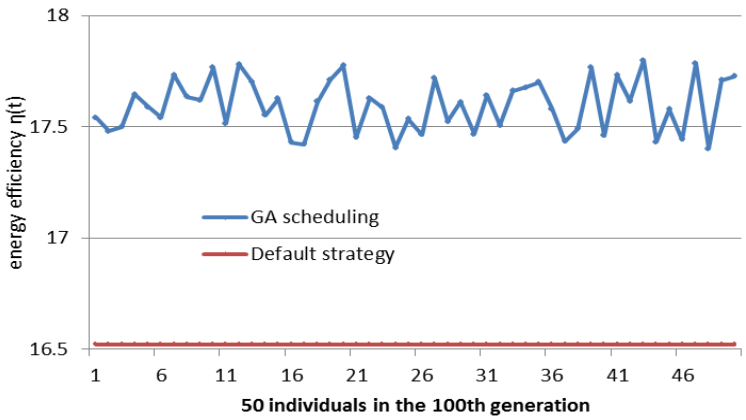

Figure 2: energy efficiency comparison with CPU variance

We also take the simulation with the variance of number of servers. We set n_1=20, n_2=30, n_3=50 while maintaining the values of other parameters. The result is that GA can obtain 13.1% higher efficiency value averagely. It demonstrates the rule that server type with larger number of nodes should be assigned more tasks for higher energy efficiency.

6. CONCLUSION

Based on the principles of time-sharing system and general formula of energy consumption, this paper introduces a special energy efficiency model. In this model the value of efficiency depends on related parameters such as CPU utilization, running frequency, etc.
Simulation experiments are implemented to compare the efficiency value between GA scheduling and default strategy in Hadoop. The results show that GA outperforms default strategy significantly after enough number of iterations. Variances on CPU utilization and number of servers are also taken in the simulation to show the robustness of GA.

7. ACKNOWLEDGEMENTS

Authors thank Henry Brewer for his kind hearted assistance in experiments.

8. *REFERENCES*

[1] Brown R. Report to congress on server and data center energy efficiency public law. Ernest Orlando Lawrence Berkeley National Laboratory, Berkeley : LBNL-363E, 2007, 109–431.

[2] Hamilton J. Cooperative expendable micro-slice servers (CEMS) : low cost, low power servers for internet-scale services. California, 4th Biennial Conference on Innovative Data Systems Research (CIDR), 2009, 93–98

[3] L. Wang, G. Von Laszewski, J. Dayal. Towards energy aware scheduling for precedence constrained parallel tasks in a cluster with dvfs. 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid). IEEE, 2010, 68–137.

ISSN- 2350-0530(O) ISSN- 2394-3629(P)
IF: 4.321 (CosmosImpactFactor), 2.532 (I2OR)

[4] Michael Cardosa, Aameek Singh. Exploiting Spatio-Temporal Tradeoffs for Energy-Aware MapReduce in the Cloud. IEEE TRANSACTIONS ON COMPUTERS, 2012, 61(12): 1737-1751

[5] P. Graubner, M. Schmidtand, B. Freisleben. Energy-efficient Management of Virtua Machines in Eucalyptus. Proceedings of the 4th IEEE International Conferenceon Cloud Computing, 2011, 243–250

[6] B. Yin, Y. Wang. A New Virtual Machine Migration Strategy Based on Migration Cost and Communication Cost for Power Saving in Cloud. Journal of Beijing University of Posts and Telecommunications, 2012, 35(1): 68-71.

[7] Y. J. Ge, G. Y. Wei. GA-Based Task Scheduler for the Cloud Computing Systems Web Information Systems and Mining (WISM). 2010 International Conference on Date of Conference, 2010, 181–186.

[8] L. Zhang, Y. Chen. A task scheduling algorithm based on PSO for grid computing. International Journal of Computational Intelligence research 2008, 4(1): 37–43.

[9] C. T. Ying, J. Yu. Energy-aware task scheduling algorithms in cloud computing. MicroElectronics & Computer, 2012, 29(5): 188–192.

[10] Leverich J, Kozyrakis C. On the energy efficiency of Hadoop cluster. In Newsletter ACM SIGOPS Operating Systems Review archive, 2010, 4(1): 61–65.

[11] Contreras G, Martonosi M. Power predicti on for Inter XScale processors using performance monitoring unit events. ISLEPD, IEEE, 2005, 27(3): 221–226.

[12] Fan X, Heath T. Power provisioning for a warehouse-sized computer. In ACM SIGARCH Computer Architecture News, New York.

[13] Economou D, Rivoire S, Kozyrakis C. Full-system power analysis and modeling for server environments. In 2nd Workshop on Modeling, Benchmarking, and Simulation (MoBS). Palo Alto: Stanford Press, 2006, 22-27.

[14] J. Song, T.T. Li, Y.Z. Xing. Energy-Efficiency Model and Measuring Approach for Cloud Computing. Journal of Software, 2012, 23(2): 200-214.

[15] T. Li, L. Jone. Run-time modeling and estimation of operating system power consumtio. Measurement and Modeling of Computer System, 2003, 37(2): 160-171