TOWARDS ENERGY EFFICIENT HIGH PERFORMANCE COMPUTING PERCEPTIONS, HURDLES & SOLUTIONS

Muhammad Zakarya[1], Izaz Ur Rahman[2]

Abstract

Global Warming (GW[1]) and Energy Crisis (EC[2]) have forced the researchers at teaching institutes, organization, research laboratories and other academia to study and minimize the power requirements of digital and electronic devices especially the huge amount of computers in the global village. With the rise of new computing era i.e. Green computing there is a need to reduce the power consumption in High Performance Computing (HPC[3]) like Clusters, Grids (GC[5]) and Clouds (CC[6]). This research work will study HPC and green computing (GC[4]) and will result in an integrative solution to reduce the energy consumed by processors in HPC. In recent times it is realized that there is a need for energy reduction in processors. And a lot of work has been done on minimizing the energy expenditures and use. When we reduce the energy consumption then the response time is increased. And it will degrade the performance of processor and the underlying real time systems. The same scenario is also applicable in HPC. The processors taking part in a cluster, grid or cloud environment are to be energy efficient without losing its peak performance. In our work we will try to study and propose some solutions that reduce the power consumption of a HPC, keeping its performance and response time to the best level [1].

Keywords Global Warming, Energy Crisis, High Performance Computing, Green Computing, Grid Computing,Cloud Computing, Static Power Management, Dynamic Power Management

Introduction & Concepts

The reliability of distributed systems is superior to monolithic single processor machines, which makes distributed systems a most widely usable technology. In such systems single failure of one network node does not prevent the whole process from completion as it happens in single CPU resource. Distributed computing prototype knot jointly the power of huge number of resources dispersed across a network. Millions of processors linked across the Internet are frequently unused and can become a part of cooperative computing. A distributed system has the capability to answer the problem alternatively. The chief endeavor of *Cluster Computing* is to propose a competent hardware media, networks, and software to develop and improve the performance and accessibility of computing platform that uses a collection of commodity computer resources integrated through some a single computer resource. The concept of *Grid Computing* is based on using the high speed connection network i.e. Internet as a medium for the wide spread availability of powerful computing resources as low-cost commodity components *[8]*. *Cloud Computing* describes a new supplement, consumption, and delivery model for IT services based on the Internet, and it typically involves over-the-Internet provision of dynamically scalable and often virtualized resources *[2]*. One of the major issues in computer system is reducing the energy consumption. The energy consumption should be reduced due to the operational cost. For example about 25% of the operational cost is spent on air conditions, backup cooling and power release systems. The entire power dissipation for desktop system is 160MW in 1992, and in 2001 it is increased to 9000MW *[13]*.

[1]Department of Computer Science, Abdul Wali Khan University (AWKU), Mardan, Khyber Pakhtun Khwa (***KPK***), Pakistan
mohd.zakarya@awkum.edu.pk
[2]School of Information Systems, Computing and Mathematics,
Brunel International University, London, UK
izaz.rahman@brunel.ac.uk

When the power consumption of the system is increased, then it would produce a lot of heat and according to the data from two popular vendors the rate of failure would be doubled, when there as an increase of 10^0C *[14]* and the cost of the system would also be increased, because we need a complex cooling system to deal with it according to the data from Intel Corporation when the power dissipation is more than 35-40W, then it would need more than 1W per CPU chip *[15]*. For battery powered, advanced Personal Digital Assistants (PDAs), cell phones, laptops and pocket PC's, when the system is exactly more than 500MIPS then the power consumption should be best if it is kept at or below 500MW *[16]*.

One solution to this problem would be to increase the battery capacity but it would not work because the battery capacity is increasing only 5-10% annually while the power required by the system is increasing day by day. Much slower than what is needed to support ever increasing processor power *[17]*. Therefore the solution of the problem would be to decrease the energy consumption by using some energy efficient techniques. Techniques that are used for the reduction in energy consumption are the same as that are used to increase the battery capacity.

The energy expenditure & utilization can be diminished either by hardware or through the software perception. Through the hardware viewpoint, numerous power states are incorporated into the micro-architect, circuit land, and device level. Many engineering standards are made, such as ACPI[3] and the APM[4] specification, from the software point of view, the power shutdown procedures can be used that makes the system sleep or stop the carrying out of task when the system is idle. But still this software and wakeup operation has a high overhead, so it can not be used in many situations. When the system goes to sleep mod then the response time of the system would be slower. When the system does not have continuous idle time, then this technique can not be used. Then some other sophisticated approaches are used such as DVS[5].

The internet and open nature of the CC makes it vulnerable to security issues. Table 1 summarizes some common features of three major HPC systems. One of the most important dissimilarity between CC and cluster computing is that clouds frequently take the shape of web-based applications that can be accessed using internet browser *[3]*.

The rest of the paper is structured as follows. Section I introduced HPC i.e. Cluster Computing, GC and CC. We conclude in section VIII, with some future directions and work in subsequent section IX.

Table 1. Cluster, Grid and Cloud Systems [2]

Feature	**Cluster**	**Grid**	**Cloud**
Size	Small	Large	Small to large
Network	LAN	WAN	WAN
System cost	Very High	High	Very Low
Network Type	Private	Private	Public
Resource Support	Homogeneous	Heterogeneous	Heterogeneous
Security	Medium	High	Low
Initial Infrastructure Cost	Very high	High	Low
Administrative Domain	Single	Multi	Both
QoS	Excellent	Good	Moderate
Availability	Very High	High	High
Reliability	High	Moderate	Low
Fault Tolerance	High	Good	Good

Related Work

In *[3]* the authors divide power management into two diverse mechanisms: (a) SPM[1] technique with the intention of utilization low-power apparatus to save power and (b) DPM[2] technique which utilize software and power-scalable apparatus to save power.

Energy efficiency can assist to improve HPC reliability by diminishing heat quantity in the system. We know that computing at higher temperature is more erroneous and result in less reliability. According to Arrhenius equation (Eq. 1) components failure ratio doubles with every 18^o F increase in temperature *[5]*.

$$k = A * e^{-\frac{E_a}{R \cdot T}} \qquad (1)$$

Where ***k*** is the rate constant, ***A*** is the pre-exponential factor, ***Ea*** is the activation energy, ***R*** is the gas constant, and ***T*** is the absolute temperature. The quantity of energy (E) consumed in HPC system over a time period (T) is equivalent to the product of the time period T and the average system power (P) consumed over the time interval T. The relation between power and energy is shown in equation (Eq. 2).

$$E = P * T. \qquad (2)$$

It is clear from Eq. 2, if we reduce power P or time interval T, then overall energy consumed is also diminished. The prospective hazard of existing SPM mechanisms is that humanizing energy efficiency via low-power components is expensive.

On the other hand DPM mechanisms have revealed swear for civilizing energy efficiency; yet, scheming power-aware schedulers is not slight. Energy savings diverge considerably with applications, workload, and scheduling tactic *[6]*.

In *[7]* the authors proposed an approach that implements DVFS[3] to safeguard processor power expenditure during communication, and uses load balancing mechanism to on-line and off-line memory reducing power consumption.

HPC are multiprocessor that can be defined as a computer system having more than one processor, each one sharing system main memory & some peripherals, to concurrently process and execute programs. A lot of work on energy efficiency is related to the tasks scheduling algorithms. The concept behind the scheduling mechanism is to assign tasks and processes to processors based on their execution speed and other properties. It enables some processor in idle mode and hence results in energy saving. The scheduling algorithms optimal on uniprocessor machines are not subject to be optimal on multiprocessor machines. So for multiprocessors are concerned, we use different scheduling algorithms like DM[4], RM[5], and EDF[6]. In *[9]* the authors reduced the response time using concept of aperiodic server.

The periodic server or total bandwidth server services aperiodic requests as soon as possible. The aperiodic server is consisting of a period and fixed execution time called server capacity and is scheduled with the same algorithm that is used for the periodic tasks. There are two types of scheduling techniques for multiprocessors i.e. HPC. In partitioned scheduling, each task is assigned to a specific processor and then it is executed on that processor without migration. These processors are then scheduled independently and separately.

This reduces the multiprocessor scheduling into a set of uniprocessor scheduling. With this scheduling we can use an optimal uniprocessor scheduling algorithm for multiprocessor systems. The alternative is the global scheduling, in which all the tasks are stored in a single priority queue. The scheduler selects the task having the high priority for execution. In global scheduling the tasks are not fasten to a particular processor and it can be executed on any processor.

In *[10]* the authors proposed DVS. They claim that for energy reduction we can use the DVS in latest processors. It means that power is a linear function of frequency i.e. f and a quadratic function of the voltage i.e. V given by

$$p \propto fV^2 \qquad (3)$$

The voltage adjustment at an instant of time is called DVS, which is an effective way for power saving in current HPC systems. In recent processors the relationship between frequency f and power p gives foundation to Dynamic Voltage Scaling explained in equation 4.

$$E = Pt \qquad (4)$$

Where E is energy consumed, t is time taken and P is power consumed. We can achieve the low performance by simply reducing the operating frequency of the processor when the peak speed is not required. As a result DVS scales the operating voltage of the processor along with the frequency. DVS is a standard for managing the power consumption of a system. It is based on the fact that the *dynamic* (switching) power P of CMOS circuits is strongly dependent on the core voltage V and the clock frequency f. It is shown that the execution time is inversely proportional to the frequency and thus, the total energy E for the computation is then proportional to the square of the voltage.

$$E = v^2 \qquad (5)$$

The average power dissipation in processor is:

$$P_{avg} = P_{capt} + P_l + P_{stdby} + P_{sc} \qquad (6)$$

Where P_{capt}, P_l, P_{stdby}, and P_{sc} is capacitance, leakage, standby and short circuit power. The P_l, P_{stdby} and P_{sc} are important but they are least important as compared to P_{capt}. So we will not consider P_l, P_{stdby}, and P_{sc}. So the P_{capt} is equal to:

$$P_{capt} = \alpha C V_d^2 f \qquad (7)$$

Where α is the transition activity dependant parameter, C, V_d and f is switched capacitance, supply voltage, and clock frequency. Equation (3) shows that the supply voltage V_d is quadratic as compared to clock frequency f ; furthermore it also shows that lowering the supply voltage would be the most efficient way to reduce the power consumption. But when V_d is reduced then the circuit delay t_d would be increased:

$$t_d = \frac{mV_d}{(V_d - V_{tv})^2} \qquad (8)$$

Where t_{delay} threshold voltage and m is is a constant which will depend on gate size and capacitance. As from equation (8) the f and t_d are inversely proportional, so it would mean that the energy expenditure would be reduced in CMOS devices at the expense of performance delay.

The frequency f is:

$$f = \frac{(V_d - V_{tv})^2}{kV_d} \qquad (9)$$

Equation (9) shows that the clock frequency is directly proportional to supply voltage. If we would consider $P = P_{capt}$, then equation (7) can be written as:

$$P = \alpha C V^2 f \qquad (10)$$

Equation (10) shows that when the clock speed f and voltage is changed then it would effect power consumption linearly and quadratically, respectively.

Problem Statement

HPC is a great source for today's communication. HPC has altered the face of computing and presented rapid and clear-cut solutions for a diversity of complex issues & problems for diverse disciplines and fields. With the rise of new computing era distributed systems like Cluster, Grid, Cloud systems, there is a need to implement cloud computing over the sheer amount of data in bioinformatics. Researchers are trying to implement HPC over huge amount of data, making the data easily accessible, thus increasing the HA. But still its implementation gives birth to other problems like heat and cooling cost. We need some mechanisms to reduce the amount of heat that is produced during operation i.e. OPEx and cooling cost. The area is still in infancy and needs researcher's attention.

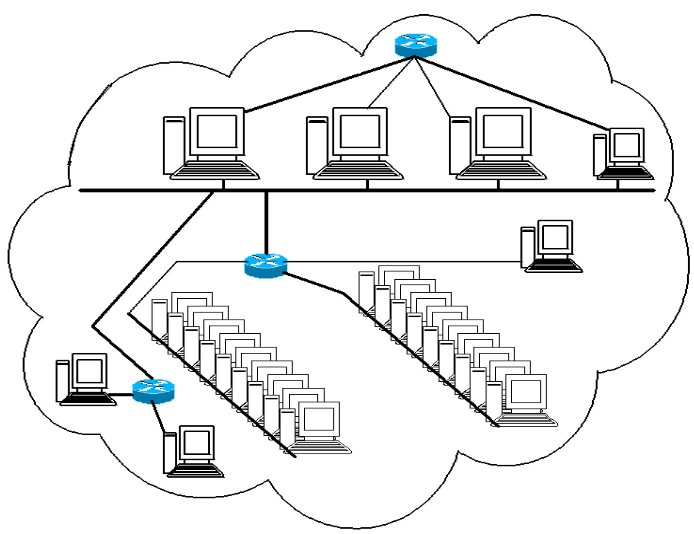

Figure 1. High Performance Computing

In *[30]* the authors have sketched an overview of existing mechanism for energy and power reduction in cloud data centers. According to them techniques like Power-aware scheduling of virtual machines in DVFS-enabled clusters, Energy-efficient management of data centre resources for cloud computing: a vision, architectural elements, and open challenges *[31]*, Machine learning, Green scheduling algorithm for energy savings in Cloud computing, GreenCloud: a new architecture for green data center *[32]*, Performance and energy-aware cluster level scheduling of compute-intensive jobs with unknown service times, Allocation and migration policies considering workload types and behaviors, Dynamic selection of physical nodes in heterogeneous environments, Prediction mechanisms for a smart workload distribution, Improved resource monitoring and Live migration's overhead reduction mechanisms are subject to energy reduction but have not mainly focused on performance and QoS. Some approaches have arisen frustrating to diminish energy spending at cloud computing data centers. On the other hand, cloud providers are in want of mechanisms and techniques not only for sinking energy eating to support the offered prices and demand but also for accomplishing with the required QoS to guarantee the customer happiness. From their analysis it is probable to wind up that there still stay alive some gaps that must be sheltered to attain the energy-performance stability that is essential in cloud computing environments and other HPC.

Discussion

Climate change and environmental damage due to emission of CO_2, cost savings, rising energy costs, reliability of power, power crisis, and heat produced by HPC and datacenter are major issues that have forced researchers at academia and research laboratories to study and propose power diminishing mechanisms *[22]*. In [22] the author has proposed some mechanism to reduce the power requirements. Energy efficient processors, multi-core processors, virtualization, storage area network (SAN) and some proper planning like performance, capacity, reliability considerations, cooling considerations, maximization of available resources and designing planning are the major solutions that are proposed to make the systems energy efficient. Code and data migration has been one of the most challenging criteria in the filed of distributed systems.

In *[18]* the authors have discussed advantages of migration including load sharing, load balancing, communication performance, availability, fault tolerance, and utilizing special capabilities. In a distributed system network i.e. HPC, Cluster, Grid, Cloud, the procedures usually communicate with each other by means of remote procedure calls (RPC). As a result, the consumed energy by message passing can be outstandingly condensed with the help of code and data migration that is to say putting more communicating codes in close proximity to each other. In *[18]* the authors implemented data migration technique to wireless ad-hoc sensor networks.

We observed that the same mechanism can also be implemented in HPC like Cluster, Grid or Cloud systems. In this way the energy consumed in the network channels i.e. media can be diminished to some limit.

The total energy consumed is given by Eq 11 where d_s is the distance among two HPC systems.

$$EC = \frac{1}{2}\sum_{u \in G}\sum_{v \in G} w_G(u,v).d_S(F(u),F(v)) \qquad (11)$$

The above equation can be written as Eq 12 where d represents the Euclidian distance among two points i.e. HPC systems. In order to reduce the value presented by Eq 11 derivative method is used.

$$EC = \frac{1}{2}\sum_{u \in G}\sum_{v \in G} w_G(u,v).d(F'(u),F'(v)) \qquad (12)$$

Using Euclid Formula for the distance of two points in the field we conclude:

$$EC = \frac{1}{2}\sum_{u \in G}\sum_{v \in G} w_G(u,v).\sqrt{(x_u - x_v)^2 + (y_u - y_v)^2} \qquad (13)$$

Where x_u, y_u, x_v, y_v, represents different dimensions of the points u, v respectively. The partial derivatives according to x, y are used to find the most favorable point of the function. Here the computation process for x_u is presented.

$$\frac{\partial EC}{\partial x_u} = \frac{1}{2}\sum_{v\in G}\frac{w_G(u,v).(x_u - x_v)}{\sqrt{(x_u - x_v)^2 + (y_u - y_v)^2}} = 0 \qquad (14)$$

The coefficient a, is defined as follows:

$$a_{u,v} = \frac{w_G(u,v)}{\sqrt{(x_u - x_v)^2 + (y_u - y_v)^2}} \qquad (15)$$

Simplifying equation (14) with respect to definition of Eq (15), it could be deduced that the optimal energy consumption, EC, formulated in (12), can be reached by positioning the procedures at the solution points of the following system of equation:

$$x_u = \left(\sum_{v\in G} a_{u,v}.x_v\right) \Big/ \sum_{v\in G} a_{u,v} \qquad (16)$$

$$y_u = \left(\sum_{v\in G} a_{u,v}.y_v\right) \Big/ \sum_{v\in G} a_{u,v} \qquad (17)$$

It is worth noting that in this system of equations, for each unknown value x_u or y_u an equation exists. There exist diverse numerical methods for solving system of equations like (16) and (17). Jacobi and Gauss-Seidel are two well-known iterative techniques *[19, 20]* that are used to solve identical systems. Due to above discussions, a simple algorithm can be wished-for to discover a placement for procedures that minimizes EC. In a distributed approach, the time is divided into equivalent sized divisions, called epochs. Then, at the start of each time division each node computes its destination coordinates with respect to formula (16), and (17) *[21]*. In other words, in each step, the procedure computes the weighted average of its incoming and outgoing remote procedure calls in the field, and starts moving toward there.

Conclusion

HPC is currently an area of great academic and industrial interest; the concept of being able to dynamically scale the amount of computational resources available to an organization on demand offers great benefits to individual companies and economic opportunities for high performance services providers. However, in order to reduce costs to all parties, it is necessary to reduce the power consumption overhead of a HPC *[11, 12]*. We can minimize the energy consumption at OSI layer with different mechanisms. For example at Hardware layer we can use power efficient circuits, smarter antennas and better batteries. At Link or Physical layer we must try to reduce the number of collisions and resends requests. At Network layer we must implement some energy conserving topology and / or energy conserving message transfer. Similarly at higher layers we have to modify transport and application layers according to our criterion.

Future Work

In HPC the main issue is heating and energy conservation. Our goal is to minimize the energy consumption so that the cooling cost will be reduced. Scheduling periodic and aperiodic tasks such that the load is balanced among different processors and the energy consumption is reduced, is a major concern and an active research topic. Runtime power reduction mechanisms can also reduce the energy expenditure to some extent. Some approaches have arisen frustrating to diminish energy spending at HPC but still HPC providers are in want of mechanisms and techniques not only for sinking energy eating but also for accomplishing with the required QoS to guarantee the customer happiness. There still stay alive some gaps that must be sheltered to attain the energy performance stability that is essential in HPC

Acknowledgment
This work is fully supported by Abdul Wali Khan University, Mardan, Khyber Pakhtun Khwa (KPK), Pakistan. The author(s) of this article are greatly thankful to SAIMS i.e. Society for Advancement & Integration of Multiple Sciences for full guidance and major support. iFuture is also given a number of credits for arranging seminars on the subject matter. iFuture is a Research Group in the Department of Computer Science, Abdul Wali Khan University, Mardan.

Nomenclature

SPM **S**tatic **P**ower **M**anagement
DPM **D**ynamic **P**ower **M**anagement
ACPI **A**dvanced **C**onfiguration & **P**ower **I**nterface
APM **A**dvanced **P**ower **M**anagement
DVS **D**ynamic **V**oltage **S**caling
DVFS **D**ynamic **V**oltage & **F**requency **S**caling
DM **D**eadline **M**onotonic
RM **R**ate **M**onotonic
EDF **E**arliest **D**eadline **F**irst

References

[1] Muhammad Zakarya, Izaz Ur Rahman, Mukhtaj Khan, 2011 Cloud QoS, High Availability & Service Security Issues with Solutions, BUJICT
[2] Hameed Hussain, Nasro Min-Allah, Samee Ullah Khan, A Survey on Resource Allocation in High Performance Distributed Computing Systems
[3] Walter Lassonde, Samee Ullah Khan, Nasro Min-Allah, An Overview of Achieving Energy Efficiency in Cluster Computing Systems, NDSU
[4] Dzmitry Kliazovich, Pascal Bouvry, Yury Audzevich, Samee Ullah Khan, GreenCloud. 2010 A Packet-level Simulator of Energy-aware Cloud Computing Data Centers, IEEE Globecom proceedings
[5] K. W. Cameron, R. Ge, and X. Feng., High-performance, Power-aware Distributed Computing for Scientific Applications,
[6] R. Ge, X. Feng, and K. W.Cameron. 05, November 2005 "Performance Constrained Distributed DVS Scheduling for Scientific Applications on Power-Aware Clusters." In Supercomputing Conference
[7] M. Y. Lim, V. W. Freeh, and D. K. Lowenthal. 2006 "Adaptive, Transparent Frequency and Voltage Scaling of Communication Phases in MPI Programs." In ACM/IEEE Supercomputing (SC06),
[8] Muhammad Zakarya, Ayaz Ali Khan, Hameed Hussain, 2010 “Grid High Availability & Service Security Issues with Solutions”, ICIIT, 978-1-4244-813 8-5/10 / $ 26.00 C 2010 IEEE
[9] M. Spuri and G. Buttazzo, 1996 Scheduling Aperiodic Tasks in Dynamic Priority Systems, Journal of Real-Time Systems, 10(2):179-210.
[10] T. D. Burd., T.A. Pering, A. J. Stratakos, and R. W. Rodersen, 2000 Adynamic Voltage Scaled Microprocessor system. IEEE Journal of Solid State Circuits, Vol. 35, No. 11,pp. 1571-1580.
[11] D. Shin and J. Kim, 2006 Dynamic voltage scaling of mixed task sets in priority-driven systems. IEEE Transaction on CAD of Integrated Circuits and Systems 25(3): 438-453.
[12] Nasro Min-Allah, Asad-Raza Kazmi, Ishtiaq Al, Xing Jian-Sheng, Wang Yong-Ji, Minimizing Response Time Implication in DVS Scheduling for Low Power Embedded Systems

[13] M. Spuri and G. Buttazzo, *"Scheduling Aperiodic Tasks in Dynamic Priority Systems,"* The Journal of Real-Time Systems
[14] Wu chun Feng, Michael S. Warren, and Eric Weigle. 2002 The bladed beowulf: A coste _ective alternative to traditional beowulf. In IEEE International Conference on Cluster Computing (CLUSTER 2002), 23-26, Chicago, IL, USA.
[15] Vivek Tiwari, Deo Singh, Suresh Rajgopal, Gaurav Mehta, Rakesh Patel, and Franklin Baez. 1998.Reducing power in high-performance microprocessors. In DAC '98: Proceedings of the 35th annual conference on Design automation, pages 732{737, New York, NY, USA, ACM Press.
[16] K. Nowka, G. Carpenter, and B. Brock. September/November 2003 The design and application of the powerpc 405LP energy-e_cient system on chip. IBM Journal of Research and Development, 47(5/6).
[17] Kanishka Lahiri, Sujit Dey, Debashis Panigrahi, and Anand Raghunathan. Battery-driven system design: A new frontier in low power design. In ASPDAC
[18] E. Jul, H. Levy, N. Hutchinson, A. Black, 1988 'Fine-Grained Mobility in the Emerald System", ACM Trans. Comput. Syst. 6(1): 109-133.
[19] John H. Mathews 1987. Numerical Methods for Computer Science, Engineering, and Mathematics, Prentice-Hall International, Inc.
[20] Dahlquist, Germund, and AkeBjork 1974. Numerical Methods, Prentice-Hall, Inc., Englewood Clifs, N.J.
[21] Aitkinson, Kendall 1978. An Introduction to Numerical Analysis, John Wiley & Sons, Inc., New York.
[22] Phillip Carinhas, PhD, Green Computing Guide, http://fortuitous.com
[23] Samee Ullah Khan, Sherali Zeadally, Pascal Bouvry, Naveen Chilamkurti, 09 June 2011 Green Networks, Springer
[24] Samee Ullah Khan, Pascal Bouvry, Thomas Engel, 12 October 2010 Energy-efficient high-performance parallel and distributed computing, Springer
[25] Giorgio Luigi Valentini, Walter Lassonde, Samee Ullah Khan, Nasro Min-Allah, Sajjad A.Madani, Juan Li, Limin Zhang, Lizhe Wang, Nasir Ghani, Joanna Kolodziej, Hongxiang Li, Albert Y.Zomaya, Cheng-Zhong Xu, Pavan Balaji, Abhinav Vishnu, Fredric Pinel, Johnatan E.Pecero, Dzmitry Kliazovich, Pascal Bouvry, 10 September 2011An overview of energy efficiency techniques in cluster computing systems, Springer
[26] Sherali Zeadally, Samee Ullah Khan, 31 May 2011 Energy-efficient networking: past, present, and future, Springer
[27] Dzmitry Kliazovich, Pascal Bouvry, Yury Audzevich, Samee Ullah Khan, GreenCloud, 2010. A Packet-level Simulator of Energy-aware Cloud Computing Data Centers, IEEE Globecom Proceedings
[28] Samee Ullah Khan, Nasro Min-Allah, 19 April 2011 A goal programming based energy efficient resource allocation in data centers, J Supercomput Springer
[29] NasroMin-Allah, Hameed Hussain, Samee Ullah Khan, Albert Y. Zomaya, 2011 Power efficient rate monotonic scheduling for multi-core systems, J. Parallel Distrib. Comput. ELSEVIER
[30] Ismael Solis Moreno, Jie Xu, Energy-Efficiency in Cloud Computing Environments: Towards Energy Savings without Performance Degradation, University of Leeds, UK
[31] Buyya, R., Beloglazov, A., & Abawajy, J. 2010, July 12-15. Energy-Efficient Management of Data Center Resources for Cloud Computing: A Vision, Architectural Elements, and Open Challenges. Paper presented at the Proc. of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications, Las Vegas, NV, USA.
[32] Liu, L., Wang, H., Liu, X., Jin, X., He, W. B., Wang, Q. B., et al. 2009. GreenCloud: a new architecture for green data center. Paper presented at the Proc. of the sixth International Conference on Autonomic Computing Barcelona, Spain.