Energy Management for Cloud Computing: A survey from Scheduling Perspective of Heuristic, Game Theory and Learning Strategy

Norhashimi Bin Mohd Nor*, Masnida Binti Hussin, Hasan Bin Selamat

Faculty of Computer Science & Information Technology
University Putra Malaysia, Selangor, Malaysia
*Corresponding author email: norhashimimohdnor@yahoo.com

Abstract: Cloud computing has been a champion in Information Technology (IT) world for supporting flexibility on-demand services. It is leveraged to instantly fulfill dynamic demands from heterogeneous environment. However, the massive demand for the Cloud computing services contributed to high energy consumption, thus affecting cost of service and processing components lifetime. Quite a portion of the energy consumed is dissipated for ensuring processing components (i.e., processor) running but idle. An effective task scheduling is the solution for better processing performance while reducing energy consumption. In this paper we compared several energy efficient scheduling from three (3) different perspectives (i.e., heuristic, economic and intelligent points of view). These different perspectives are chosen particularly for exemplifying essential principles and features of energy management in task scheduling. Novel knowledge on the energy management are identified through similarities, differences and energy models. We believed that better trade-off between energy consumption and system performance is the solution to green Cloud computing.

Keywords: energy efficient, resource management, scheduling, performance, heuristic, learning, game theory.

1. Introduction

Evolution of computing started when Grid computing hit the world in the beginning of 90's [1]. The evolution continued with commercialization of Cloud computing in the early 21st century with Amazon Web Services (2002), Google Docs (2006) and Azure by Microsoft (2009) [2]. Cloud computing is a state-of-the-art technology where companies or organizations are able to have processing and storage services without having to physically own them. It has changed the way of computing massively from isolated sequential computing to high performance parallel distributed computing. This practical technology eliminates the hassles of procuring, managing and maintaining data centers while saving the maintenance and operation costs. Furthermore, the Cloud provides high reliability, high scalability, high security and location independency services to its users [2].

According to Wang [3] the highest energy cost of data centers is used to maintain servers (Fig. 1). It is basically to fulfill service level agreement (SLA) between the Cloud providers and users. In order to sustain business demands, more hardware are added up to facilitate the processing requirements. The issue of expanding and upgrading hardware to meet the SLA needs leads to excessive and ineffective energy consumption.

The massive and rapid growth of the Cloud computing in order to fulfill the world's needs ironically impacts the world in a negative way. Large computing infrastructure needs more energy thus more fuel is needed to be burnt to generate sufficient electricity. As a result, more greenhouse gas (GHG) is released in the atmosphere that leads to global warming, acid rain and smog.

The energy consumption issues have inspired many researchers to focus on green Cloud computing [4-6]. Among the basic aspects in the green Cloud computing are power management, cooling, recycling, electronic waste disposal and virtualization. Based on Rizvandi and Zomaya [7] the virtualization technology is the core enabler for green Cloud computing. It is a method of enhancing the energy efficiency of servers in the Cloud by sharing limited resources with multiple workloads [6]. Through virtualization, the number of physical machines (PMs) is reduced and the task execution is performed by virtual machines (VMs). The VMs are so flexible that they act as independent servers/nodes thus maximize resources utilization.

Abundance of work had been done to manage energy consumption in green Cloud. Dynamic voltage and frequency scaling (DVFS) are among the techniques used to adjust voltage and frequency of processors that are based on current workload. A special hardware that is compatible with DVFS is needed to control the energy consumption while sustaining business's SLA. Energy model, on the other hand, is a holistic method where it captures data centers at large. It covers not only the components related to computation but also the components for cooling and other common equipment in the data centers that consume electricity.

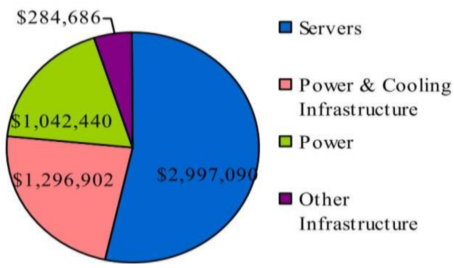

Figure 1. Energy cost of data center

In particular, the effectiveness of VM, DVFS and energy model is utilized for energy efficiency through resource management or more specifically scheduling. Under the energy management in Cloud, the issue that needs to be concerned is minimizing the energy consumption of components in data centers especially processors. Based on Aziz and El-Rewini [8], most processors on average consume approximately 32Watt even when they are in idle mode compared to storages that merely use 6Watt. In peak processing state the energy consumption of a processor is boosted up to more than 80Watt to 95Watt [9]. Task scheduling approach promotes for better energy consumption by scheduling jobs into the processor in an effective way. The task scheduling approach contributes to the green Cloud by providing the adaptive scheduling decision in order to handle dynamicity and heterogeneity of resources. In this paper we focus on energy management in Cloud by comparing scheduling approaches from the three (3) different perspectives; heuristic, economy and intelligent.

2. Energy efficient scheduling

There are many energy efficient scheduling approaches [8-24] that have been studied from different scheduling perspectives. Energy efficient task scheduling is a scheduling algorithm that dynamically allocates jobs into processor to achieve better performance and to minimize energy consumption. The system performance and energy consumption should be measured throughout the task execution either during peak or idle state, hence accumulative energy consumption [25] is attained and inclusive conditions are catered. The main issue in this area of research is how to balance between the performance of the task scheduling while minimizing energy consumption [16]. It is a challenge to find the best trade-off (balance/equilibrium) between the best output (i.e., performance) and energy consumed. In this paper we have compared three (3) different scheduling approaches for energy efficiency, and have studied their strength, effectiveness and potential use in the future. We have investigated the strategy of heuristic, game theory and learning to determine the features and parameters that are used to evaluate energy consumption and system performance.

2.1 Heuristic Strategy

Heuristic is one of the most popular scheduling approaches to be chosen for focusing on green Cloud. The strategy performs effectively for scheduling jobs into processors in uncertain, dynamic and large-scale environments [11,16] [23-25]. The heuristic strategy is initiated by designing a small companion problem and is extended to complex issues to solve big problems. An effective guess can be made through manipulating a lot of rules and knowledge in this strategy. With the prediction technique, researchers are able to organize the overall structure of problem solving while identifying which parts to focus on. This strategy is really helpful if it involves the dynamic environment of the Cloud.

2.2 Game Theory Strategy

Game theory provides a strategy on dealing with conflict situation that involves bargaining and persuading. For example, the rule of bargaining helps in reducing energy consumption in the system. It is because the game theory offers efficient mathematical tools in providing multiple level of optimality. With the mathematical tools, it is suitable to deal with heterogeneous environment where the Cloud has to optimize multiple goals from multiple perspectives [17]. Basically, assumptions made in game theory about payoffs are known in advance. However, it is hard to apply it in dynamic environment like Cloud.

2.3 Learning Strategy

The gist of learning strategy is to provide agents or task schedulers with intelligent features to enable them to learn and improve their actions. At a glance, the approach is very promising since it has the capability to grow and evolve by learning capacity. Task scheduling that adopted this strategy basically stores the scheduling actions. The scheduler, then reviews and categorizes the actions either they are good or bad. The next action chosen by the scheduler is based on information stored in the learning memory. This learning process continues until it reaches a certain goal. One of the biggest challenges of adopting intelligent feature for energy efficient scheduling is that it might increase the computational complexity due to rapid changing environment of Cloud computing.

3. Heuristic strategy for Energy Efficient Scheduling

There are many energy efficient schedulings that adopted heuristic strategy (i.e., Power-Efficient Distributed Scheduling [10], PASTA [11], Power Aware PRISM [8], Priority-Based Scheduling [16], EADAG [19], Bi-Objective Hybrid Genetic [20], EETS [22] and 2 Phase Heuristic [23]). By applying heuristic strategy, certain level of knowledge and experience need to be effectively guessed and achieved. Power-Efficient Distributed Scheduling by Sharifi et al. [10] proposed strategy of assigning a set of virtual machines (VMs) to physical machines (PMs) in data centers. Based on the authors, the 'blind consolidation' of VMs is one of the causes of energy wastage. The state of system in their work is calculated based on VMs and PMs Disk utilization. The consolidation fitness then determined the right VMs that need to be redistributed to other PMs, while the current PMs are switched off. The work proved that the scheduling approach is 24.9% better in energy efficiency compared to static assignment. By dynamically assigning VMs to PMs, scheduling performance improved, however the challenge is to determine the best VM due to unpredictable changes in the system workload.

Power-Aware Solution to Scheduling of Precedence-constrained Tasks (PASTA) proposed by Sharifi et al. [11] is based on Dynamic Voltage and Frequency Scaling (DVFS). Their versatile DVFS approach is able to utilize all processors even though not every processor in the distributed systems has the DVFS capability. In two phases, PASTA selects a subset of computing resources (CRs) from a pool of available CRs to balance between performance and energy consumption. Then, it uses low-complexity energy aware algorithm to create a schedule that executed jobs on the selected CRs. Based on their experiments, PASTA achieved 60% energy efficiency than Heterogeneous Earliest Finish Time (HEFT). However, PASTA's makespan is longer compared to HEFT's. It is because there is higher duration of time that an application spends in the selected CRs in PASTA and it affects the result.

Power Aware PRISM by Aziz and El-Rewini [8] is based on Priority Scheduling in Management Architecture for Resource Services (PRISM) framework. They adopted various heuristic for energy-aware scheduling algorithms that employed multi-objective function for diverse efficiency-performance trade-offs. Their scheduling algorithms consist of three steps (i.e., job clustering, re-evaluating to give better scheduling alternatives and selecting the best schedule). Power Aware PRISM provides 25% to 94% better energy consumption where 25% energy efficiency is based on performance heuristic while 94% assessing is based on minimum cost heuristic. However, the experiments have been conducted in a homogeneous environment where the time between machines transition is fixed.

Priority-Based Scheduling by Hussin et al. [16] proposed task scheduling that caters different priorities (deadlines) while exploiting the heterogeneous resources. Their scheduling approach applied hierarchical scheduler by utilizing two levels of queues (i.e., global and local schedulers). The queue in the global scheduler follows first come first serve basis. The jobs are then categorized based on their deadlines; high, medium or low. Priority-Based scheduling is proven to achieve 30% to 50% better energy efficiency compared to random selection scheduling. This research successfully balanced the energy efficiency and processing performance as it applied priority queue. By categorizing the jobs in the local (priority) queue it contributes for better makespan. The queue waiting time is also reduced, hence it becomes possible to improve the resource utilization and to minimize energy consumption.

Energy Aware Directed Acyclic Graph (EADAG) by Baskiyar et al. [19] proposed combination of Decisive Path Scheduling (DPS) and Dynamic Voltage Scaling (DVS) techniques for heterogeneous processors. The scheduling approach calculated the energy consumption rate based on processors idle time (i.e., the higher unutilized processors the more energy efficient). Particularly, the processors idle time increase when there are high numbers of processing nodes or communication to computation ratios. EADAG employed the DPS on the Directed Acyclic Graph for better makespan while applying the DVS for the slack time to achieve energy efficiency. EADAG is proven to be 40% more energy efficient than DSP with better makespan. The work, however, does not consider types of tasks (e.g., independent and interdependent).

Bi-Objective Hybrid Genetic by Mezmaz et al. [20] used hybrid scheduling approach by combining genetic algorithm and energy-conscious scheduling heuristic (ECS). It inherits the strength of exploration in genetic algorithm (GA) and the strong heuristic capability in ECS. Their scheduling approach basically created the task parts of each gene, then formed the processor parts of these genes. Finally, it calculated the fitness of a solution in terms of energy consumption and makespan. The Bi-Objective Hybrid Genetic scheduling was able to achieve 47.49% energy efficiency. The high energy efficiency achieved in this scheduling strategy is related to the hybrid approach of GA and heuristic technique. However, based on the experiment results, the energy efficiency decreased as the number of tasks increased. This gap needs to be explored in order to tackle larger and more complex tasks in the real environment.

Energy Efficient Task Scheduling (EETS) by Yao et al. [22] adopted the offloading process on Cloud and non-offloading process for mobile devices. The scheduling algorithm decides whether to change offload or non-offload status that was based on data size, data source, data storage, data compression and bandwidth speed. EETS calculated energy consumption from both offloading and non-offloading status, and then selected the lower energy consumption from the workload status for task execution. EETS was able to effectively reach 99% better performance compared to other scheduling algorithms in the experiments. The energy consumption of EETS depends on bandwidth, however, it is hard to determine its current state due to rapid bandwidth fluctuation. There is an opportunity to solve the issue by using compression technique. As high compression is able to reduce data size, thus it saves energy consumption in communication.

2Phase Heuristic (2PH) by Pinel et al. [23] based on Min-Min heuristic adopted Parallel Asynchronous Cellular Genetic Algorithm (PA-CGA) in the scheduling approach. 2PH basically was executed on sequences of Min-Min and the local search operator called Highest To Lower Loaded (H2LL). The scheduling approach highlights the incorporation of low-energy computing nodes in heterogeneous distributed computing systems and its ability to improve energy efficiency. The approach was able to prevent overloading or overheating that was capable to impact the nodes lifetime. 2PH provides better energy consumption compared to Min-Min Heuristic. The scheduling was designed for independent tasks, therefore there is an opportunity to enhance the scheduling approach for dependence-constraint tasks as future work.

Some of the researchers [8, 16, 23-24] proposed an energy efficient scheduling and managed to reach an appropriate balance between the system performance and the energy consumption. However, the dynamicity and heterogeneity on their computing environment is limited to some extent. Table 1 summarizes the metrics and system behaviors from the above heuristic scheduling approaches to design the energy model.

There are various energy models adopted by heuristic strategy for energy efficiency. Most of them take the resource utilization and idle time of resources in calculating energy consumption into account. For scheduling approaches that adopted the DVFS technique they have additional parameters such as voltage and frequency in their energy model.

Table 1. Energy model for energy efficient task scheduling with heuristic approach

Scheduling Name	Energy Model
Power-efficient distributed [10]	Power consumption = Summation of power consumed by processor and disk. Energy consumption = Integral of the power consumption over a period of time.
PASTA [11]	Power consumption of task schedule* = Power consumption of each CR** from the beginning of schedule to its finish time. *Schedule = Queue of application tasks. **CRs= Computing resources.
PRISM [8]	Resource utilization ratio = Ratio between the average active workstations and the total number of workstations. Energy saved = 1− Resource utilization ratio.
Priority [16]	Resource utilization = Power usage of processor at 100% utilization multiplied by total task execution time. Resource idle = Power usage of idle processor multiplied by total idle time. Energy consumption = The summation of resource utilization and idle.
EADAG [19]	Total energy consumption (without voltage scaling) = Total power consumed by processors multiplied by makespan. Total energy consumption (with voltage scaling) = Summation of non-idle power consumption* and idle power consumption** *non-idle power consumption refers to average power consumed by processors at normal voltage multiplied by non-idle time **idle power consumption refers to average power consumed by processors at lower voltage multiplied by idle time
Bi-Objective Hybrid Genetic [20]	Energy Consumption = Processor voltage multiplied by total makespan.
EETS [22]	Total Energy Consumption (offloading Mode) = Power for sending & receiving data multiplied by byte of data over network bandwidth. Total Energy Consumption (non-offloading Mode) = Summation of processing* and communication energy** *processing energy refers to power for processing instructions multiplied by byte of compressed data over network bandwidth. **communication energy refers to power for sending & receiving data multiplied by byte of compressed data over network bandwidth.
2 Phase Heuristic [23]	Energy consumption = Power of machine multiplied by completion time of machine.

The DVFS technique requires altering voltage and frequency in an attempt to reduce the energy consumption. The energy model for offloading and non-offloading scheduling on the other hand is different in the sense that they include the energy consumption for network bandwidth, data sending and receiving. It is due to the fact that processing in offloading and non-offloading scheduling involves data transmission.

4. Game Theory strategy for Energy Efficient Scheduling

Equilibrium is a key concept in game theory and it aims to provide optimal solutions against several choices [14]. Nash equilibrium is a non-cooperative strategy where each player has its own game with its own way without bothering the other players. It shows the true colors of selfish behavior in reality, but remarkably effective. There are also cooperative game theories such as Nash axiomatic which finds bargain point and settle down [18].

Pre-scheduling and Scheduling using Game Theory (PSGT) by Abdeyazdan et al. [14] proposed energy efficient task scheduling with initial pre-scheduling process. PSGT aims for the best trade-off between makespan and energy consumption by discovering the equilibrium number of processors. Each level of task graph is considered as a selfish player with non-cooperative behavior by using Nash equilibrium concept that tried to get the best number of processors for themselves. PSGT determined the appropriate number of processors and then, merged the tasks for assigning to processors. PSGT is able to reach better energy consumption with less number of available parallel processors. The pre-scheduling stage as implemented by the authors is a good technique in massaging the tasks. It is because the system is 'more prepared' for scheduling the tasks thus it improves the performance. Instead of the non-cooperative or Nash equilibrium, it could be interesting to alter this scheduling with the cooperative or Nash bargaining to see the result.

Cooperative and Non-cooperative Task Scheduling by Bielik and Ahmad [17] that was based on Energy Aware Task Allocation (EATA) are converted into bargaining game for distributed heterogeneous grid. Results from both scheduling approaches (cooperative and non-cooperative) are used by Dynamic Voltage Frequency Scaling (DVFS) in the processor to reduce the energy consumption. Their work proved that cooperative scheduling saves 8% to 26% more energy compared to non-cooperative scheduling. In other words, non-cooperative scheduling is 3 to 4 times faster than cooperative scheduling. Low performance of the cooperative scheduling is caused by the complex pseudo-code. Incorporation of cooperative and non-cooperative scheduling would create a hybrid game theory approach for better performance.

Meanwhile, Nash Bargaining Solution – Energy Aware Task Allocation (NBS-EATA) by Ahmad et al. [18] identified the bargaining point by combining the classical game theory techniques that applied the Kuhn-Tucker conditions and the Lagrangian. NBS-EATA performed better trade-off between energy consumption and performance compared to Greedy Heuristic. The authors mentioned that they selected the cooperative approach in this scheduling because their intention was to optimize the cumulative performance rather than individual cores.

Generalized Tit-For-Tat Temperature-aware Scheduling (GTFTTS) by Wu et al. [24] used generalized Tit-For-Tat game theory for resource-rich environment. The scheduling decision for cooperation depends on the hardness factor (i.e., percentage of cooperation among other processors in prior round). While lowering the energy consumption in the whole system, the scheduling approach managed to avoid local hotspot in the processor thus increases the processors lifetime. GTFTTS is slightly better than Pure Tit for Tat scheduling and NBS-EATA in term of energy efficiency. It is achieved by occasionally forcing the neighboring cores to cooperate when some cores are overloaded.

Table 2. Energy model for energy efficient task scheduling with game theory approach

Scheduling Name	**Energy Model**
PSGT [14]	Appropriate number of processor = Applying Nash equilibrium at each level of task graph.
Cooperative and Non-cooperative Task Scheduling [17]	Power consumption = Multiplication of voltage, frequency and effective capacitance.
NBS-EATA [18]	Energy consumption = Multiplication of capacitance, voltage and task cycles.
GTFTTS [24]	Power density of core = Multiplication of power and tasks cycles divided with area of core.

The game theory approach is suitable for task scheduling in Cloud where there are limited knowledge of system states and resources. Although large payoff matrix in the game theory approach would create complex challenges but this approach is able to deal with energy consumption and system performance in dynamic computing environment. Table 2 summarizes the energy models with metrics and system behaviors that used to calculate energy efficiency based on the game theory scheduling discussed above.

5. Learning Based strategy for Energy Efficient Task Scheduling

There are several types of learning approaches that are applied in task scheduling such as reinforcement-learning (RL), online learning, self-learning and meta-learning and many more [9, 13, 26-28]. The RL scheme is a goal directed learning approach where it enhances its actions through its experience with trial and error [9]. Adaptive Reinforcement Learning Based Scheduling (ARLS) by Hussin et al. [9] proposed a dynamic scheduling that observes and adapts to various tasks and heterogeneous resources through feedback signal. ARLS provides the scheduling capability by developing associations between the scheduling decisions and the current state of the environment. The scheduling algorithm calculates the feedback signal i.e., reward and penalty in order to assign the task to the right resource. The authors incorporated task grouping technique by categorizing the various priorities tasks into related task group. The ARLS achieved better trade-off between energy consumption and performance in their experiments. The scheduling approach has the potential to be further enhanced with consideration of dependent-task in the scheduling process.

Memory-Based Scheduling by Niemi and Hameri [13] utilized fuzzy-logic memory based scheduling to dynamically allocate the memory threshold that was based on the overall load in the distributed systems. The fuzzy control is formulated using free memory, with IF x AND y THEN z rule. Then, the result is calculated using a centroid i.e. the canter of mass method to get the appropriate memory threshold. This scheduling adopted the multi-levels scheduling (i.e., cluster level and node level schedulers). The authors also highlighted that computing and communication intensive applications are required to run in parallel in order to improve performance and energy efficiency. Through their experiments, the scheduling strategy improves throughput and reduces energy consumption compared to one job per CPU core scheduling.

Online Energy Saving Resource Optimization (OESRO) by Yuan et al. [15] proposed comprehensive resource management method for virtual machines (VMs) to improve energy efficiency and performance. Their energy saving mapping algorithm incorporated reinforcement learning into threshold-based VMs migration to dynamically find the lowest energy consumption between VMs and hosts while avoiding resources contention. In the experiments, they used 0.8 as the threshold value, which means, the next VM needs a new suboptimal host if the current host CPU utilization exceeds 80%. It is proven that the scheduling approach achieved at least 30.3% energy efficiency compared to traditional random placement. The author also has adopted the Sarsa algorithm (i.e.; state-action-reward-state-action) which is a kind of online Q-learning algorithm. Sarsa is better in a way that it uses the real-time Q value rather than the highest Q value, to achieve time-precision decisions. The algorithm is resistant to rapid changing workloads (i.e.; number of tasks and requirements). The work, however, does not consider on how the algorithm handle high computational complexity in the reinforcement learning strategy.

The learning based scheduling approach is capable to enhance and improve the scheduling decisions from one cycle to another. It is a very powerful feature for task scheduling in heterogeneous distributed systems especially to survive in the long run. However, the learning approach requires a pre-processing time in order to reach the optimal pace in the learning process [9]. Task scheduling that adopted this approach has to formulate better analyzing strategy during the learning process, hence it reduces overhead in computation. Table 3 gives the calculation of the energy model used by the learning based scheduling approach discussed above for energy efficiency.

The learning process aims to gain many experiences, therefore the scheduler will be able to justify the best scheduling decision. Due to huge integration process between memory and current state in the systems, the learning-based scheduling might leads to high computational complexity. There are a lot of specifications of learning process that still can be explored for energy efficiency.

Table 3. Energy model for energy efficient task scheduling with learning approach

Scheduling Name	Energy Model
Adaptive Reinforcement Learning Based [9]	Resource utilization = Power usage of processor at 100% utilization multiplied by total task execution time Resource idle = power usage of idle processor multiplied by total idle time Power consumption = The summation of resource utilization and idle
Memory-based [13]	Energy consumption = Multiplication of capacitance, voltage and task cycles.
Online Energy Saving Resource Optimization [15]	Energy level = Summation of resource utilization difference between hosts and optimal level, and disk utilization difference between hosts and optimal level

6. Differences and Similarities

There are different strategies used in task scheduling towards energy efficiency. There are many factors that contribute to the differences such as technique used, assumptions and research limitations. There are also some similarities in the scheduling approaches since they operate in the same distributed computing platform. We have highlighted the differences and similarities of scheduling approaches towards energy efficiency for Cloud computing that were based on two main features; scheduling rule and energy model as in Table 4 and 5.

Table 4. Differences of approaches

Strategy	Scheduling Rule	Energy Model
Heuristic	1. Heuristic approach used for pre-scheduling processing. 2. The predictable rule satisfies energy saving equals to less processor and processing time.	Consider busy and idle time.
Game theory	1. Cooperative game theory strategy aims for energy saving, while the non-cooperative game theory considering better scheduling performance. 2. Equilibrium is similar to pareto-optimal hence balance energy consumption and performance.	Consider capacitance as a component in the energy model.
Learning process	1. Using information from previous actions and current environment status. 2. The learning experience improves scheduling actions.	Consider memory utilization.

Table 5. Similarities of approaches

Strategy	Scheduling Rule
Heuristic, game theory and learning process	1. The main objective of task scheduling for energy efficient is to balance between energy consumption and performance. 2. Correct threshold for processor, memory, disk and communication link is vital for performance and energy efficiency. For example, too low threshold could lead to underutilization of resources while creating bottleneck when the threshold is too high. 3. It is expected to maximize resource utilization for energy efficiency and enhancing system performance. 4. Virtualization leads to energy efficiency by migrating or re-mapping a set of physical machines (PMs) to set of virtual machines (VMs) . 5. Lower power consumption does not only save energy also enhance hardware lifetime. 6. All scheduling approaches are capable of handling dependent and independent tasks. 7. Heat generation reduction is as important as energy consumption reduction because of cooling and lifetime factors. 8. Standard benchmark for measuring energy efficiency degree of task scheduling should be established. 9. All task scheduling approaches should be tested on heterogeneous tasks and resources.

7. Conclusion

In this work we analyzed three (3) different perspectives of scheduling algorithms for energy efficiency. The heuristic strategy is the most common method used for scheduling tasks while managing the energy consumed. However, researchers need to have fair knowledge on the environment, task characteristics and behaviors of resources in the distributed systems. Game theory, on the other hand, is suitable for exploring unknown situation or environment and it is able to give significant outcome. The intelligent-based scheduling through learning capability is another promising strategy for energy efficient distributed systems. The learning process is able to repair and improve the performance when the previous actions exhibit deficient results. Finding optimal protocols for energy efficient task scheduling is still an open task. Hence, there is a lot of potential for more researches. Optimistically, we hope our comprehensive comparative study will help other studies for better overview of energy management structure in order to develop the (near) optimal energy efficient scheduling.

References

[1] M. Naqaash, M. A. Iqbal, Q. Pervaiz, O. Shamim, and S. A. Hussain, "Grid Computing Used For Next Generation High Speed Processing Technology", vol. 02, no. 05, pp. 1926–1933, 2010.

[2] M. Torkashvan, H. Haghighi, "A Service Oriented Framework for Cloud Computing", *Proceedings of the 3rd International Conference on Information and Communication Systems*, no. 25, 2012.

[3] X. Wang, Y. Wang, "Energy-efficient Multi-task Scheduling Based on MapReduce for Cloud Computing", *2011 Seventh Int. Conf. Comput. Intell. Secur.*, pp. 57–62, 2011.

[4] J. Baliga, R. W. A. Ayre, K. Hinton, R.S. Tucker, "Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport", *Proc. IEEE*, vol. 99, no. 1, pp. 149–167, 2011.

[5] S. Y. Jing, S. Ali, K. She, Y. Zhong, "State-of-the-art research study for green cloud computing", *J. Supercomput.*, vol. 65, no. 1, pp. 445–468, 2011.

[6] A. Uchechukwu, K. Li, Y. Shen, "Improving cloud computing energy efficiency", *2012 IEEE Asia Pacific Cloud Comput. Congr.*, pp. 53–58, 2012.

[7] N. B. Rizvandi, A.Y. Zomaya, "A Primarily Survey on Energy Efficiency in Cloud and Distributed Computing Systems", *eprint arXiv:1210.4690*, 2012.

[8] A. Aziz, H. El-Rewini, "Power efficient scheduling heuristics for energy conservation in computational grids", *J. Supercomput.*, vol. 57, no. 1, pp. 65–80, 2011.

[9] M. Hussin, Y. C. Lee, A. Y. Zomaya, "Efficient Energy Management Using Adaptive Reinforcement Learning-Based Scheduling in Large-Scale Distributed Systems", *2011 Int. Conf. Parallel Process.*, pp. 385–393, 2011.

[10] M. Sharifi, H. Salimi, M. Najafzadeh, "Power-efficient distributed scheduling of virtual machines using workload-aware consolidation techniques", *J. Supercomput.*, vol. 61, no. 1, pp. 46–66, 2011.

[11] M. Sharifi, S. Shahrivari, H. Salimi, "PASTA: a power-aware solution to scheduling of precedence-constrained tasks on heterogeneous computing resources", *Computing*, vol. 95, no. 1, pp. 67–88, 2012.

[12] T. Zhu, C. Shu, H. Yu, "Green Scheduling: A Scheduling Policy for Improving the Energy Efficiency of Fair Scheduler", *2011 12th Int. Conf. Parallel Distrib. Comput. Appl. Technol.*, pp. 319–326, 2011.

[13] T. Niemi, A. P Hameri, "Memory-based scheduling of scientific computing clusters", *J. Supercomput.*, vol. 61, no. 3, pp. 520–544, 2011.

[14] M. Abdeyazdan, S. Parsa, A. M. Rahmani, "Task graph pre-scheduling, using Nash equilibrium in game theory", *J. Supercomput.*, vol. 64, no. 1, pp. 177–203, 2013.

[15] J. Yuan, X. Miao, L. Li, X. Jiang, "An Online Energy Saving Resource Optimization Methodology for Data Center", *J. Softw.*, vol. 8, no. 8, pp. 1875–1880, 2013.

[16] M. Hussin, Y.C. Lee, A.Y. Zomaya, "Priority-Based Scheduling for Large-Scale Distribute Systems with Energy Awareness", *2011 IEEE Ninth Int. Conf.*

Dependable, Auton. Secur. Comput., pp. 503–509, 2011.

[17] N. Bielik, I. Ahmad, "Cooperative versus non-cooperative game theoretical techniques for Energy Aware Task scheduling", *International Green Computing Conference (IGCC)*, 2012.

[18] I. Ahmad, "Using game theory for scheduling tasks on multi-core processors for simultaneous optimization of performance and energy", *2008 IEEE Int. Symp. Parallel Distrib. Process.*, pp. 1–6, 2008.

[19] S. Baskiyar, R. Abdel-Kader, "Energy aware DAG scheduling on heterogeneous systems", *Cluster Comput.*, vol. 13, no. 4, pp. 373–383, 2010.

[20] M. Mezmaz, Y. C. Lee, N. Melab, E. G. Talbi, A. Y. Zomaya, "A bi-objective hybrid genetic algorithm to minimize energy consumption and makespan for precedence-constrained applications using dynamic voltage scaling", *IEEE Congr. Evol. Comput.*, pp. 1–8, 2010.

[21] J. Kołodziej, S. U. Khan, L. Wang, A. Byrski, N. Min-Allah, S. A. Madani, "Hierarchical genetic-based grid scheduling with energy optimization", *Cluster Comput.*, vol. 16, no. 3, pp. 591–609, 2012.

[22] D. Yao, C. Yu, H. Jin, J. Zhou, "LNCS 8147 - Energy Efficient Task Scheduling in Mobile Cloud Computing", *International Federation for Information Processing*, pp. 344–355, 2013.

[23] F. Pinel, B. Dorronsoro, J. E. Pecero, P. Bouvry, S. U. Khan, "A two-phase heuristic for the energy-efficient scheduling of independent tasks on computational grids", *Cluster Comput.*, vol. 16, no. 3, pp. 421–433, 2012.

[24] G. Wu, Z. Xu, Q. Xia, J. Ren, "GTFTTS: A Generalized Tit-for-Tat Based Corporative Game for Temperature-Aware Task Scheduling in Multi-core Systems", *2010 3rd Int. Symp. Parallel Archit. Algorithms Program.*, pp. 81–88, 2010.

[25] S. Wang, J. J. Chen, Z. Shi, L. Thiele, "Energy-Efficient Speed Scheduling for Real-Time Tasks under Thermal Constraints", *2009 15th IEEE Int. Conf. Embed. Real-Time Comp. Sys. App.*, pp. 201–209, 2009.

[26] E. Yom-tov, Y. Aridor, "A Self-optimized Job Scheduler for Heterogeneous Server Clusters", *Springer-Verlag Berlin Heidelberg*, pp. 169–187, 2008.

[27] R. Priya, B. F. de Souza, A. L. D. Rossi, A. C. P. L. F. de Carvalho, "Predicting execution time of machine learning tasks using metalearning", *2011 World Congr. Inf. Commun. Technol.*, no. 1, pp. 1193–1198, 2011.

[28] V. Rajan, L. E. Celis, "CrowdControl: An online learning approach for optimal task scheduling in a dynamic crowd platform", *ICML Workshop: Machine Learning Meets Crowdsourcing*, 2013.