International Journal of Technical Innovation in Modern Engineering & Science (IJTIMES)
Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585
Volume 5, Issue 04, April-2019

ENERGY EFFICIENT TASK SCHEDULING USING DVFS SCHEME

Akanksha, Nirmal kaur, Ravreet kaur

UIET, Panjab University Chandigarh, India

Abstract- For redundant energy consumption, power optimization appears as a key performance parameter for the HPC (High Performance Computing) platform. Scheduling plays a vital role in attaining minimum make-span. The main focus of this paper is to minimize the energy consumption of application using DVFS. In this research, an algorithm named as is proposed for efficient energy to schedule DAG on heterogeneous graph G(V,E). The system performance is enhanced by reducing the energy consumption by considering computation energy of processors. DVFS with HEFT task scheduling algorithm is proposed. The proposed algorithm copes effectively and reduces energy consumption of applications. The work has generated heterogeneous graphs from homogenous graph models. From the experiments, it is observed that when DVFS is used, the energy consumption is reduced by 11.77%.

Keywords: Task scheduling, Heterogeneous Earliest Finish Time, Dynamic voltage and frequency scaling.

I. INTRODUCTION

Today, high end computing services require large amount of energy to perform task, though it provides high-tech computing solutions for science as well as business applications [1]. For an instance, to operate medium volume data, it requires 80, MW power. Therefore, it is understood that the computing sources utilized approximately 0.5 percent of the overall available power [2]. Energy utilization for high performance services make available to an important electric bill. If the processors consume high energy then the components get heated and hence require high cooling cost. Also, if power consumption is high while operating the resources for a long period this will lead to affects the system's reliability as well as availability. Thus, to lessen the consumption of electricity for high end computing becomes an important area of research [3]. However, modern processor performance is elevated daily; therefore, the power consumed by high-performance computer system becomes an essential and immediate problem. At present, performance along with energy efficiency is the two main measure of modern processor. Designing efficient energy and making clusters that are environment friendly is extremely desirable. Energy awareness management is derived as a strategy approach for Green computing [4], while reducing operating costs and thus raising system reliability. DVFS is the dynamic voltage and frequency scaling technique, which allows processors to save energy by operating at multiple frequencies under diverse supply voltages to perform tasks during slack time without interrupting overall completion time as a whole [5].
Modern processors utilized the DVFS techniques, which facilitate processors to use several frequencies under diverse supply voltages. The DVFS method therefore provides a widespread opportunity to minimize the consumption of energy in high performance computing system through 'scaling supply- voltages' of processor [6].

Our research focuses on the development of algorithm that schedules the task in such a way to reduce the energy consumption of parallel task execution by utilizing the DVFS method in slack time. Activities scheduled for a time-consuming processor are subject to dynamic voltage and frequency scaling while saving the total end time of the activity graph. Often, earlier DVFS energy savings were acquired at the expense of the increased make span period. The designed algorithm saves the energy of the processor by choosing the best combination of voltage & frequency levels without influencing the makespan and also without contravene the precedence relationship between different activities [7].

II. LITERATURE SURVEY

Enhancing the energy efficiency of better performed clusters becomes a significant area of research. Therefore to minimize the energy consumption rate Liang et al. [8, 2015] proposed a new algorithm named as NEASA (energy aware scheduling algorithm). This algorithm has been decrease the energy consumption rate without affecting the SLR y creating difference between static and dynamic slack time. It has been determined that the energy consumption rate has been reduced by 10.8%. To resolve the difficulty of scheduling length and 'energy consumption' while creating clusters has been solved by Ruan et a in. [9, 2007]. The authors have designed a TDVAS (energy efficient scheduling algorithm) in combination with DVFS technique that helps to save energy in an efficiency manner. This algorithm utilized the idle times to decrease the voltage of processor and hence decrease the energy consumption which has been examined by the parallel application operating on clusters. Manzak et al. [10, 2003] presented a dynamic voltage activity scheduling algorithm which reduces the energy when the arrival time of task, deadline- time, execution -time etc have been known. The authors have considered the earliest time and periodic scheduling algorithm. The relationship between the task voltage has been measured mathematically using "Legrage Multiplier" algorithm and then iterative algorithm has been designed which satisfies the relation among them. The problem of power consumed by server while executing the task has been solved by Lee et al. in[11, 2009] by using the concept of task scheduling in DCS (distributed computing system)

IJTIMES-2019@All rights reserved

which reduce the energy consumption. Author utilized DVFS technique to reduce energy consumption which enables various processors to work with various supply voltage. But it has been observed that there is a compromise between the schedule -quality and the ‘energy consumption ‘rate. Mei et al. [12, 2012] presented an “Energy aware scheduling by minimizing duplication that considered both time as well as energy consumption of application. The duplicate tasks have been discarded by using algorithm which is based on the “duplication algorithm”. The experiment has been performed on arbitrary generated DAG and concluded that energy consumption rate p to 15.59% has been saved as compared to the exiting duplication algorithm. Nath et al. [13, 2015] proposed a DVFS algorithm in combination with HPC (high performance coupling) clustering approach that is utilized in heterogeneous computing system. The proposed scheduling algorithm helps to adjust voltage as per the system load. Xie et al. [14, 2017] proposed a local energy efficient scheduling algorithm to resolve the issues of energy consumption of parallel application executed in heterogeneous distributed system. The proposed technique saved approximately 36 to 55 percent of energy. Arabnejad et .al [15, 2016] has presented scheduling algorithm is presented with a time quadratic complexity that takes into account two important aspects of metrics that are based upon the workflow namely time and cost, called DBCS. DBCS achieves successful table rates similar to the high-time intricacy algorithms for both randomized workloads and physical applications on a variety of platforms as well as a set of deadline and budget constraint values. Wang et .al [16, 2013] develops the extraction schedule to reduce power consumption for task execution. The relationship among power consumption as well as task execution time has been discussed. Scheduling algorithms in the groups that enable DVFS to perform tasks in parallel manner are: “PATC and PALS”. Algorithms presented in recession are looking for functions with no increment in the length of the scheduling. Ma et .al [17, 2009]has proposed algorithm is more realistic for controlling dependency graphs and online attitudes. Through a large number of experiments, the algorithm has proven to be nearly ideal energy efficiency, the average artificial workload is better than 16.4%, and the actual workload is better than 12.9% based on the EAD algorithm. The algorithm of online extension is better as contrasted to EDF, which has an average energy savings of up to 26% and a deadline of 22%. Experimental results show that the fixed EES algorithm has the same reduction of energy.

III. H-DVFS ALGORITHM

In this research work, heuristic algorithm termed as HEFT algorithm has been utilized for assigning of task to processors with minimum execution time. HEFT algorithm is used to measure rank values for each and every tasks in DAG. HEFT algorithm is used to calculate rank values and then assign the tasks to an appropriate processor. HEFT algorithm is used to calculate the communication cost between the tasks in the DAG graph. The rank of job has been calculated starting from the last node and go in upward direction until the source node has not obtained [18].

The HEFT computation is a convincing answer for Scheduling issue of DAG on heterogeneous structure. Detention of HEFT computation is that it use procedures effectively that are all static methodologies of the mapping problem which allow static conditions for a particular time. HEFT is a two-part scheduling algorithm and different processors. For assigning tasks on the basis of priority, first phase of the HEFT is used, which is known as task prioritization phase. For assigning the priority, the upward rank of every task is determined. The upward rank is the critical path of the task that is used to calculate the maximum amount of communication time and the average performance time can be calculated from the beginning of any task to the end of any task[19]. The phase that is used to schedule the tasks into the practice gives the EFT of the task which is the Processor Selection phase (second phase). In second phase, DVFS algorithm is applied to the jobs that are already scheduled. The required time frequency of the host is measured and according to the time-tested measure that the Central Processing Unit (CPU's) are greatly increased or reduced by DVFS method to save energy[20]. DVFS is therefore is an energy conservation technique where overall power consumption is reduced by adjusting the frequency of the processor as needed [21]. At DVFS CPU frequencies it may be desirable to be called by the load and generally it can be activated in four different tasks:

i. More frequency
ii. Less frequency
iii. Obtainable CPU frequency
iv. Selection of the frequency level dynamically

Also, the concept of energy saving has been introduced by using the well known algorithm named as H-DVFS.

Algorithm :H-DVFS ALGORITHM

Algorithm: proposed H-DVFS algorithm
Input: G (T, E), computation energy of DVFS enabled processor
Output: Maximum energy consumed by H-DVFS
Design a DAG for the entire submitted jobs T_i
1.Defined the computation cost of task T_i as well as communication edges among the tasks.
2.For i=1to T_i
3.If $T_i \rightarrow last\ task\ in\ DAG$ then
4.Tasks are ordered as per the average of tasks on the entire processor
5.Else

6.Tasks are ordered as per the addition of average of tasks on all processor +max value of previous task of current task + communication between previous task to recent task
7.End if
8.End for
9.Organize tasks in a list in decreasing order as per their task's order value
10. For task in the list
11. Assign tasks to the processor that have less execution time and store their execution pattern
12. End for
13. End
14. Calculate size of Execution Pattern as row and column // Execution Pattern of the graph model
15. For each I in range(1 to row)
16. Calculate, Total Execution Time (I) = End Time(I)-Start Time(I)
17. Calculate, Processor value(I)= Normal Energy Consumption
18. $Energy\ consumed(I) = Total\ Execution\ Time(I) \times Processor Energy\ Consumtion(I)$
19. End For
20. Define parameters
21. Initial optimal frequency, fbest=f(1)
22. Occurrence of the Processor =[0 0 0]
23. Slack range=[0 0 0]
24. First processor=Normal Energy Consumption
25. Define, Processor occurrence(First processor)=1
26. For each I in range (2 to r-1)
27. Initialize occurrence flag; Flag=0
28. Current processor=Execution pattern(I, 4) // 4th column of Execution pattern
29. Check Current processor status,
Recheck Value=find out (processor occurrence==Current processor)
30. If Recheck Value is not empty then,
31. Processor occurrence(Recheck Value)=I
32.fbest=f(1) // Replacing the Optimal Value and consider as true
33. While fbest less than fhigh & flag equal to 0
34. Wtpk(I)= start Time of next task on selected processor-end time of previous task on selected processor
35. $wnew = wtpk \times fhigh/fbest$
36. $eftnew = StartTime + wnew$
37. If eftnew<execution_pattern(I+1,2)
38. Defined flag=1 for condition
39. Else
40. Increment in optimal frequency list, fbest=fbest+1
41. End If
42. End For
43. Else
44. Current(I)=Energy Consumption
45. $execution\_pattern(I,5) = (EndTime - StartTime) \times (idlep(currentp) + busyp(currentp))^\wedge 2/ f(currentp)$
46. End If
47. Return; Energy consumption for execution pattern
48. End Algorithm

A. TASK MODEL

The task- model may be represented by means of a DAG so as to consist of 'v' number of vertices with 'E' number of edges. The vertices of the graph can be varied from '0' to n-1 and the set of task is defined by V $\{t_i : i = 0,1,2,3,4, \dots \dots \dots \dots . . n-1\}$. The graph have dependency between t_i as well as t_j. Here, t_j starts execution after the completion of t_i. Every task t_i has a defined 'execution-time' on a particular 'processor represented by p_k through maximum supply -voltage as well as frequency.

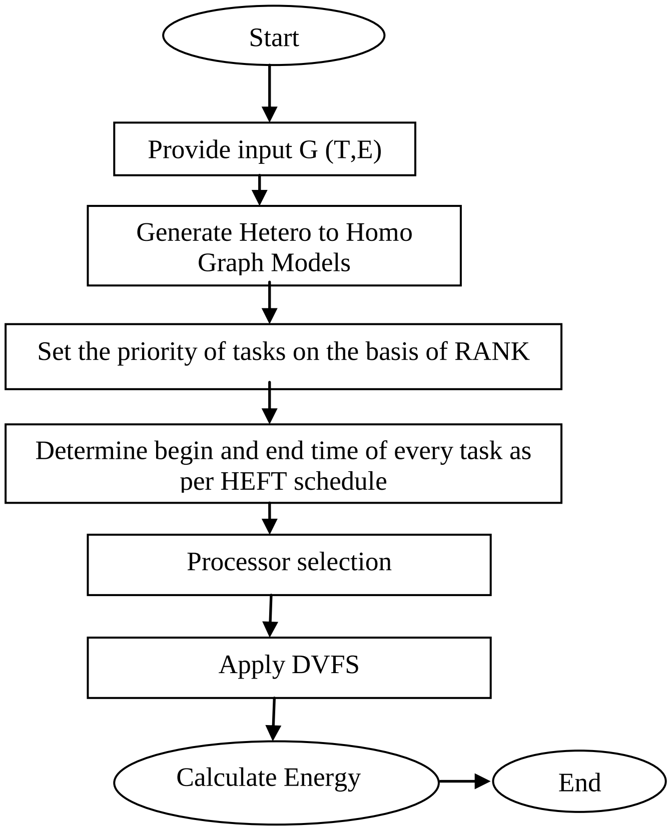

Fig. 1 Flowchart of proposed work

A. Computation Energy –Model

Let, the dynamic energy consumed by tasks t_i , scheduled on processor p_k with maximum supply- voltage and 'frequency' (V_1, f_1) can be written as the:

$$energy\ (t_i, p_k) = PP_{maximum} \times y((t_i, p_k)) \quad (2)$$

$Here, y((t_i, p_k)$ is the time of execution of task t_i on p_k processor.

$PP_{maximum}$ represents the power consumed by processor

The total dynamic consumed by all processors to execute 'T' task set is written by:

$$energy_{active} = \sum_{i=1}^{v} \sum_{k=1}^{p} energy\ (t_i, p_k) \quad (3)$$

Let us consider $PP_{mimimum}$ is the lowest energy consumed by processors in the idle state (i.e processor is not executing task) and hence the energy consumption rate can be written as:

$$energy_{idle}^{k} = PP_{mimimum} \times idle\ time \quad (4)$$

Computational tasks consume the idle energy as shown below:

$$= \sum_{k=1}^{p} energy_{idle}^{k} \quad (5)$$

Hence, the total dynamic energy of active and idle periods is given as:

$E_{total} = energy_{active} + energy_{idle}$

TABLE I
NOTATION USED

HPC	High Performance Computing
DAG	'Directed acyclic graph'
DVFS	'Dynamic voltage and frequency scaling'
HEFT	Heterogeneous Earliest Finish Time
DCS	Distributed computing system
EAD	Efficient and Adaptive Decentralized
CPU	Central Processing Unit
$y(t_i, p_k)$	amount of tasks and processors
$PP_{maximum}$	Total energy being observed by the processor
$PP_{mimimum}$	lowest energy observed by processors

IV. Result and Analysis

For determining the proposed work performance, we are considering a DAG and HEFT algorithm has been applied to determine the rank and processor selection of each task then existing HEFT algorithm integrate with DVFS technique to reduce the dynamic energy consumption of processor. The performance parameters such as SLR, energy consumption, and makespan are measured.

A. Processor Configuration details

The results are tested for random graphs using "AMD_Athlon 64 3000+ processor and is defined by three unique P-states. Consumption of power by three P-states in idle and busy case is shown in table below:

TABLE II.

DESIGN INFORMATION OF AMD_ATHLON 64 3000+ PROCESSOR

P-States	**Frequency**	**'Busy-state Power'**	**'Idle -State Power'**
1	2000 MHz	100 W	87 W
2	1800MHz	87 W	78 W
3	1000 MHz	61 W	59 W

Table III

COMPUTATION PARAMETERS

Number of tasks	**Make span**	**Energy Consumption**		**SLR**
		DVFS	**HEFT**	
10	74.16	4836.75	5909.36	1.08
20	67.8	10875.45	12065.72	1.08
30	99.28	12298.75	15477.01	1.05
40	135.51	16675.16	18840.86	1.05
50	165.41	22975.57	25220.08	1.033
60	168.84	23914.36	26270.08	1.03

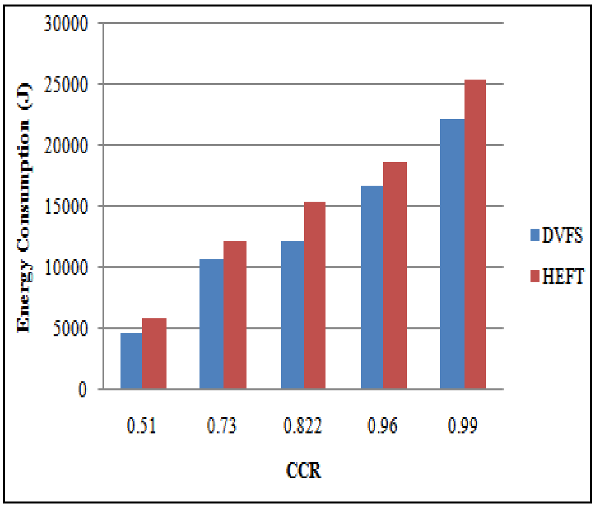

Fig. 2 Comparison of Energy Consumption

The above figure represents the comparison of HEFT task scheduling algorithm with DVFS scheme. X-axis defines the CCR values and y-axis defines the values of energy consumption obtained for the HEFT as well as DVFS algorithm. From the above graph it is clear that average value obtained for HEFT and DVFS are 17363.85 J and 14979.34 J respectively. It is observed that energy consumption has been reduced by 13.73%.

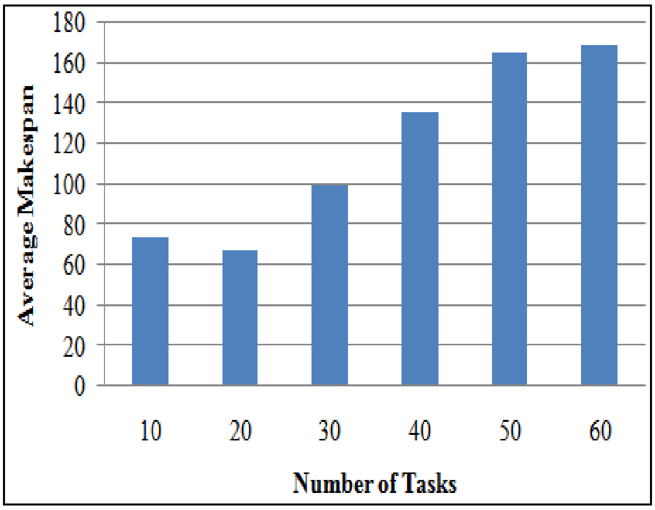

Fig. 3 Makespan

The above figure depicts the value of makespan obtained for the proposed work. As shown in the figure, it is clear that with the increment in the tasks, the value of makespan decreases. The average value of makespan obtained for the proposed work is 118.5.

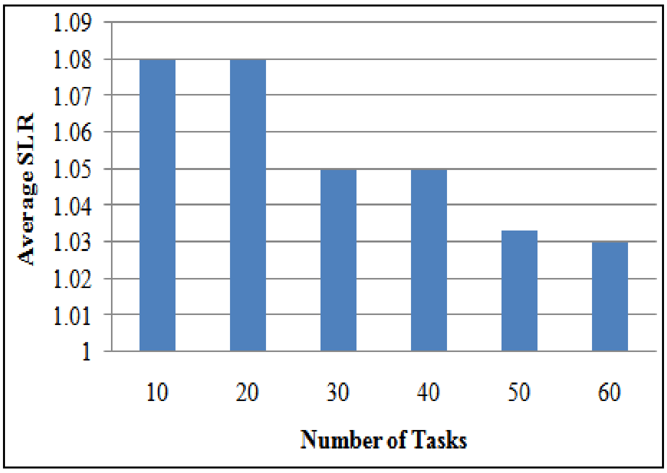

Fig. 4 SLR

The above figure represents the values of SLR obtained for the proposed work. From the above graph, it is clear that the average value obtained for SLR is 1.05.

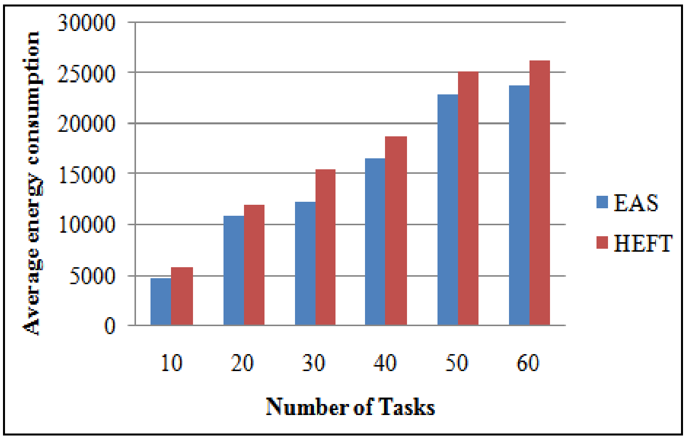

Fig.5 Evaluation of Energy Consumption

The above figure defines the evaluation for energy consumption while using DVFS as well as HEFT techniques. X-axis and y-axis represents the number of tasks and energy consumed by tasks. Red and blue bar represents the values of energy consumed by communication tasks in the presence of DVFS and HEFT algorithms reprehensively. The average value of proposed model using DVFS and HEFT algorithm are 17297.041 J and 15262.67 J respectively. Thus it is clear that when DVFS algorithm is used the energy consumed by the communicating tasks is reduced by 11.77%.

VI. CONCLUSION

In this manuscript, the energy consumption has been reduced using DVFS scheme. The DVFS approach is used to determine the best frequency level for obtaining enhanced work and the overall energy consumption has been reduced the CPU energy for the server. From the simulation work, it has been analyzed that DVFS has performed well and has reduced energy consumption value by 11.77%.

REFERENCES

[1]. Kaur, N., Bansal, S., & Bansal, R. K. , "*Energy efficient duplication-based scheduling for precedence constrained tasks on heterogeneous computing cluster*", Multiagent and Grid Systems, vol.12, no.3, (2016),pp. 239-252.
[2]. Kaur, N., Bansal, S., & Bansal, R. K. , "*Towards energy efficient scheduling with DVFS for precedence constrained tasks on heterogeneous cluster system*.", In Recent Advances in Engineering & Computational Sciences (RAECS), 2015 2nd International Conference on, (2015) December 1-6.
[3]. Liang, A., Liang, J., & Yuan, J., "*Energy aware scheduling based on two-phase frequency scaling for parallel tasks in cluster*", International Journal of Grid Distribution Computing, vol.8, no.5, (2015),pp 205-214.
[4]. Ruan, X., Qin, X., Zong, Z., Bellam, K., & Nijim, M. , "*An energy-efficient scheduling algorithm using dynamic voltage scaling for parallel applications on clusters*", In Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on , (2007) August, pp. 735-740.
[5]. Manzak, A., & Chakrabarti, C., "*Variable voltage task scheduling algorithms for minimizing energy/power*", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.11, no. 2, (2003), pp270-276.
[6]. Lee, Y. C., & Zomaya, A. Y., "*On Effective Slack Reclamation in Task Scheduling for Energy Reduction*," JIPS, vol. 5, no.4, (2009),pp. 175-186.
[7]. Nath, R., & Nagaraju, A., "*DVFS based heterogeneous scheduling for power optimisation and load balancing in HPC clustersaper*", In Parallel Computing Technologies (PARCOMPTECH), 2015 National Conference on (2015) February, 1-4.
[8]. Liang, A., Liang, J., & Yuan, J., "*Energy aware scheduling based on two-phase frequency scaling for parallel tasks in cluster*", International Journal of Grid Distribution Computing, vol.8, no.5, (2015), pp.205-214.
[9]. Ruan, X., Qin, X., Zong, Z., Bellam, K., & Nijim, M. , "*An energy-efficient scheduling algorithm using dynamic voltage scaling for parallel applications on clusters*", In Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on (2007), August, 735-740.
[10]. Manzak, A., & Chakrabarti, C, "*Variable voltage task scheduling algorithms for minimizing energy/power*", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.11, no.2,(2003),pp. 270-276.
[11]. Lee, Y. C., & Zomaya, A. Y., "*On Effective Slack Reclamation in Task Scheduling for Energy Reduction*", JIPS, vol. 5, no. 4, (2009), pp. 175-186.
[12]. Mei, J., & Li, K., ". *Energy-aware scheduling algorithm with duplication on heterogeneous computing systems*," In Proceedings of the 2012 ACM/IEEE 13th International Conference on Grid Computing , (2012) September , 122-129).
[13]. Nath, R., & Nagaraju, A., "*DVFS based heterogeneous scheduling for power optimisation and load balancing in HPC clustersaper*", In Parallel Computing Technologies (PARCOMPTECH), 2015 National Conference on (2015), February, 1-4.
[14]. Xie, G., Zeng, G., Xiao, X., Li, R., & Li, K. , "*Energy-efficient scheduling algorithms for real-time parallel applications on heterogeneous distributed embedded systems*.", IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 12,(2017),pp. 3426-3442.
[15]. Arabnejad, H., Barbosa, J. G., & Prodan, R., "*Low-time complexity budget–deadline constrained workflow scheduling on heterogeneous resources*", Future Generation Computer Systems, vol. 55, (2016), pp.29-40.
[16]. Wang, L., Khan, S. U., Chen, D., KołOdziej, J., Ranjan, R., Xu, C. Z., & Zomaya, A. , "*Energy-aware parallel task scheduling in a cluster", Future Generation Computer Systems*, vol. 29, no. 7, (2013), pp. 1661-1670.
[17]. Ebaid, A., Rajasekaran, S., Ammar, R., & Ebaid, R*, "Energy-aware heuristics for scheduling parallel applications on high performance computing platforms,"* In Signal Processing and Information Technology (ISSPIT), 2014 IEEE International Symposium on (2014) December, 000282-000289).
[18]. Xie, G., Zeng, G., Xiao, X., Li, R., & Li, K. , "*Energy-efficient scheduling algorithms for real-time parallel applications on heterogeneous distributed embedded systems*", IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 12, (2017), pp. 3426-3442.
[19]. AlEbrahim, S., & Ahmad, I., "*Task scheduling for heterogeneous computing systems*," The Journal of Supercomputing, vol. 73, no. 6, (2017), pp.2313-2338.
[20]. Tang, Z., Qi, L., Cheng, Z., Li, K., Khan, S. U., & Li, K, "*An energy-efficient task scheduling algorithm in DVFS-enabled cloud environment*,"Journal of Grid Computing, vol. 14, no. 1, (2016), pp.55-74.
[21]. Tang, Z., Cheng, Z., Li, K., & Li, K, " *An efficient energy scheduling algorithm for workflow tasks in hybrids and DVFS-enabled cloud environment,"* In Parallel Architectures, Algorithms and Programming (PAAP), 2014 Sixth International Symposium on , (2014), July, pp. 255-261.

IJTIMES-2019@All rights reserved