Global Journals LATEX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

Developing Frequency Falling EDF with Relatively Greater Power Efficiency and Low Deadline Miss Ratio

Mohammad Nowsin Amin Sheikh[1], Nazmul Hossain[2] and Ripon Kumar Saha[3]

1

Received: 10 December 2016 Accepted: 4 January 2017 Published: 15 January 2017

Abstract

Many scheduling algorithms are available for the real -time system, which maintains hard deadline to solve the issues related to the time that is critical for scheduling in real time system and to provide a better system design in real -time system avoiding poor and erroneous choices for scheduling algorithms. The system is based on a real -time deals with the resource to ensure maximum performance and utilization in real -time. Processor availability plays the main role in choosing the best scheduling algorithm for a real -time system. D VFS is being used extensively for the technique of energy management. The aim of DVFS platform is to minimize energy consumption. In this paper, we will give a new algorithm for DVFS and compare the power consumption and deadline miss ratio of other RT-DVF S algorithm with our algorithm. There are many real time dynamic voltage frequency scheduling (DVFS) algorithms. We have analyzed two independent under loaded task -sets for RT -DVFS scheduler algorithm that is Base -EDF and Static -EDF and devise a new DVFS s cheduler algorithm named Frequency Falling EDF. Our devised FF-EDF algorithm is more efficient than Base -EDF algorithm in terms of power consumption. It also gives better performance than static-EDF in terms of future deadline handling. FF -EDF algorithm focuses on dynamic voltage frequency scheduling.

Index terms— RT; DVFS; RT-DVFS; EDF; static EDF; WECT; FF-EDF.

1 Introduction

he usages of energy are growing rapidly with the increase of portable devices, embedded system, automation and much real time devices with its energy consuming application. Research is going on to provide better power efficiency both in hardware and software level [1]. In dynamic voltage frequency scheduling(DVFS) many scheduling algorithms are available which can provide a great power efficient task schedule system, but almost not usable in real time system as they have performed very low in case of deadline miss ratio. We have analyzed those algorithms and unlike other DVFS algorithms, we have developed an algorithm named Frequency Falling EDF (FF-EDF) which can provide a greater power efficiency with it's dynamic frequency and also can perform very well in case of future task execution without a great dealing of deadline miss ratio. Here we have presented FF-EDF with it's pseudo code, mathematical model and working principle with it's comparison to Base EDF and Static EDF in case of power consumption and deadline miss ratio. It has been shown here that Frequency Falling EDF can deliver on average 2X power consumption in case of Base EDF without missing relatively as much deadline as other DVFS algorithm miss.

DVFS is a technique that is used in operating system level for optimizing power consumption. When CPU is active its power consumption is calculated by Pactive = C * F3 [2], where C is a constant, F is the speed or frequency of the processor and Pactive is processor active power consumption. Therefore, energy saving highly depends on the number of frequency of the processor while running a task.

RT-DVFS scheduling algorithm takes two important decisions. Firstly, which task we should run and secondly which frequency it should run. Static slack and dynamic slack is available for this algorithm. Static slack depends on the characters of task set and dynamic slack is available for variation of execution time. Based on the amount of tasks and their execution time with the actual period Frequency Falling EDF start from a higher frequency and start to lower down the frequency over time to the end of the period. In case of future tasks, it again increases it is frequency to execute the new task within the period. In this paper we have shown up the mathematical model of FF-EDF, how it works and from which frequency it will start and to which frequency it will go. Simulation is also done by a C++ program to compare this developed algorithm with Base EDF and static EDF.

A Significant amount of research has been done in the field of RT-DVFS. Pillni and Shin devised five RT-DVFS algorithms and found that EDF based schedulers outperform the RMA based one [8]. RT-DVFS algorithms are designed for real-time systems and aim at saving energy while maintaining hard real-time constraints.

2 T

3 Global Journal of Researches in Engineering ( ) Volume XVII Issue III Version I

They scale the CPU frequency based on the worst-case execution times of the real-time application. Most of the RT-DVFS algorithms differ in their techniques to utilize the static slack available due to the low CPU utilization of the application or dynamic slack available due to the actual execution time being much lesser than the worst case execution time of the real-time application. We have observed that the performance of a RT-DVFS algorithm is highly dependent on the energy efficiency of the idle states of the processor [2]. Energy constraints real time scheduling is discussed and developed by T. A. AlEnawy and H. Aydin [3]. Energy minimization is found by E. Bini, G. Buttazzo, and G. Lipari [4].

Performance comparison of dynamic voltage scaling algorithms for hard real-time systems [7]. Whole system power measured by P. Pillai and K. G. Shin [8].Energy efficient real time task scheduling by C.-M. Hung, J.-J. Chen, and T.-W. Kuo [6]. Voltage scaling for mobile multimedia by W. Yuan and K. Nahrstedt [13]. Optimal procrastinating voltage scheduling for hard realtime systems by Y. Zhang, Z. Lu, J. Lach, K. Skadron, and M. R. Stan [14].Energy efficient real time operating system by Gordon Parke [17]. Dynamic voltage scaling in embedded real time system by Rubathas Thirumathyam [19].

4 II.

5 Algorithms

There are two types of slacks e.g. static slack and dynamic slack and RT-DVFS Algorithms use these two types of slacks. Three RT-DVFS schedulers are evaluated based on their performance in this section we will discuss about these three algorithms. Two important decisions have to make by RT-DVFS schedulers (i) which task to run and (ii) which frequency to run it at. Each other differ in a way, when they estimate slack to scale the frequency [7]. static slack, which is available due to the characteristic of the task-set itself, such as less than 100% CPU utilization, and dynamic slack, which is available due to variations in the execution time [7].

We will describe the algorithms with the help of an example task-set. Let us consider a three task periodic task-set with tasks T1, T2 and T3 whose characteristics are described by the Table **??**. The Base EDF scheduler does not involve any type of frequency scaling and run at maximum frequency [8]. The task scheduling is based on the earliest deadline and every task is run at maximum frequency. We have included this experiment for comparison.

6 Fig.1: Base-EDF

Static-EDF scheduler uses static slack estimation technique for the CPU frequency to scale up [8]. Based on the utilization value that is static, the task set is used to scale the frequency. From the pseudo code of Base-EDF, we can see that, all the task is running at same frequency so that the utilization of processor is scaled frequency that is 1. This algorithm makes sure that no deadline will miss, although the utilization is equal to or less than 1. The main purpose of this algorithm is to minimize the idle time. Discrete frequency behavior is found in non-ideal processor. A frequency is equal to or less than k is selected for the selected task to run.

7 Roposed Frequency Falling Edf

In our optimized EDF, from the task set of table 1 if we run Static -EDF algorithm Utilization of task set will, U = So U=0.8 as we know that our maximum frequency fm=1 all the task will run at (U*fm) = (1*0.8) =0.8 frequency. As we can see that, STATIC EDF did not consider actual time. In Frequency Falling EDF, initial frequency and ending frequency will follow this rule. Let the frequency we get form STATIC EDF is Fs, FF-EDF initial frequency is Fi and ending frequency is Fe.

Then if F s > 0.5 * f m F i = f m , and F e = 2* F sf m In Static-EDF F s < 0.5 f m F i = 2 * F s and F e = 0. Fig. **??**: Proposed Frequency Falling EDF So in that case our frequency will be: -Fi = 1 and Fe = 2

* 0.8 -1 = 0.6 When a task start for any time frame the amount of portion of task set completed by Frequency Falling EDF is larger than Static-EDF, because FF-EDF try to finish task sooner as the frequency is higher in the first half of the total time frame of combined task set to complete. And in case of task set that occur in last half section of time frame will also execute sooner as the task before that section had executed before the time frame. Moreover, theoretically the last task will finish it execution at the same time as the Static-EDF but with a lower frequency.

From the Figure **??**, new task come at 6s whose period is 4s and worst case execution time is 2. From the Figure **??**, new task come at 6s whose period is 4s and worst case execution time is 2.4s. Now in case of Static-EDF time remaining is = 10 -6 =4s, task remaining for previous task set in Static-EDF is = 4*0.5 = 2.0s, as new task come total task remaining for Static-EDF is =2.0s +2.4s =4.8s Now our Static -EDF should run at (4.4/4) = 1.1 frequency which is beyond the limit of our maximum frequency. So task set will fail to run in terms of Static -EDF in future task handling. For our algorithm, Time remaining is =10-6=4s, task remaining for our algorithm is = 0.5 * 4 * 0.4 = 0.8, as new task come total task remaining for our algorithm is =0.8 + 2.4 = 3.2s. As Fs> 0.5 Staring Fi = fm = 1 Ending Fe = (2 * 0.8) -1 =0.6.

8 IV. Energy Consumption In Two Approaches

Base-EDF always runs the task set with maximum frequency f = fm = 1 and considering texecution = execution time, we can calculate the power consumption of Base-EDF, ?? = ð ??"ð ??" 3 × ?? ???????? ?????????? ?? = 1 3 × 5 = 5

Static-EDF run the task with utilization U = 5/10 = 0.5 So U=0.5 as we know that our maximum frequency Fm=1 all the task will run at (U *Fm) = (1*0. In case of our developed FF-EDF, we can assume the initial frequency = fi, Ending frequency = fe, the total time to run the task = t. So, we see that at 10s period and with 5s execution time considering maximum frequency 1 unit the power usage of Base EDF, Static EDF and FF-EDF are consecutively 5, 1.25 and 2.5 unit. Therefore, FF-EDF consumes half power with compare to Base EDF and double power with compare to Static EDF. However, this will be not the case in most of the time. It actually depends on the density of task, which is the ration of execution time and period. Here is the data simulated from Static EDF, Base EDF and FF-EDF:- So, we have presented this graph with different task density. Here in X-axis task density is presented that defer from 0 to 10 and in Y-axis the power consumption is presented. As we can see that from the above chart that for very low value of time FF-EDF algorithm give less power than Base EDF and Static EDF. If time increases FF-EDF power consumption is better than Base EDF but not better than Static-EDF.

In case of deadline miss ratio, we have tested this entire algorithm by 1 million of task trial at 10 different task densities. Here task density can be defined as the static frequency that it may need to run to execute its entire task within the period.

Base EDF is best algorithm in case of executing the largest number of task without meeting deadline, because it executes its entire task at maximum frequency. It only falls its frequency when there is no task remaining at CPU.

Static algorithm is the best algorithm in case of power saving. However, in case of deadline miss ratio it has the worst performance. Because it runs the entire task at the same frequency. Therefore, when a new task

9 Base EDF

Static EDF FF-EDF comes may not be the time to run the new task at its worst-case execution time. FF-EDF works as a great tradeoff between these two algorithms. Because this developed algorithm starts the task at a higher frequency and start to fall down by a slope to a lower frequency. By this, it adjusts the power and deadline miss. Here we can see that at the start of the time it has higher execution rate (very similar to Base EDF) with a very low deadline miss ratio and over time, it starts to decrease the execution rate. We have compared the Base EDF, Static EDF and FF-EDF by simulation by a C++ program. This program is run at a different frequency. We have assumed that the max frequency is 1 GHz for simplicity. In addition, the simulation is done by taking 10different frequency distributions from 0.0 to 0.9 by a difference of 0.1. The test is done to check how many deadlines usually missed by Base EDF, Static EDF and FF-EDF when new tasks come. The new task can come at any time, at any period and worst-case execution time. For this reason, we have selected starting time, Period and Worst-case execution time randomly. By the simulation, we have taken 1000 trials in each frequency distribution and checked if those algorithms miss the deadline or not. We can see that FF-EDF works as a tradeoff between the base EDF and the static EDF in a efficient way. This algorithm is modeled with its pseudo code, and mathematically described how it works with its task set. Also, it has been shown how future task can be handled via changing the instant frequency and the slope of the frequency. The result we got from the simulation and the algorithm process we can decide that in case of RT it will be far useful.

FF-EDF has better power optimization (on average 2X) with compare to base EDF without compromising the deadline miss ratio as much as Static EDF. The Power optimization we got by developed EDF for this deadline miss ration with compared to the Base EDF and Static EDF is relatively high. In the graph, it has been shown that the deadline miss ratio is closer to Base EDF at the start of time. Frequency Falling EDF has designed in such a way that, it can take much load when a future task comes because it has much less task When a process arrives, it is better to run a task as soon as it arrives as higher frequency as possible because in that case, it is possible to run the entire remaining task at a higher frequency if needed to do that without missing any deadline. In addition, Frequency Falling EDF works as a trade-off between power optimization and deadline miss ratio of EDF. The Base EDF is very power consuming algorithm but has the lowest deadline miss ratio. And The Static EDF has the most efficient in case of power consumption but it almost not efficient because of its deadline miss ratio. For this reason, Static EDF is almost not usable in real time scheduling. So, considering this FF-EDF can be used effectively in any real time system scheduler.

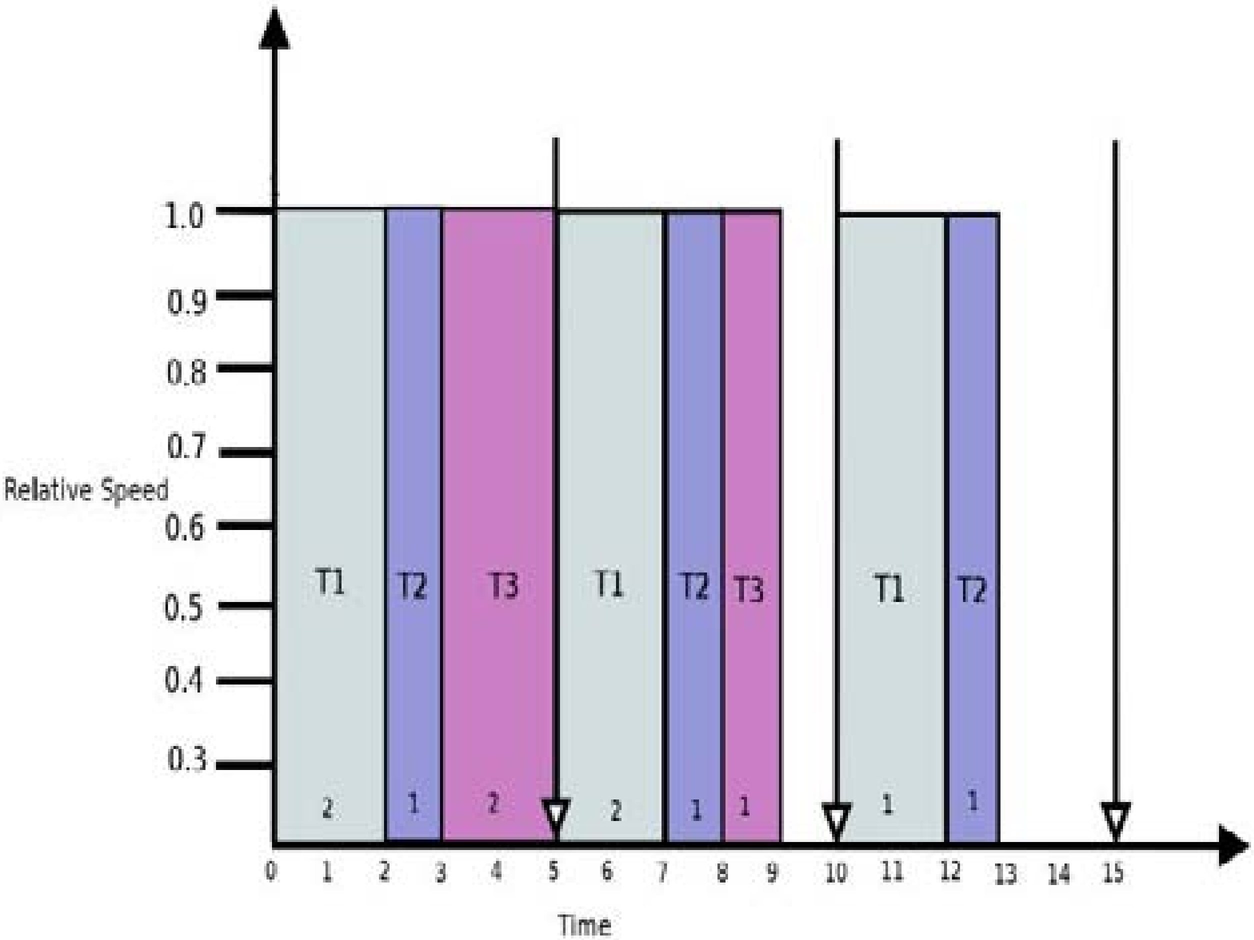
There is a limitation of FF-EDF that in this paper we have modeled and simulated the algorithm via worst case execution time. But it can be better performed via actual time. This developed EDF will automatically adjust the frequency for its actual execution time. So, the actual execution time is avoided for simplicity and to keep the simulation accurate.

V.

10 Conclusion

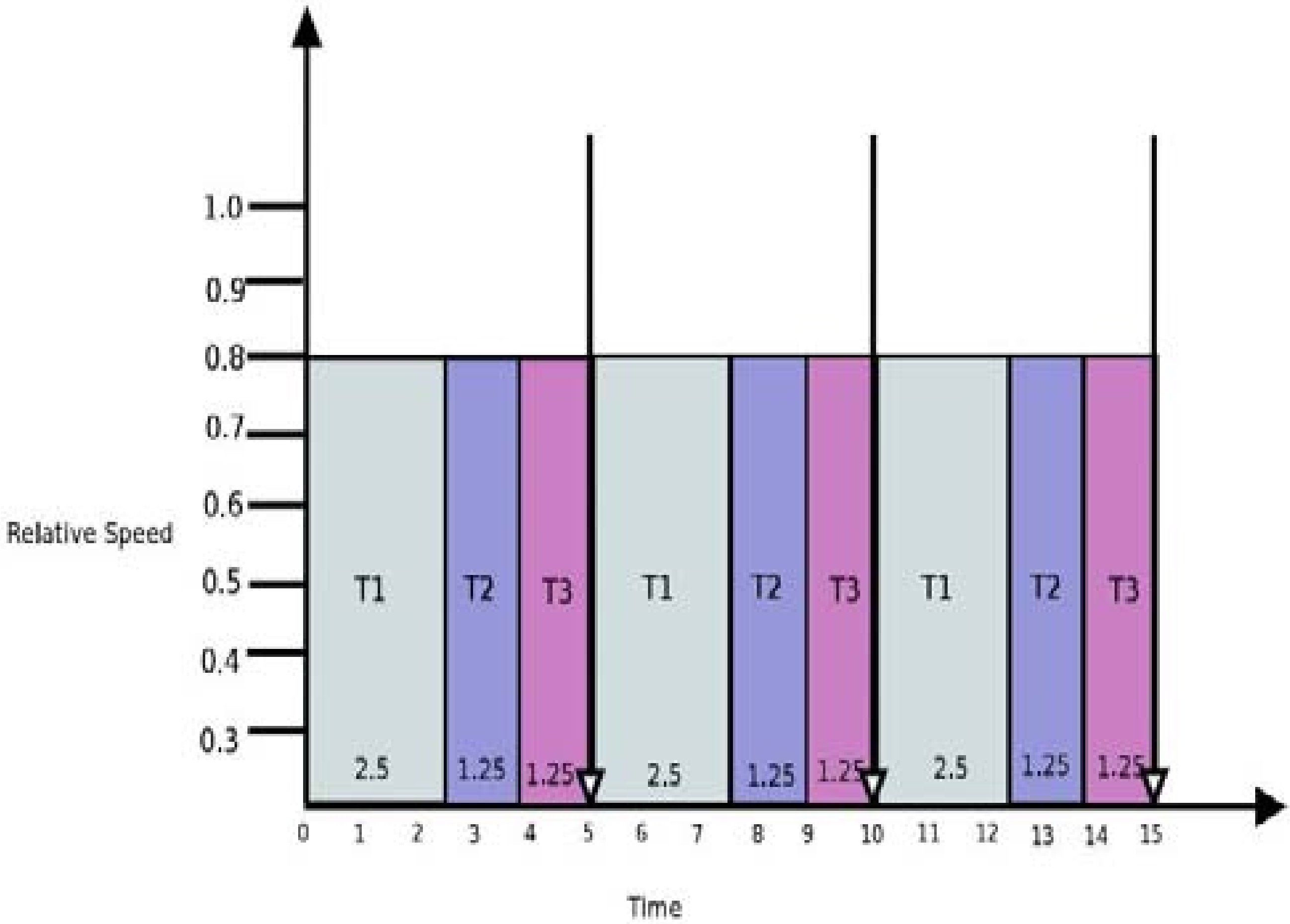
FF-EDF is working with worst-case execution time. FF-EDF frequency grows in a linear way for the worst-case execution time. If FF-EDF can work with actual time CPU frequency will drop.

We have simulated FF-EDF, but in case of real system, our result may vary for both power consumption and deadline miss ratio. FF-EDF work based on worstcase execution time of a task and did not consider actual time. If we consider actual time, FF-EDF will give a better result for both power consumption and deadline miss ratio. Base-EDF and Static-EDF keep the linear frequency, but in case of FF-EDF, the frequency will fall within a range. [1,2,3,4,5,6]



2

Figure 1: Fig. 2 :



622

Figure 2: 6 Table 2 . 2 :

Figure 3:

44

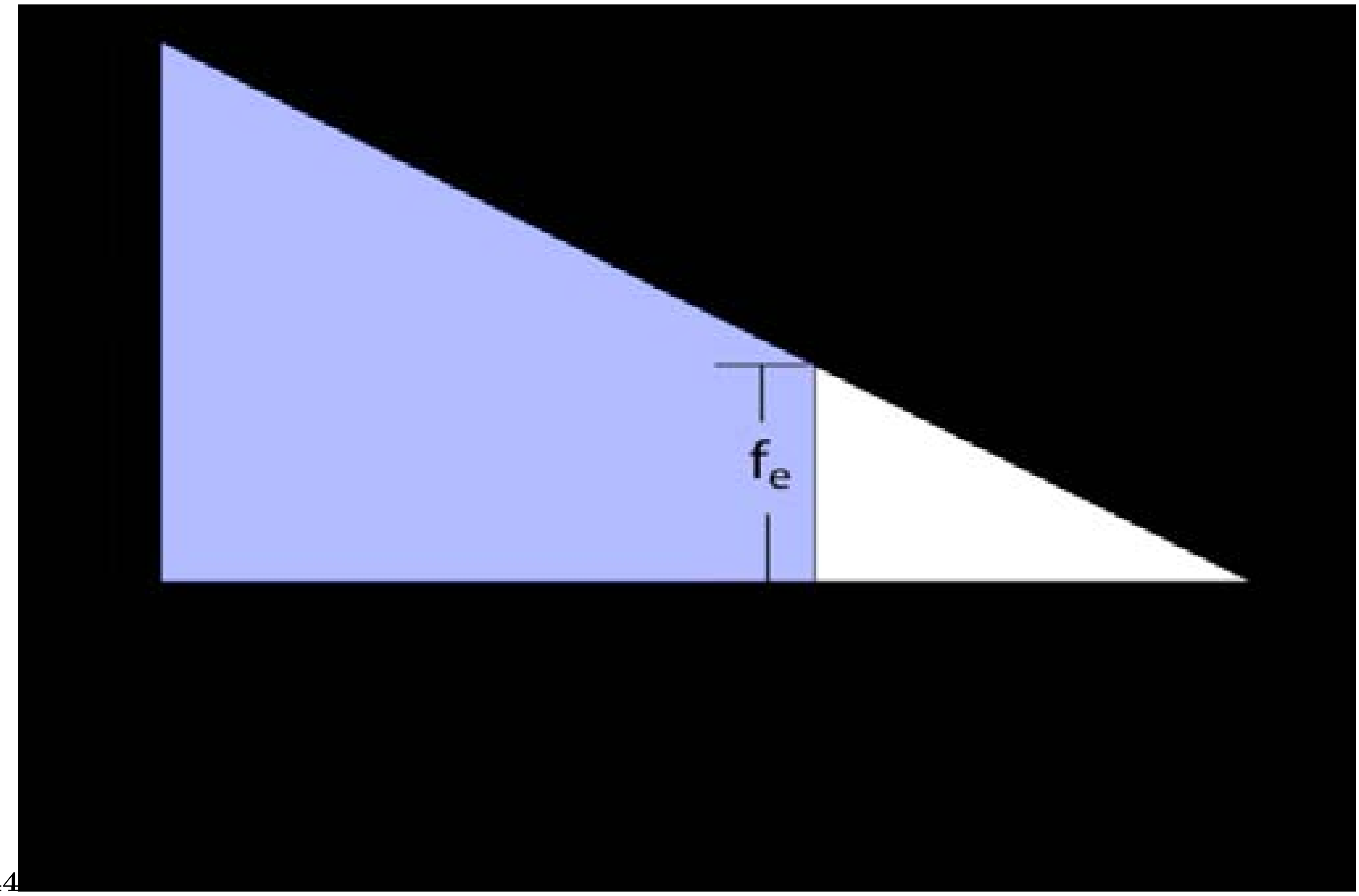


Figure 4: Fig. 4 : 4

5

Figure 5: Fig. 5 :

6

Figure 6: Fig. 6 :

21

Task	WCET	Actual Time	Period
T 1	2	1.6	5
T 2	1	0.8	5
T 2	3	2.4	15

Figure 7: Table 2 . 1 :

23

Task Density	Base EDF	Static EDF	FF-EDF
0	0	0	0
1	1	0.01	0.02
2	2	0.08	0.16
3	3	0.27	0.54
4	4	0.64	1.28
5	5	1.25	2.5
6	6	2.16	3.12
7	7	3.43	4.06
8	8	5.12	5.44
9	9	7.29	7.38
10	10	10	10

Figure 8: Table 2 . 3 :

24

Task Density	Base EDF	Static EDF	FF-EDF
0	1000000	853253	853253
0.1	995627	747090	985787
0.2	986541	678417	945665
0.3	952076	596407	906522
0.4	926187	497096	852308
0.5	853253	497090	795678
0.6	804899	375987	681505
0.7	678417	221407	564977
0.8	596407	221408	404927
0.9	375988	8	279589

Figure 9: Table 2 . 4 :

1 1000 994 975 948 928 856 759 698 605 510 504 856 808 391 845 800 697 2 3 4 5 6 7 8 1000 975 936 910

Base EDF	Statc EDF	Our Develped EDF

Figure 10:

[1] Year 2017 F © 2017 Global Journals Inc. (US)

[2] = 0.8 © 2017 Global Journals Inc. (US)

[3] © 2017 Global Journals Inc. (US)Developing Frequency Falling EDF with Relatively Greater Power Efficiency and Low Deadline Miss Ratio

[4] © 2017 Global Journals Inc. (US)

[5] Year 2017 F © 2017 Global Journals Inc. (US) Developing Frequency Falling EDF with Relatively Greater Power Efficiency and Low Deadline Miss Ratio

[6] F © 2017 Global Journals Inc. (US) Developing Frequency Falling EDF with Relatively Greater Power Efficiency and Low Deadline Miss Ratio

[Saha Binoy Ravindran ()] 'An Experimental Evaluation of Real-Time DVFS Scheduling Algorithms'. *Proceedings of the 5th Annual International Systems and Storage Conference*, Sonal Saha, & Binoy Ravindran (ed.) (the 5th Annual International Systems and Storage ConferenceHaifa, Israel) 2012.

[Cho et al. ()] 'An Optimal Real-Time Scheduling Algorithm for Multiprocessors'. Hyeonjoong Cho , & E Douglas Binoy Ravindran , Jensen . *Proceedings of the 27th IEEE International Real-Time Systems Symposium (RTSS'06)*, (the 27th IEEE International Real-Time Systems Symposium (RTSS'06)Washington, DC, USA) 2006. p. .

[Liu Qiu Qing Wu ()] 'Energy aware dynamic voltage and frequency selection for real-time systems with energy harvesting'. *Proceedings of the conference on Design, automation and test in Europe, DATE '08*, Shaobo Liu, Qinru Qiu & Qing, Wu (ed.) (the conference on Design, automation and test in Europe, DATE '08New York, USA) 2008. p. .

[Hung and Kuo ()] 'Energy-efficient real-time task scheduling for a dvs system with a non-dvs processing element'. Chia-Mei Hung , Jian-Jia Chen & Tei-Wei Kuo . *Proceedings of the 27th IEEE International Real-Time Systems Symposium*, (the 27th IEEE International Real-Time Systems SymposiumWashington, DC, USA) 2006. p. .

[Zhuo and Chakrabarti ()] 'Energyefficient dynamic task scheduling algorithms for dvs systems'. Jianli Zhuo , & Chaitali Chakrabarti . *ACM Transactions on Embedded Computing Systems (TECS)* 2008. 7 (2) p. .

[Yuan Nahrstedt ()] 'Energyefficient soft real-time cpu scheduling for mobile multimedia systems'. *Proceedings of the nineteenth ACM symposium on Operating systems principles, SOSP '03*, Wanghong Yuan, & Klara Nahrstedt (ed.) (the nineteenth ACM symposium on Operating systems principles, SOSP '03New York, USA) 2003. p. .

[Chen and Lee ()] 'General Multiprocessor Task Scheduling'. Jianer Chen , & Chung-Yee Lee . *Global Journal of Researches in Engineering* 1999. 46 (1) p. . (Naval Research Logistics. III Version I)

[Jagadeeshraja et al. ()] 'Implementation and Analysis of RMS scheduling in Real Time Operating System'. M Jagadeeshraja , B Sasikumar , A Ramesh , N Dr , Suthanthira , Vanitha . *Computers & Applied Sciences* 2013. 2 (11) p. . (JEC&AS))

[Bini et al. ()] 'Minimizing cpu energy in real-time systems with discrete speed management'. Enrico Bini , Giorgio Buttazzo & Giuseppe , Lipari . *ACM Transactions on Embedded Computing Systems (TECS)* 2009. 8 (4) p. .

[Alenawy and Aydin ()] 'On energyconstrained real-time scheduling'. T A Alenawy , & H Aydin . *Proceedings of the 16th Euromicro Conference on Real-Time Systems*, (the 16th Euromicro Conference on Real-Time SystemsWashington, DC, USA) 2004. p. .

[Dabiri et al. ()] 'On energyconstrained real-time scheduling'. Foad Dabiri , Alireza Vahdatpour , Miodrag Potkonjak & Majid , Sarrafzadeh . *Proceedings of the Conference on Design, Automation and Test in Europe, DATE '09*, (the Conference on Design, Automation and Test in Europe, DATE '09Leuven, Belgium) 2009. p. .

[Thirumathyam ()] 'Online Dynamic Voltage Scaling in Embedded Real-Time Systems'. Rubathas Thirumathyam . *Master of Science Thesis, Department of Informatics and Mathematical Modelling (IMM)*, 2012. Technical University of Denmark

[Zhang et al. ()] 'Optimal procrastinating voltage scheduling for hard real-time systems'. Yan Zhang , Zhijian Lu , J Lach , K & M R Skadron , Stan . *Proceedings of the 42nd annual Design Automation Conference, DAC '05*, (the 42nd annual Design Automation Conference, DAC '05New York, USA) 2005. p. .

[Woonse Ok Kim et al. ()] 'Performance comparison of dynamic voltage scaling algorithms for hard real-time systems'. Dongkun Woonse Ok Kim , Han-Saem Shin , Yun . *Proceedings of the 8th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'02)*, (the 8th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'02)Washington, DC, USA) 2002. p. .

[Yuan Nahrstedt ()] 'Practical voltage scaling for mobile multimedia devices'. *Proceedings of the 12th annual ACM international conference on Multimedia, MULTIMEDIA '04*, Wanghong Yuan, & Klara Nahrstedt (ed.) (the 12th annual ACM international conference on Multimedia, MULTIMEDIA '04New York, USA) 2004. p. .

[Pillai et al. ()] 'Realtime dynamic voltage scaling for low-power embedded operating systems'. Padmanabhan Pillai , & Kang , G Shin . *Proceedings of the 18th ACM symposium on Operating systems principles, SOSP '01*, (the 18th ACM symposium on Operating systems principles, SOSP '01New York, USA) 2001. p. .

[Nazmul Hossain, Md. Alam Hossain, A. K. M. Fayezul Islam, Priyanka Banarjee Tahira Yasmin ()] 'Research on Energy Efficiency in Cloud Computing'. *International Journal of Scientific & Engineering Research* Nazmul Hossain, Md. Alam Hossain, A. K. M. Fayezul Islam, Priyanka Banarjee & Tahira Yasmin (ed.) 2016. 7 (8) p. .

[Zhuo and Chakrabarti ()] 'Systemlevel energy-efficient dynamic task scheduling'. Jianli Zhuo , & Chaitali Chakrabarti . *Proceedings of the 42nd annual Design Automation Conference, DAC '05*, (the 42nd annual Design Automation Conference, DAC '05New York, USA) 2005. p. .

[Parke ()] *ThermOS: An Energy Efficient Real-Time Operating System For An Ultralow Power Embedded Device*, Gordon Parke . 2010. Institute of Computing Systems Architecture, School of Informatics, University of Edinburgh (Master of Science Thesis)

[Höglund ()] *Voltage and frequency scaling in an embedded microprocessor to enable the implementation of dynamic voltage and frequency scheduling for power management*, Jonas Höglund . 2009. Master of Science Thesis, Royal Institute of Technology