The comparison WRR-SCAN versus SCAN-EDF the algorithms that are used in REAL-TIME systems.

Julian Fejzaj[1], Aldo Frasheri[2]

[1]University of Tirana, Faculty of Natural Sciences, Department of Informatics

[2]Polytechnic University of Tirana, Faculty of Information Technology

ABSTRACT

Scheduled algorithms in disc, have a very important role in real time applications used in multimedia, in as much as the last ones generate I/O demands with their restrictions in real time. Both traditional algorithms and other two ones mentioned later during this study, serve to real-time tasks according their deadlines. However, reaching the goal is also accompanied with costs causing in main parameters such as: searching time in the disc, rotation time, the time in which disc's head stays totally free, overhead during transmission, disk utilization, throughput and response time to non periodic tasks. The aim of this study consists on determining the best algorithm on parameters accomplishing and, as well as the definition of parameter they try to optimize in particular. WRR-SCAN spends less time during the search through disc, while SCAN-EDF generates less disc rotations. Compared to SCAN-EDF, WRR-SCAN reduces a lot overhead during no-transmission, increases disc usage and offers an affordable response time toward non periodic tasks.

Keywords: *WRR-SCAN, SCAN-EDF, disk utilization, aperiodic time, disk bandwidth***.**

1. Introduction

Real time applications including database, multimedia applications ask for their I/O demands to be served in a limited time.

Traditional algorithms aim to optimize throughput without taking into consideration time limits. They include: First Come First Serve (FCFS), Shortest Seek Time First (SSTF), SCAN, Cyclical SCAN. In order to satisfy time limits, there are proposed some other limits where mentioned: Earliest Deadline First (EDF), Least Slack Time First and Rate Monotonic, that can be used to schedule I/O demands. These kinds of algorithms ignore the positions of data in the disc. Due to delay in search through the disc and retardation in rotation, their throughput is relatively low.

These types of algorithms are nothing else than a combination between those who try to optimize research time throughout the disc and those who gratify deadlines. Inside this set of there are also two other algorithms which will be treated later during the study description. WRR-SCAN is a scheduled algorithm based on rate, otherwise SCAN-EDF is a type of algorithm based on priority.

Weighted Round Robin-SCAN, WRR-SCAN schedules real time requirements between all those non periodic requirements to send in the same time data to real time tasks and to ensure a minimal data rate for non periodic tasks. Bandwidth of the disc is divided into cycles, along each cycle disc head moves in one direction and serves to tasks in a scan way. A weight is set to every real time tasks that shows the time during which the disc is transmitting and serving to that task inside the cycle. Reserving the bandwidth asked for every task, is guaranteed a specified minimal data rate even a limited time for the maximum response.

First of all, SCAN-ED helps in serving tasks with the closest deadline, but if there are requirements with the same deadline, they are served due to positions of their relative tracks in the disc. SCAN-EDF serves to non periodic tasks only if there are no current real time periodic tasks. This causes a dependence of non periodic throughput tasks on real time tasks workload.

The study target estimates and compares algorithms in terms of: total distance through disc research, the time during the disc stays totally free, overhead inserted from the time of research and rotation, exploitation of the disc, throughput and response time from non periodic tasks.

The results obtained from the study shows that serving to any task with a reserved weight due to the scan manner, WRR-SCAN reduces significantly the time of the research, improves disk utilization and gives a response time to the

non periodic tasks in an affordable way, except the save of deadlines for real time tasks.
The remaining part of the study is organized as follows. Section 2 gives a description of according to traditional algorithms and real time of scheduling. Section 3 describes the algorithm WRR-SCAN and SCAN-EDF. Implementation and experimental results are presented in section 4. Conclusions are presented in section 5.

2. Algorithm SCAN and EDF

This is a hybrid scheduled algorithm, a combination between SCAN and EDF. The first one is known as a lift algorithm, it means that the disks serves all asks in the direction that they came one by one.
SCAN-EDF combines these two algorithms as below. Requirements with the closest deadline are served at first as in EDF, but if there are requirements with the same deadline, it is served according to the positions of the tracks in which they are placed into the disc or using scheduled algorithms disc which optimize the seek.
The efficiency of this algorithm is linked up with the number of requirements with the same deadlines or said in other words on how often these optimize seeks can be applied. Rescheduling is possible only inside a local group of requirements. This algorithm requires that the requirements release time should be multiple of period, which causes requirements with a deadline being a multiple of periods. The problem consists on cases when requirements have different needs for data rate, SCAN-EDF matches with a period fill technique which makes the requirements to have the same deadline. Requirements are served under a cycle where every requirement takes a period of time service in proportional to required data rate.
The duration of the cycle is equal to the total time of service of all requirements. This technique increases the possibility of optimize seek application SCAN-EDF.

Implementation

If Di is the deadline of task i and Ni is the track position, deadline can be modified to Di + F (Ni).
Note: The function F() gives priority to requirements with the same deadlines according to their positions in the disc.
Figure 1 illustrates an example of SCAN-EDF algorithm with: Nmax=100 and F (Ni) = Ni/Nmax

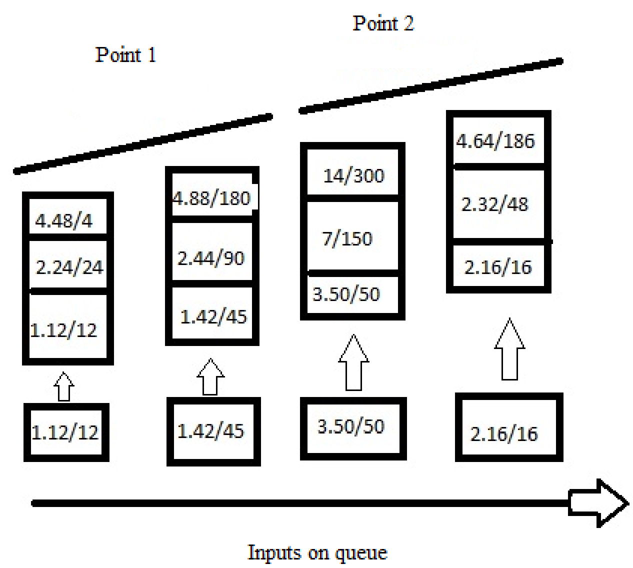

Fig.1: SCAN-EDF illustration

2.1 Algorithm WRR-SCAN

To guarantee quality on system, working on periodic and non periodic tasks we use Wrr-Scan: Weighted Round Robin over Scan scheduling input/ outputs. This algorithm puts the data rates on every requirement of the tasks. In general, WRR shares into proportions available service time (bandwidth) into cycles. A weight is assigned to every periodic task to show the maximum of time during which every task takes service in a cycle. The amount of weights is equal or less than cycle duration. So, every task is served in round-robin way during a cycle. It means that the hardware scans every segment of the disk and serves the asking of the tasks.
There are two tasks on I/O disc used commonly, tasks based on rate and those based on deadline. Tasks based on rate require a minimal data rate to fulfill limitations in real time. Whereas those based on deadline, are presented from bytes number to be transferred and their deadline. The hardware passes every byte to the controller that from his way processed them to the next step of the elaboration technique.
Figure 2 illustrates the architecture of the system WRR-SCAN. As we can see in the figure the two parts are separated and have different functions but they interwork for the same purpose, scanning and transmitting data. Each data is elaborated in the sequences way, concretely it goes in the operating system and makes a way back throw WRR-SCAN to the controller and than to the hardware. We can see the task separated in sequences of periodic time t,

ti+1, tn. And we can make an equation Ti= Qi, Di, Li This equation tells us that the algorithm time to finish the work and to back from the beginning to the end is more efficient than Scan/Edf. Divided in three element it combines the time that the period and the minimum byte need for a cycle from the beginning and need to finish the work. This time is better than the EDF algorithm and it gives a better performance in the work of data elaboration. In the figure below, this equation is better illustrated and gives us the exact idea of how the algorithm works. A simple allocation scene found for this is the reservation of a minimal memory space for this system activity

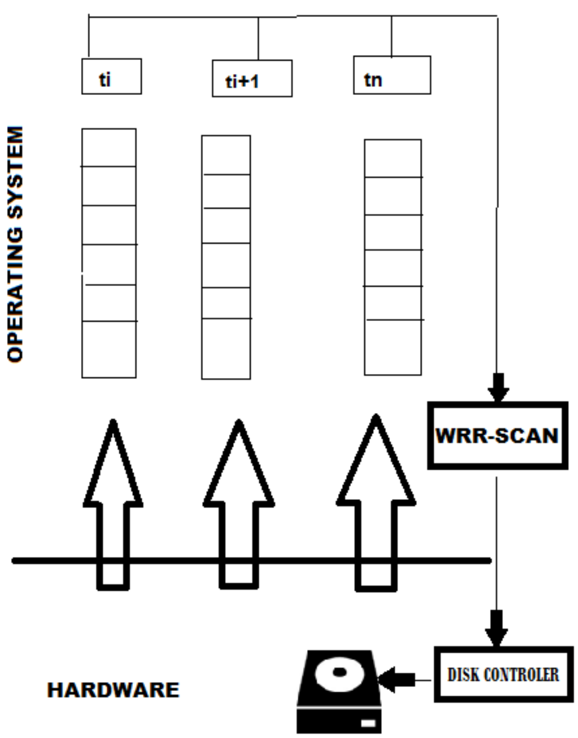

Fig. 2: The architecture of WRR-SCAN

and the allocation of the left space for the activities I/O based on the rate.

The algorithm WRR-SCAN profits from this similar period length, in which every task based on the rate allocates at the buffer in concordance with its minimal data rates. According to the figure, we see the hardware represented by the HDD controlled by the controller and forwarded in the operated system by the WRR algorithm.

This way the time for the packets to get in or be scanned and elaborated, the packets one enters from the scan of the controller thought the algorithm in a sequential way in the system this time depends of the capacity and ability of the HDD. We can give a round R length per rotation that has a place in the big equation give below. It must be smaller than the sum of the time needed per period on the hardware.

Then every real-time disc is associated with a weight Wi, which restricts the quantity of transmitted disc time in which the head serves to this disc in one only round.

Let's sign D the one that restricts the disc rotation time. The eq.(1) below is true when there are no non-periodic tasks:

$$Ti + D + \sum_{i=1}^{n} W_i \leq R, \qquad (1)$$

The schedule algorithm WRR-SCAN works according to the steps below:

First of the entire sum from n to a number i must be real in a real system trying to give better performance in time and capability. Based on all real time task periods defines the adequate value of R. Then calculates the number of times in which the task has been served during a period and the weight Wi. This way it guarantees a minimal data rate specific for the real time tasks and the non-periodic tasks profit an acceptable answer level.

Second, to reduce the bandwidth of the reserve disc in D is applied, a delay that is part of the system also of the HDD represented by the sticker and the time it need to read and decode the info.

The calculation of Wi are conservative and pessimistic, even though they satisfy limitations in real time, they lead in an over-reservation of bandwidth for Ti. The round R length per rotation needs to be bigger than the time and the delay caused by the HDD in the passing of the cycles. There are same specific cases when the capacity C of the elaboration is a problem for the system in this cases the equation has a different calculation. This time, the speed represented by V is in proportional with the work done in the hole cycle of the system. The eq.(2) take the form as follows:

$$W'_i = \frac{C}{V} * \frac{bi}{qi + 1} \qquad (2)$$

3. EXPERIMENTAL RESULTS

Experiments are based in HP845, hardware model of the disc used often in the past days, which is visualized in table 1. In this model we have made an experiment with the parameters given at the table below. As we can see, we have taken a size of the sector 512 byte combined with the capacity of the hardware of this model selected that is 4096 bytes. After that we take a cylinder number approximately 2048 divided in tracks of 22.These are the parameters to be calculated in the time of 9 periods and a speed of 4152rpm.

IJCSI
www.IJCSI.org

Bandwidth is 4 mb/s.The experiment consists in applying the parameters to both of three algorithm and see the differences they give in performance and time. Checking the periodic and non periodic time of the real-system we see same big priorities of WRR-SCAN compared to SCAN-EDF, witch make this model the best experiment viewing this performance.

Table 1: Model HP845

the size of the sector	512 bytes
C	4096 bytes
the cilinder number	2048
track per cilinder	22
E	9
speed	4152 rpm
V	4 MB/S

Depending on the number of times the experiment is repeated the I/O parameters change in a faster manner, while the WRR is used. The time which disc stays totally free is expressed in percentage, as we see the difference they are far away from each other. If we can represent those in a graphic way, the curve of the WRR will be far in a linear way up more stable than the other algorithm that goes down in curve of performance. There are other ways to determinate this performance but this is one of the best architectures of hardware in witch the results are clear and can be used in other performances. This was all about the periodic time of use and also for non-periodic with a little difference in calculation. The HP models improve this technique in their platforms in real-systems all around the world and we say that the response is more clear in this algorithm during time.

4. Throughput and response of aperiodic time

Until now, we have seen the performance of both of the algorithms in the periodic way, now we have to determinate how it ranges in the aperiodic time to see the difference of throughput. We will see that in the aperiodic time there won't be a lot of changes. The only thing is that the cycles have a better performance and the delay is not increased then in the periodic throughput. This is also illustrated in the figure 3.
As we can see on figure 3 the response of SCAN-EDF algorithm is more efficient than the other three as in addition the first is the best algorithm to use .In light load, since higher throughput lies disk has enough bandwidth to serve you aperiodic task. However, when the work load increases, the bulk of available bandwidth it handles real-time task and throughput of aperiodic task drops significantly.

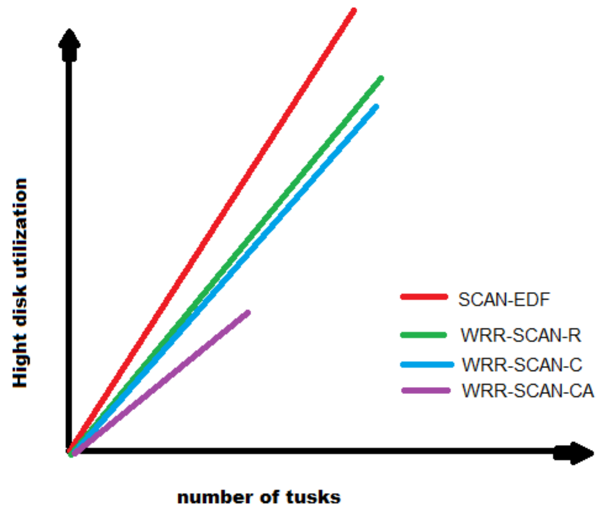

Fig. 3: Disk utilization

In light load, aperiodic time response is a extremely short so it served almost at the moment of its creation. The other time it is as illustrated in the figure divided in colors for being more understandable, the R algorithm is more stable an has a length time constant. Algorithm Wrr-Scan r,c,ca has a response time lower than the other, so a bad efficiency in the number of trucks. The results are evident for real system applications we have to use the first algorithm the other ones have low increase of performance and not applicable in the hardware at aperiodic time. But not only, the number of trucks is lower and it slows down the system hence this is another reason to use the EDF algorithm.

5. Conclusion

The work we have done in this study consist on presenting the ability of WRR-SCAN Weight-Round-Robin-Scan algorithm, that is one of the real time scheduling disc algorithms based on rate, compared to the other algorithms. WRR-SCAN includes WRR in the algorithm of the scheduling disc SCAN for scheduling real-time tasks on aperiodic tasks. First WRR-SCAN separate in rate and cycle the bandwidth of the disk. It shows how the information is being transmitted in single round. WRR-SCAN is better than the other algorithms as it has a better efficiency and low cost of trucks compared with the others.

But, the problem, is that it assumes random locations of the data, and reduces overhead caused by the rotation of the disc by placing the data in the operating system with efficiency, by allowing a cycle of greater duration.
The other algorithm SCAN- EDF SCAN-earliest Deadline First, is a disk scheduling algorithm that is based on priority. The better performance of this one was in the aperiodic time manner where it gave the best time in elaboration of the information. That was demonstrated by the example of HP845 and also this was demonstrated in the graphic way showing the flexibility of this algorithm.
In conclusion, compared with algorithms that are based on priority, WRR-SCAN reduces overhead with the best-in, increases disk utilization, and response times and ensures a level that is acceptable to the aperiodic tasks.
The time and the flexibility of this algorithm are better afforded in the real systems and when used in hardware's.

6. REFERENCES

[1] Cheng-Han Tsai, Edward T.-H. Chu, Tai-Yi Huang WRR-SCAN: A Rate-Based Real-Time Disk Scheduling Algorithm, ACM Transactions on Computer Systems, 2004

[2] D. P. Anderson, Y. Osawa, and R. Govindan. A File System for Continuous Media. ACM Transactions on Computer Systems.

[3] S.Y. Amdani* and M.S. Ali**, An Overview of Real-Time Disk Scheduling Algorithms, International Journal on Emerging Technologies 2(1): 126-130(2011)

[4] A. L. N. Reddy and J. Wyllie. Disk Scheduling in a Multimedia I/O System. In Proceedings of the 1st ACM International Conference on Multimedia, pages 225–233, 1993

[5] C. Ruemmler and J. Wilkes. An Introduction to Disk Drive Modeling. IEEE Computer, 27(3):17–28, March 1994.

[6] P. V. Rangan and H. M. Vin. Designing File Systems for Digital Video and Audio. In ACM Symposium on Operating Systems Principles, pages 81–94, October 1991

[7] W. G. Aref, I. Kamel, and S. Ghandeharizadeh. Disk Scheduling in Video Editing Systems. IEEE Transactions on Knowledge and Data Engineering, 13(6):933–950, November 2001.

[8] Carl Staelin, Gidi Amir, David Ben-Ovadia, Ram Dagan, Michael Melamed, Dave Staas, Real- time disk scheduling with concurrent I/O request, HP Laboratories HPL-2011-11

[9] Arezou Mohammadi and Selim G. Akl "Scheduling Algorithms for Real-Time Systems" Technical Report No. 2005-499,This work was supported by the Natural Sciences and Engineering Research Council of Canada, 2005.

[10] Chia-Hui Wang1, Jan-Ming Ho, A Feedback-Controlled EDF Scheduling Algorithm for Realtime Multimedia Transmission, Technical Report No. 2005-499

[11] C. M. Krishina , Kang G. Shin, "Real-Time Systems", 1997.

[12] Binny S.Gill and Dharmenda S.Modha, WOW: Wise Ordering for Writes-Combining Spatial and Temporal Locality in Non-Volatile-Memory, IBM Research Center, 2004

[13] Nianmin YAO_, Jinzhong CHEN, Ang LI, An Inter-group Seek-optimizing Disk Scheduling for Real-time System, Journal of Computational Information Systems 8: 18 (2012) 7579-7586

[14] Ray-I Chang, Wei-Kuan Shih, "Real Time Disk Scheduling for Multimedia Application with Deadline Modification Scan Scheme", pp.149-168, 2000

Julian Fejzaj in 2003 has finished studies at Faculty of Natural Sciences in Tirana University. At 2014 he finished his PhD Studies on 'Service Oriented Database Architectures'. Currently is working in Department of Informatics as lecturer. His fields of study include database services and architecture, computer network and architecture organization. He has published several research papers in the International conferences and journals.

Aldo Frasheri has completed master's degree in Informatics Engineering Department during 2014 academic year. His fields of study include computer and communication, network protocols and configuration.