International Journal of Technical Research and Applications e-ISSN: 2320-8163,
www.ijtra.com Special Issue 31(September, 2015), PP. 62-66

OVERVIEW OF DISK SCHEDULING ALGORITHM IN OVERLOADED REAL TIME DATABASE SYSTEMS

Vijayshree Shinde[#1], Seema C. Biday [#2]
[#1] Student, Second Year, ME Computers, Terna Engineering College.
[#2] Professor, Computer Department, Terna Engineering College.
Viju0907@gmail.com
scbiday@rediffmail.com

Abstract - Real time disk scheduling plays important role in time critical applications. Conventional database are mainly characterized by their strict data consistency requirements. Database systems for real-time applications must satisfy timing constraints associated with transactions. The numbers of algorithms are proposed to schedule real time transactions in order to produce the overall performance. This paper presents the overview of existing approaches for scheduling the real-time transactions. Earliest Deadline First (EDF) is a basic algorithm which meets the real time constraints, but it gives poor disk throughput. Adaptive Earliest Deadline (AED) improves the performance which uses feedback control mechanism to attain HIT ratio 1.0. Hierarchical Earliest Deadline (HED) maximizes the sum of the values of those transactions that commit by their deadline, and minimizing the number of missed deadlines becomes a secondary concern.

The study investigated performance of EDF, AED algorithms, from which experimental result shows that AED gives better performance under overloaded condition.

Keywords- Real-Time database, EDF, AED, HED, transaction scheduling, deadline

I. INTRODUCTION

Real-time systems are defined as those systems in which the accuracy of the system does not depend only on the logical results of computations but also on the time at which the results are produced [7]. Real-time systems are divided into three types; Hard Real-Time System, Soft Real-Time System and Firm Real-Time System. Hard Real-Time System never allowed to miss a deadline because that can lead to complete failure of the system. In Soft Real-Time System a deadline allowed to be missed, while there is no complete failure of the system it can lead to decreased performance. Firm Real-Time system is more strict than soft real-time system and less strict than hard real-time system. In this system, missing the deadline can lead to decreases the quality of service of the system.

Several previous Real Time Database System studies [1],[3] have addressed the issue of scheduling transactions with the purpose of minimizing the number of miss transactions. A common observation of these studies has been that assigning priorities to transactions according to an Earliest Deadline [2] policy minimizes the number of miss transactions in systems operating under low or moderate levels workload. Haritsa et al points out [5] the need for load control in RTDBMSs. In Earliest Deadline First algorithm, higher priority assign to transactions with earlier deadlines. EDF gives poor performance under overload conditions [9]; due to assigning higher priorities to transactions that are close to missing their deadlines since those transactions delay other transactions that may otherwise be able to meet their deadlines. Haritsa et al proposes the Adaptive Earliest Deadline (AED) priority assignment algorithm [5] for priority assignment as well as for the load control of the system. In this algorithm, transactions are divided into two groups hit group and miss group. AED algorithm used feedback control to dynamically adjust the capacity of hit group to improve the performance of the system under overload condition. Likewise many more algorithms are proposed to overcome the problem of overload condition and increase the system performance.

In real-time disk-based database system, disk I/O requires maximum transaction execution time. Like CPU scheduling, Disk scheduling algorithms that followed timing constraint can significantly improve the real-time performance. Earliest Deadline First and Highest Priority First are the popular real time CPU scheduling algorithms have to be modified before they can be applied to I/O scheduling. The main reason is that disk seeks time, which accounts for a very significant fraction of disk access latency, depends on the disk head movement. The order in which I/O requests are serviced, therefore, has an immense impact on the response time and throughput of the I/O subsystem [6].

II. REAL-TIME SCHEDULING ALGORITHMS

Scheduling algorithms are major part of real-time systems and there exists many different scheduling algorithms due to varying needs and requirements of real time systems.

A. Earliest Deadline First(EDF)

In 1973 Liu and Layland, suggested the most popular real time scheduling algorithms Earliest Deadline First (EDF) [2]. EDF is a dynamic priority algorithm in which task with the earliest deadline has the highest priority. EDF algorithm gives best performance and minimize miss ratio, when systems operating under low or moderate levels of resource and data contention. However, the performance of Earliest Deadline First algorithm is suddenly degrades in an overloaded system. This is because, under heavy loading, transactions gain high priority only when they are close to their deadlines.

Table 1 represents a sample transaction set that will be used as common example throughout this paper to better understand the differences among real time transaction scheduling approaches. Table I consists of different parameters like arrival time Ai, block size Bs, start block Sb, end block Eb, execution time Ei, deadline Di, transfer time Tt. Here service table 2 is calculated to find how much time required for servicing each transaction from T_i to T_n.

TABLE I
PARAMETER OF SAMPLE TRANSACTION SET

Ti	Ai	Bs	Sb	Eb	Ei	Di	Tt
T1	0	40	60	116	60	120	24
T2	0	65	76	93	98	196	39
T3	1	77	93	165	120	241	46.2
T4	1	80	939	1105	120	241	48
T5	1	267	150	243	401	803	160.2
T6	3	300	654	980	450	903	180
T7	4	500	8	904	750	1504	300
T8	4	200	20	203	300	604	120
T9	5	336	236	432	504	1013	201.6
T10	6	185	185	368	278	562	111
T11	7	32	160	183	48	103	19.2
T12	8	20	120	222	30	68	12

TABLE II
SERVICE TABLE OF SAMPLE TRANSACTION SET

C(j,i)	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	i=11	i=12
j=1	0	51	53.1	294.9	170.4	341.4	332.4	148.8	237.6	131.7	32.4	13.2
j=2	33.9	0	46.2	301.8	177.3	348.3	325.5	141.9	244.5	138.6	39.3	20.1
j=3	55.5	65.7	0	280.2	164.7	326.7	347.1	163.5	222.9	117	20.7	25.5
j=4	337.5	347.7	349.8	0	446.7	315.3	629.1	445.5	462.3	387	302.7	307.5
j=5	78.9	89.1	91.2	256.8	0	303.3	370.5	186.9	203.7	128.4	44.1	48.9
j=6	300	310.2	312.3	60.3	409.2	0	591.6	408	424.8	349.5	265.2	270
j=7	277.2	287.4	289.5	58.5	386.4	255	0	385.2	402	326.7	242.4	247.2
j=8	66.9	77.1	79.2	268.8	176.1	315.3	358.5	0	211.5	116.4	32.1	36.9
j=9	135.6	145.8	147.9	200.1	244.8	246.6	427.2	243.6	0	185.1	100.8	105.6
j=10	116.4	126.6	128.7	219.3	225.6	265.8	408	224.4	241.2	0	81.6	86.4
j=11	60.9	71.1	73.2	274.8	170.1	321.3	352.5	168.9	217.5	111.6	0	30.9
j=12	72.6	82.8	84.9	263.1	181.8	309.6	364.2	180.6	205.8	122.1	37.8	0

Figure 1 illustrates the schedule produced by Earliest Deadline First.

EDF scheduled the deadlines by giving priority to earliest deadline first.

T12, T11, T1, T2, T3, T4, T10, T8, T5, T6, T9, T7

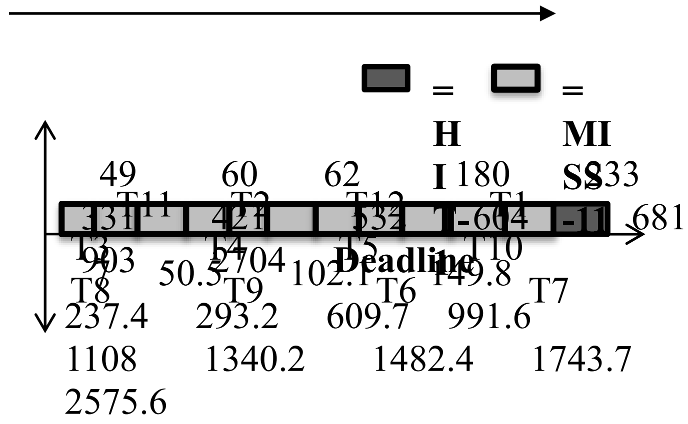

Fig. 1. Schedule produce by Earliest Deadline First algorithm

Figure 1 illustrates the schedule produced by Earliest Deadline First. The execution time for transaction T11 is 50.5 and its deadline is 49, transaction T11 executed after its deadline. Similarly from the timing diagram we can say that the transactions T2, T12, T1, T3, T4, T5, T10, T8, T9, T6 executed after their deadline, so all these transactions are considered as MISS transactions. As clear from the figure 1, the EDF gives worst performance in overloaded condition as only one transaction is HIT.

B. *Adaptive Earliest Deadline (AED)*

The Adaptive Earliest Deadline algorithm is the modified version of the Earliest Deadline First algorithm. The AED as in [5] algorithm uses a feedback control mechanism to estimate the number of transactions that are sustainable under an EDF schedule. In AED algorithm, transaction executing in the system are divided into groups, **HIT group** and **MISS group** as in. whenever transactions are arrive in the system, AED assign a unique integer key to each transaction randomly. Then these transactions are arranged into key ordered list i.e increasing order of key and position of each transaction is noted. If position of transaction is less than or equal to HitCapacity transaction is come under Hit Group otherwise in Miss Group. Here HitCapacity is Dynamic control variable of AED algorithm, which control the load of the Hit Group.

EDF algorithm is used to schedule the transactions in the Hit group and transactions in Miss group are schedule by Random Priority as shown in figure 2. HIT Ratio of a transaction group is fraction of transaction that had completed the execution before its deadline.

After scheduling the transaction next step is to take the information from the system to calculate the new HitCapacity, if new HitCapacity is greater than previous HitCapacity then continue this process until new HitCapacity = priv HitCapacity [5].

International Journal of Technical Research and Applications e-ISSN: 2320-8163,
www.ijtra.com Special Issue 31(September, 2015), PP. 62-66

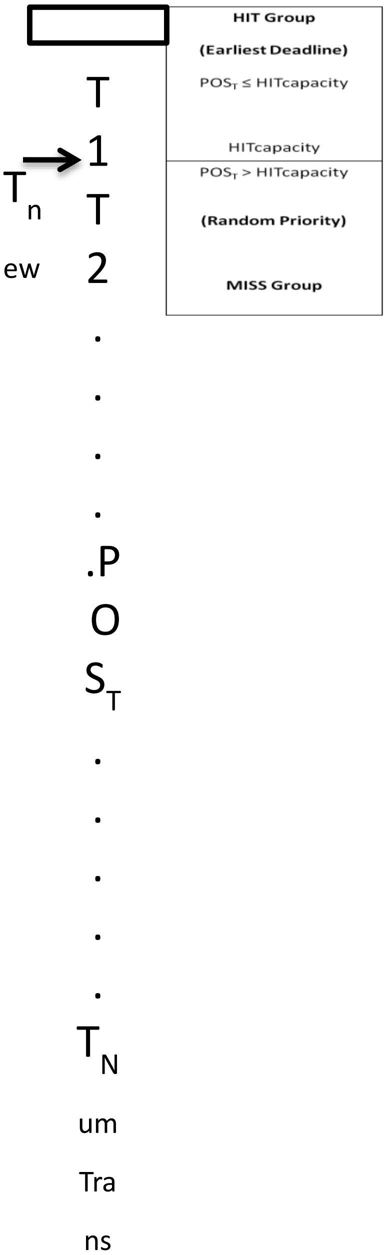

Fig. 2. AED Priority Mapping

Main aim of the AED algorithm is to maximize the number of transaction that can be completed before their deadlines in the HIT group.and all this work is done by controlling the size of Hit group using the control variable that is HitCapacity.

Consider total number of transactions are 12, therefore HITcapacity is 6 (assume)

No. of transactions in HITgroup= 6

No. of transactions in MISSgroup= 6

Order T1 T2 T3 T4 T5 T6 according to Earliest Deadline First we get T2 T1 T3 T4 T5 T6

Order T7 T8 T9 T10 T11 T12 according to Random Priority we get T7 T8 T9 T10 T11 T12

Therefore Schedule become

"T2 T1 T3 T4 T5 T6 T7 T8 T9 T10 T11 T12"

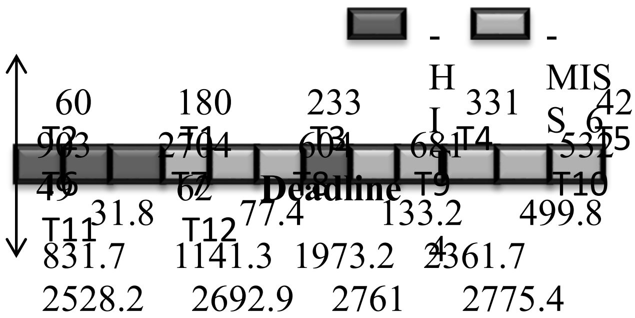

Fig. 3. Schedule produce by Adaptive Earliest Deadline

HITratio(HIT)= 3/6 = 0.55
HITratio(ALL)=4 /12 = 0.33
HITratio(ALL)<0.95 then
HITcapacity= Max (HITcapacity, HitRatio(ALL)* NumTrans *1.25)
HITcapacity=Max (6, 0.33*12*1.25)

Figure 3 illustrates the schedule produced by Adaptive Earliest Deadline First. Form figure it is clear that AED gives better performance than EDF algorithm in overloaded condition.

C. Hierarchical Earliest Deadline(HED)

This algorithm considers the transactions with different values assigned to them. The goal of Hierarchical Earliest Deadline algorithm is to maximize the sum of the values of those transactions that completed before their deadline expired, and minimizing the number of transaction that are completed after their deadline[5]. The main problem when we consider value and deadline of transactions are to construct a priority ordering of transactions. Hierarchical Earliest Deadline (HED) is extension of the AED algorithm. In HED algorithm grouping of transactions are based on their values. According to values of transaction it makes buckets and within each AED-like algorithm is use to determine the relative priority of transactions belonging to the bucket. The priority mapped unit maintains a dynamic list of buckets. When transactions arrived in the system, it assigned to a particular bucket based on its value. Each bucket in the list has minimum value (MinValue) and maximum value (MaxValue) attributes. There are two special buckets, TOP at the head and BOTTOM at the tail of the list. The MinValue and MaxValue attributes of TOP bucket are set to ∞, while the MinValue and MaxValue attributes of BOTTOM bucket are set to zero. When a new transaction, arrives in the system, it check bucket for MinValue ≤ Valuenew ≤ MaxValue. If no such bucket exists, a new bucket is inserted in the list that satisfied the value of transaction. The MinValue of the bucket are computed as (AvgValue / SpreadFactor) whereas MaxValue of bucket are computed as (AvgValue * SpreadFactor), respectively. The SpreadFactor parameter of

HED algorithm controls the maximum spread of values within a bucket. Whenever a transaction enters or leaves the system, the value of AvgValue, MinValue and MaxValue attributes are updated. After creating value based bucket transactions scheduling are performed using AED algorithm. That means within each buckets transactions are scheduled using AED algorithm.

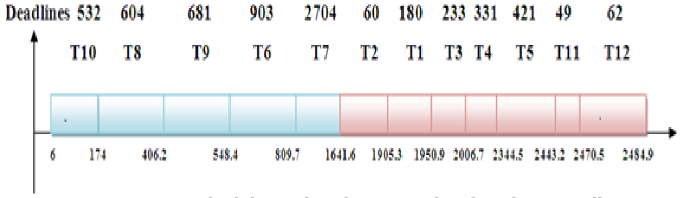

Fig. 4. Schedule produce by Hierarchical Earliest Deadline

In HED algorithm deadlines are consider as values of a transactions. From figure 4 it is clear that HED gives better performance. It behaves like EDF in underloaded condition and AED in overloaded condition.

III. EXPERIMENTAL RESULTS

In order to evaluate the performance of the algorithms, we choose the MISS and HIT as the performance index to analyzed the algorithms. We test Earliest Deadline First, Adaptive Earliest Deadline and Hierarchical Earliest Deadline algorithms in JAVA platform using NetBeans IDE 6.9. MISS and HIT ratioresults of transactions after scheduling with Earliest Deadline First algorithm shown in table III, as load of the system increasing, transaction MISS ratio of EDF algorithm increases significantly, which would brings catastrophic consequences to the system. As can be seen from table IV, transactions scheduled by Adaptive Earliest Deadline algorithm gives better performance even under overloaded condition. This is due to the introduction of feedback control mechanism. It increases the HIT ratio of the transaction by using control variable that is HitCapacity.

Table III and IV shows that overloaded workload condition, is a baffling problem when Earliest Deadline First (EDF) is used to schedule the transactions. As EDF performed worst as compare to all AED real-time scheduling policy in overloaded condition because In EDF priority is assigned to the transaction according to their deadline, although AED improved the performance in the same. Hierarchical Earliest Deadline behaves like EDF in underloaded condition and like AED in overloaded condition.

TABLE III

RESULT OF EARLIEST DEADLINE FIRST

EDF RESULT

SEQ-SEQUENCE TID-TRANSACTION ID AT-ARRIVAL TIME BA-BLOCK TO ACCESS BS-BLOCK SIZE AET-AVERAGE EXECUTION TIME
DL-DEADLINE RT-REALESE TIME IHP-INITIAL HEAD POSITION SBA-START BLOCK TO ACCESS EB-END BLOCK ST-SEEK TIME
TRT-TRANSFER TIME TTT=TOTAL TRANSMISSION TIME CT-COMPLETION TIME TT-TRANSACTION TYPE

SEQ	TID	AT	BA	BS	AET	TT	DL	RT	IHP	SBA	EB	ST	TRT	TTT	CT	STATUS
1.0	2.0	6.0	599.0	5.0	7.5	2.0	119.77905...	6.0	5.0	599.0	603.0	178.2	6.0	184.2	190.2	MISS
2.0	11.0	14.0	122.0	8.0	12.0	2.0	194.96080...	190.2	603.0	122.0	129.0	144.3	9.6	153.9	344.1	MISS
3.0	8.0	1.0	863.0	16.0	24.0	2.0	362.35208...	344.1	129.0	863.0	878.0	220.2	19.2	239.4	583.5	MISS
4.0	13.0	0.0	133.0	25.0	37.5	0.0	563.47806...	583.5	878.0	133.0	157.0	223.5	15.0	238.5	822.0	MISS
5.0	6.0	9.0	30.0	26.0	39.0	0.0	594.68023...	822.0	157.0	30.0	55.0	38.1	15.6	53.7	875.7	MISS
6.0	10.0	19.0	97.0	40.0	60.0	1.0	919.91494...	875.7	55.0	97.0	136.0	12.6	24.0	36.6	912.3	HIT
7.0	9.0	21.0	111.0	47.0	70.5	0.0	1079.4419...	912.3	136.0	111.0	157.0	7.5	28.2	35.7	948.0	HIT
8.0	5.0	13.0	250.0	48.0	72.0	3.0	1094.1042...	948.0	157.0	250.0	297.0	27.9	86.4	114.3	1062.3	HIT
9.0	4.0	15.0	49.0	52.0	78.0	2.0	1185.7783...	1062.3	297.0	49.0	100.0	74.4	62.4	136.8	1199.1	MISS
10.0	15.0	20.0	918.0	56.0	84.0	0.0	1281.3644...	1199.1	100.0	918.0	973.0	245.4	33.6	279.0	1478.1	MISS
11.0	1.0	10.0	355.0	64.0	96.0	1.0	1451.1744...	1478.1	973.0	355.0	418.0	185.4	38.4	223.8	1701.9	MISS
12.0	3.0	16.0	507.0	83.0	124.5	3.0	1884.7457...	1701.9	418.0	507.0	589.0	26.7	149.4	176.1	1878.0	HIT
13.0	7.0	13.0	163.0	87.0	130.5	3.0	1971.5187...	1878.0	589.0	163.0	249.0	127.8	156.6	284.4	2162.4	MISS
14.0	12.0	16.0	857.0	89.0	133.5	0.0	2019.8506...	2162.4	249.0	857.0	945.0	182.4	53.4	235.8	2398.2	MISS
15.0	14.0	8.0	318.0	98.0	147.0	1.0	2214.1524...	2398.2	945.0	318.0	415.0	188.1	58.8	246.9	2645.1	MISS

TABLE IV

RESULT OF ADAPTIVE EARLIEST DEADLINE

AED Table

SEQ-SEQUENCE TID-TRANSACTION ID AT-ARRIVAL TIME BA-BLOCK TO ACCESS BS-BLOCK SIZE AET-AVERAGE EXECUTION TIME
DL-DEADLINE RT-REALESE TIME IHP-INITIAL HEAD POSITION SBA-START BLOCK TO ACCESS EB-END BLOCK ST-SEEK TIME
TRT-TRANSFER TIME TTT=TOTAL TRANSMISSION TIME CT-COMPLETION TIME TT-TRANSACTION TYPE

SEQ	TID	AT	BA	BS	AET	TT	DL	RT	IHP	SBA	EB	ST	TRT	TTT	CT	STATUS
1.0	2.0	6.0	599.0	5.0	7.5	2.0	119.77905...	6.0	5.0	599.0	603.0	178.2	6.0	184.2	190.2	MISS
2.0	6.0	9.0	30.0	26.0	39.0	0.0	594.68023...	190.2	603.0	30.0	55.0	171.9	15.6	187.5	377.7	HIT
3.0	5.0	13.0	250.0	48.0	72.0	3.0	1094.1042...	377.7	55.0	250.0	297.0	58.5	86.4	144.9	522.6	HIT
4.0	4.0	15.0	49.0	52.0	78.0	2.0	1185.7783...	522.6	297.0	49.0	100.0	74.4	62.4	136.8	659.4	HIT
5.0	1.0	10.0	355.0	64.0	96.0	1.0	1451.1744...	659.4	100.0	355.0	418.0	76.5	38.4	114.9	774.3	HIT
6.0	3.0	16.0	507.0	83.0	124.5	3.0	1884.7457...	774.3	418.0	507.0	589.0	26.7	149.4	176.1	950.4	HIT
7.0	7.0	13.0	163.0	87.0	130.5	3.0	1971.5187...	950.4	589.0	163.0	249.0	127.8	156.6	284.4	1234.8	HIT
8.0	8.0	1.0	863.0	16.0	24.0	2.0	362.35208...	1234.8	249.0	863.0	878.0	184.2	19.2	203.4	1438.2	MISS
9.0	9.0	21.0	111.0	47.0	70.5	0.0	1079.4419...	1438.2	878.0	111.0	157.0	230.1	28.2	258.3	1696.5	MISS
10.0	10.0	19.0	97.0	40.0	60.0	1.0	919.91494...	1696.5	157.0	97.0	136.0	18.0	24.0	42.0	1738.5	MISS
11.0	11.0	14.0	122.0	8.0	12.0	2.0	194.96080...	1738.5	136.0	122.0	129.0	4.2	9.6	13.8	1752.3	MISS
12.0	12.0	16.0	857.0	89.0	133.5	0.0	2019.8506...	1752.3	129.0	857.0	945.0	218.4	53.4	271.8	2024.1	MISS
13.0	13.0	0.0	133.0	25.0	37.5	0.0	563.47806...	2024.1	945.0	133.0	157.0	243.6	15.0	258.6	2282.7	MISS
14.0	14.0	8.0	318.0	98.0	147.0	1.0	2214.1524...	2282.7	157.0	318.0	415.0	48.3	58.8	107.1	2389.8	MISS
15.0	15.0	20.0	918.0	56.0	84.0	0.0	1281.3644...	2389.8	415.0	918.0	973.0	150.9	33.6	184.5	2574.3	MISS

Table III and IV shows that overloaded workload condition, is a baffling problem when Earliest Deadline First (EDF) is used to schedule the transactions. As EDF performed worst as compare to all AED real-time scheduling policy in overloaded condition because In EDF priority is assigned to the transaction according to their deadline, although AED improved the performance in the same. Hierarchical Earliest Deadline behaves like EDF in underloaded condition and like AED in overloaded condition.

IV. CONCLUSION

We have analyzed various real time transactions scheduling algorithms like Earliest Deadline First, Adaptive Earliest Deadline and Hierarchical Earliest Deadline. In EDF transactions are ordered according to deadline and the request with earliest deadline is serviced first. Experimental result shows that, EDF gives poor performance in overloaded condition. To avoid this AED algorithm works and performs better than EDF. On hit group it applies EDF and on miss group it applies random priority algorithm which increases the hit ratio 1.0.

In some real-time applications, different transactions may be assigned different values. Setting tradeoff between value and priority is difficult task, it is addressed by HED algorithm where the goal here is to maximize the sum of the values of those transactions that commit by their deadline, and minimizing the number of missed deadlines becomes a secondary concern.

REFERENCE

[1]. R. Abbot and H. Garcia-Molina.”Scheduling Real-Time Transactions”, *SIGMOD Record*, Vol. 17, No. 1, March 1988

[2]. C. Liu and J.W. Layland, “Scheduling Algorithms for Multiprogramming in a Hard Real Time Environment”, J*ournal. ACM*, pp. 104-112, Jan. 1973.

[3]. R. Abbot and H. Garcia- Molina. “Scheduling Real-Time Transactions: A performance Evaluation”, *Proceeding of the 14th VLDB conference, Los Angeles, California*, March 1988.

[4]. Sanjoy K. Baruah and Jayant R. Haritsa, “Scheduling for Overload in Real-Time system”, *IEEE Transaction on Computers*, vol. 46, no. 9, pp.1034-1039, 1997

[5]. Jayant R. Haritsa, MironLivny, Michael J. Carey, “ Earliest Deadline Scheduling for Real-Time Database Systems” National Science foundation under grantIRI-8657323.

[6]. S. Y. Amdani, M. S. Ali,” An Overview of Real-Time Disk Scheduling Algorithms”, *International Journal on Emerging Technologies* vol. 2(1): 126-130, ISSN 0975-8364, 2011

[7]. K. Ramamritham and J. A. Stankovik, “Scheduling Algorithms and operating support for real-time System”, *Proceeding of the IEEE*, vol. 82, January 1994.

[8]. Ben Kao and Hector Garcia-Molina, “ Overview of Real-Time Database System”, Princton University, Princton NJ 08544 USA Standford University, Standford, Ca, 94305,USA(1991).

[9]. S. Y. Amdani, “Comparison of Earliest Deadline with Adaptive Earliest Deadline algorithm”, *International conference on Global Technology Initiative* vol. no. 1 ISBN 978-93-5067-450-5