Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 December 2019 doi:10.20944/preprints201912.0072.v1

Scheduling in Real-time Systems

Mehreen Sirshar
Faculty Of Software Engineering
Fatima Jinnah Women University
Rawalpindi, Pakistan
mehreensirshar@fjwu.edu.pk

Huma Rizwan
Software Engineering
Fatima Jinnah Women University
Rawalpindi, Pakistan
huma.rizwan567@gmail.com

Hajra Nawaz
Software Engineering
Fatima Jinnah Women University
Rawalpindi, Pakistan
hajranawaz555@gmail.com

Muneeba Shabbir
Software Engineering
Fatima Jinnah Women University
Rawalpindi, Pakistan
muneebashabir@gmail.com

Abstract—Scheduling of real time tasks are very important aspect in systems as processes should complete its task at a specific time. There is a need of high energy efficiency and low response time in large data stream so for this energy efficient resources and optimized frameworks are needed. Both hard real time and mixed critically systems are targeted. Soft deadline can be handled while hard deadlines are difficult to cater. Different algorithms are used to schedule tasks like rate monotonic, earliest deadline first, deadline monotonic etc.

Index Terms—Sporadic tasks, fault tolerance, scheduling, real time system, virtualized clouding, petri net, distributive systems

I. Introduction

Real time system should be scheduled carefully as any delay may cause some catastrophic event to occur. There is a need of energy efficient resources and optimized framework for high energy efficiency and low response time. Previous strategies used for scheduling real time tasks are not very efficient and also energy efficiency factor is not considered. All events for the real time system should be predicted before time to meet the desired deadline. Different approaches and models are proposed for energy optimization like;

A. Dynamic voltage frequency scaling (DVFS):

It is used to set speed and power of device to maximize power savings when resources are not required or when processor is idle.

B. Critical path in directed acyclic graph (DAG):

It involves constructing directed acyclic graph in which edges shows activities that must be performed after one milestone shown at the vertices are completed. Edges are weighted with estimated amount of time that activity requires to complete its tasks. The longest path that takes the earliest time to complete is the critical path. [25]

C. Self-suspension

It is also very important phenomena for the fixed priority (FP) and fixed relative deadline (FRD). In real time tasks are suspended by the operating system when they request for some external device like disk if it is busy with some other task at that time. This is self-suspension.

Self-suspension tasks are classified in to model named as;

D. Dynamic self-suspension:

It depend upper bound that tells greatest time for job that cause tasks to dynamically suspend if the limit given is nor crossed.

E. Multi segment self-suspension:

Multi segment characterize execution of tasks by computation segments and suspension intervals. [24]

Mixed critically systems were used to schedule some tasks that can miss their deadlines due to low critically. For hard real time systems if deadlines got missed catastrophic event can occur while soft critically systems deadline can be missed but performance may degrade if too many are missed [23]. There are different approaches and algorithms applied on scheduling of real time systems: Static approach need pre-knowledge about tasks and assign them priority at design time Dynamic approach assigns priority to tasks at run time.

Section II of this paper describes the study that have been done before our research in that real-time scheduling. It basically provides an idea of how different researchers in recent years have done study in this field.

© 2019 by the author(s). Distributed under a Creative Commons CC BY license.

Section III of this paper concerns with Review Process of this paper that includes inclusion criteria, identification of paper and analyzing the area under study.

Section IV of this paper of this paper consists of Research Models that have been categorized according to different realtime systems such as fault tolerant, IoTs, Cloud Computing etc.

Section V of this paper deals with Conclusion and Future Work of real-time scheduling about the ways of improvement

Section VI of this paper includes Appendices

Section VII includes the References

II. Literature Review

Paper in 2014 proposes multiprocessor scheduling algorithm that use low power state of multiprocessor architecture. In this mixed critically systems were targeted.[23] In 2015, Qian, Tao, Frank Mueller, and Yufeng Xin reported Real-time scheduling for distributed systems, implementing earlier deadline first has many challenges.[20] In 2015, paper centered algorithms that are energy efficient and fault tolerant which is EFTR.To evaluate usability of EFTR different experiments were conducted on clouds in platform.[7] In 2015 Awadalla, Medhat Hussein Ahmed. reported that with DVS (dynamic voltage scaling) during scheduling of real time tasks energy consumption can be decreased by taking suitable decision on the speed of processor.[30] In 2015, Hassan, Mohamed, Hiren Patel, and Rodolfo Pellizzoni, reported that a Scheduler along with optimized framework is proposed for mixed-time critical system.[17] In same year, VM (Virtual Machine) scheduling algorithm known as SEATS (smart energy-aware task scheduling) is proposed to allocate resources of VM's effectively so that energy of cloud could be optimized allowing host to consumed energy efficiently. This algorithm helps VM's to reduce total energy consumption along with resolving lengthy tasks at less time [15]

In 2015,the formal SCT (Supervised Control Theory) of TDES is a tool to dynamic reconfigure the non-preemptive scheduling. Real-time system is claimed non-schedulable, so an approach is required for safe execution sequences [18] In 2016 Wu, Jun. reported the energy efficient scheduling of real time task by DVS processors in presence of resource sharing.as tasks are sporadic the may access to shared resources. Two algorithm called EDF and RM are used to schedule the given tasks.[27] In 2016, X. Wang et al. Wonham reported that task are planned,they are accepted by a RTS and completed according to their deadline depending on algorithm applied. [2].

In 2017, R. Devaraj, A. Sarkar and S. Biswas, reported that in same year research was done on modeling of decision system. [10] In 2017 Fan, Ming, Qiushi Han, and Xiaokun Yang. reported that there is increase in transistors density on single processing chip due to increasing development of semiconductor technology. [29] In 2018 Chen, Huangke, et al. reported a research was on shedulability of tasks that are fault tolerant to increase reliability in clouds fault tolerance is too much important issue.[6] In 2018, Guo, Pengze, et al. reported that Existing scheduling approaches for dynamic workflow in cloud environment is inadequate that's why a research on RT task in cloud environment was also done in same year[8].

III. Review Process

A. Inclusion criteria:

The main principle for comprising a research papers in our review is that the research paper describes our research area.It relates that scheduling in real time system are taken to improve the performance of real time system. We also include different survey and review papers to get more knowledge about scheduling and real time systems.

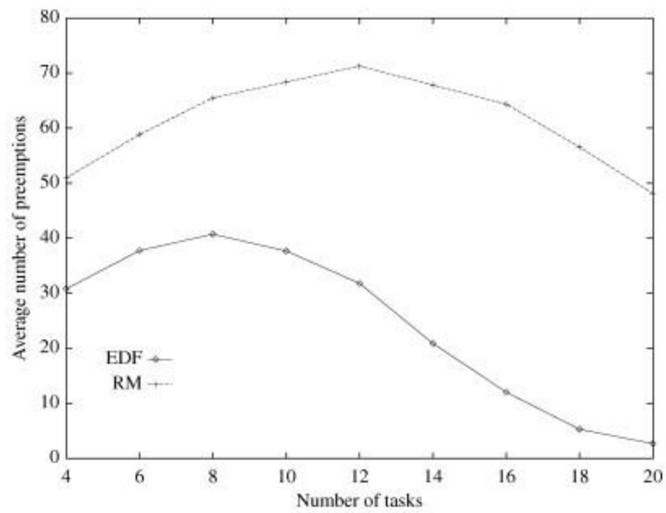

Fig. 1. Preemption introduced by RM and EDF as a function of the number of tasks

B. Identification of paper:

We took papers of different journals. Our all papers are current and mostly of authentic websites like IEEE, ACM, Springerlink, Research gate and Elsevier. We took 30 paper of latest year. We took paper of simple DES techniques, fault tolerance of real time systems, cloud computing, 5G networks and IOT's etc.

C. Analysis:

Schedulability is very important factor in real time systems as tasks are time dependent. Any delay in Schedulability will results in deadline misses. Hard deadlines are difficult to cater

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 December 2019 doi:10.20944/preprints201912.0072.v1

as if they got missed the whole system fails but different is the case with soft deadlines that the user get acknowledged about the deadline miss but the system did not fail. Software relates to the physical world so it should function properly according to timings constraints. Mixed critically system should be targeted to balance between deadline misses and consumption reduction.

1) TASKS IN SCHEDULABILITY: : All events for the real time system should be predicted before time to meet the desired deadline. There are different types of tasks like in
Figure 2

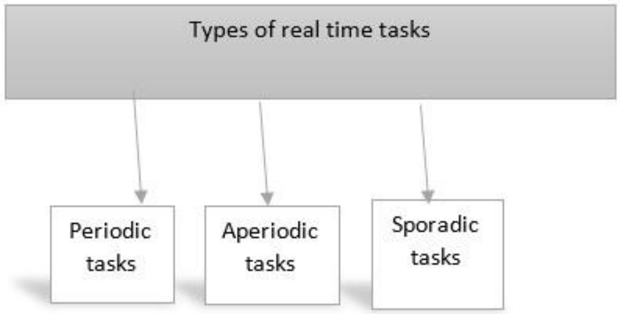

Fig. 2. Types of Real time Tasks

- Periodic tasks that occurs after fixed interval of time and have hard deadlines,
- Sporadic tasks that request at any time so they occur random interval and
- Aperiodic that have irregular arrival time and they are to handle processing requirement of the event. Aperiodic task has no deadline or soft deadlines.[23]

2) Real-time task models: There are various types of task models as in Figure 5 in real-time scheduling systems. Some of them are listed below:

- Release time
  ”It is the earliest time when a job can start execution. in other words it is the time at which job becomes ready for execution.”
- Start time
  Its the same as release time of job.
- Absolute Deadline
  ”This type of deadline of a job is equal to its relative deadline plus its release time.”
- Relative Deadline
  ”The length of time from release time of a job to the instant when it finishes.Maximum allowable response time of a job is called its relative deadline.”

3) APPROACHES: There are different approaches as in Figure 4 and an algorithms is applied on scheduling of real time systems:

- Static approach need pre-knowledge about tasks and assignthem priority at design time
- Dynamic approach assigns priority to tasks at run time.

4) SCHEDULABILITY ALGORITHMS: Optimal scheduling algorithms are applied for scheduling real time system like given below and further description also depicted in Table I

- Rate monotonic called optimal pre-emptive static priority algorithm while EDF is of dynamic priority..
  - These algorithms were complex and have high workloads.
  - RM and EDF are used to schedule sporadic tasks.
- Fuzzy logic approach is also used to reduce energy consumption and its saves energy and creates feasible schedule from real time tasks.

TABLE I
RATE MONOTONIC(RM) VS EARLIEST DEADLINE FIRST(EDF)

Categories	RM	EDF
implementation	multi-level	prior heap
processor utilization	less utilization	full utilization
context switches	at most	at less
guarantee test	less important	simple
predictability test	good	bad

IV. RESEARCH MODELS:

A. Optimize Scheduling in Non-Preemptive Tasks

1) Supervisory Control Theory (SCT): Scheduling of real-time system was done by LL model 1973 in which deadlines are equal to periods. Later NB model that is opposite to LL model.Now a days RTS or EDF is used. DMPR use SCT and TDES.[1] As optimal scheduling of non preemptive task is intractable and expensive online that’s why
SCT and TDES approach use offline formal scheduling.[2] This approach was better than previous all approaches because this model address minimal inter arrival times which is necessary for precise scheduling of sporadic task. This approach was good because this is for preemptive and aperiodic tasks as they have soft deadlines and CPU is allocated preemptively which increase the speed. This research overcome previous problem of lack of predictability and poor resource usage.

2) Synthesis Approach: An aperiodic task with known timelines was developed later, time lines are automated generated. It was a synthesis approach that consider dynamically arriving sporadic task. This approach was best at that time but later a new approach introduces in which find exact completion time of the task and allow processors to fastly change the task until to wait for WCET.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 December 2019 doi:10.20944/preprints201912.0072.v1

This approach has promising results with good accuracy. Previously there was only one method of WCET which was unable to detect exact completion time. this approach shows that scheduler allow reclamation of a higher amount of system resources at runtime.[8]

3) PMC: It is a programmable MC for mixed-time criticality systems that include both soft and hard.. For a realtime periodic task scheduling, TDES (Timed discrete-event system) model is used. So by this, instead of scheduling tasks preemptively, schedule them non-preemptively to enhance efficiency.

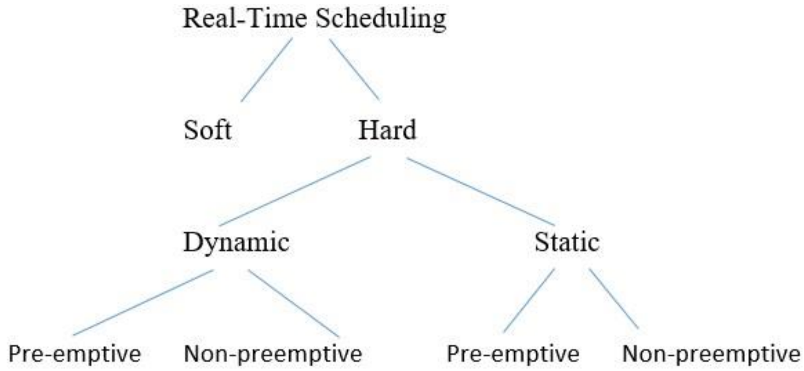

Fig. 3. Taxonomy of Real time Scheduling

B. Scheduling in IoT

With the advancement in Internet of Things (IoT), there is a need for scheduling of realtime systems.

1) Deep Q-Learning model: A deep Q-Learning model for efficient scheduling is proposed and its performance is evaluated with hybrid DVFS (Dynamic Voltage and Frequency Scheduling) scheduling. There is high energy consumption in cloud datacenters and to cater such problem DVFS and machine virtualization are two approaches used.

2) Polynomial Remainder Sequence: PRS is developed that dynamically exploit proactive and reactive scheduling method but this was not accurate for dynamic workflow. In 2017 EDPRS was introduced which shows the issue of predefine time and WCET. It tells us to minimize cost and to improve utilization of resources. IOT's have limitation in power storage .it uses fog network fog is in-between sensors and cloud. In this approach data is save for small period of time in clouds. as clouds are located faraway from sensors therefore delay in data transmission. That's why two approaches introduce which was knapSOS and knapSACK that improve life time of sensor by 5% and energy consumption by 18%. this approach was good because it reduces network traffic.

C. Scheduling in Real-time distributed Systems

1) EDF-os: To achieve schedulability in real time distributed systems, almost any scenario and reduce tardiness bounds is through an algorithm used in sporadic systems called (Earliest Deadline First) EDF-os. Semipartitioned scheduling was proposed for Hard Real Time (HRT) Systems where deadline misses are intolerable.

EDF scheduler utilizes periodic tasks to transmit messages over network to decrease interrupt on system predictability while tasks are in EDF order by scheduler.

D. Scheduling in Cloud Computing

A cloud consists of massive servers having great numbers of CPU cores, bandwidth and memory. Scheduling is a need in cloud systems for managing an urge for competing resources.

1) DRS Algorithm for Scheduling: For that ease Dynamic Resource Scheduler (DRS) is used to process each input efficiently.This Scheduler has overcome many challenges including where to best place resources, measure system overload with minimal overhead and model relationship between resources and performance.

2) DFT Algorithm: it is achieved through algorithm FASTER whose goal is to reduce hardware tolerability increasing cloud performance and efficient cloud utilization.

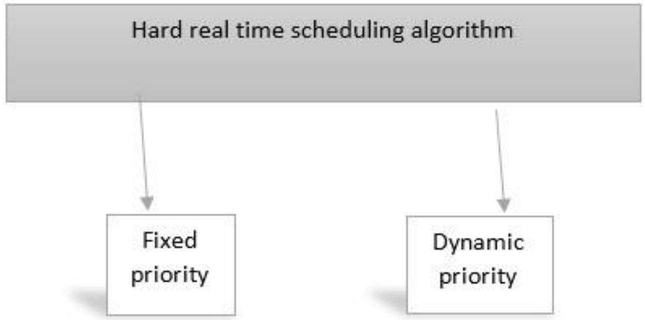

Fig. 4. Hard Real-time Scheduling Model

E. Scheduling in Fault Tolerance

Fault tolerance is capability of computer system that help system that if failure occurs it should be able to operate as intended.Fault tolerance in cloud computing is important.

1) Dynamic Fault Tolerance Algorithm: Dynamic Fault Tolerance Algorithm is efficient because it improves schedulability and resource utilization while tolerating hardware failures. It uses PB model for this purpose. It increases utilization of cloud by using overlap technologies. this approach was good because it uses best backup shifting method to use idle resources.

2) Task Classification Method for Scheduling: This approach was limited because no work was done on hybrid real time task in computer. For this purpose, a novel fault tolerance algorithm ARCHER is proposed. It uses task classification method that reduce response time of cloud. This approach was good but it only focuses on inter dependent task. Later in 2018 a research was done that focuses on independent task.

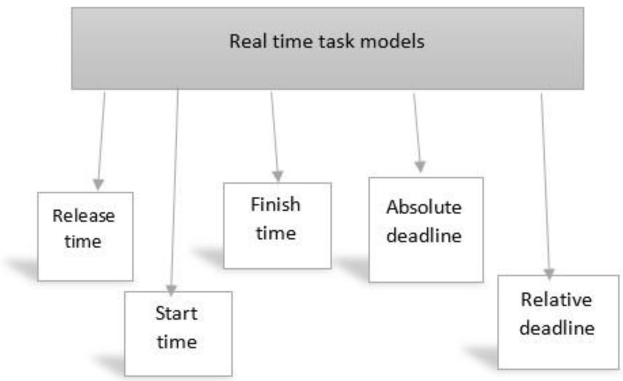

Fig. 5. Real time Task Models

F. EFTR Algorithm

Later in 2018 a research was done that was on independent task. A novel algorithm EFTR adopt a driven tragedy to improve processing of system.this shows good results in planing and conserving energy for tasks.

V. Conclusion and Future Work

Real-time systems constitute alot of part now-a-days in many disciplines. So far, real-time scheduling has been used for only one node and and for planing the processor.This needs to be applied for wider areas and distributed nodes.Especially internet with multiple applications have opened a new area of research in this field.Now a days embedded systems are made real-time but for that they should satisfy all requirements of real-time systems. Thus mathematical techniques and design methodologies have been used for shaping embedded systems in accordance with real-time scheduling.

For Future Work, we will use fault tolerant method and extend our study to tackle multiple hosts failure. To improve the accuracy in scheduling, we will develop a prediction model with feedback. We will also take int account the working of EDF scheduler as it gives better results for real-time scheduling.

VI. Appendix

Appendix A
LIST OF INCLUDED PAPERS

1) Dynamic Multiple-Period Reconfiguration of Real-Time Scheduling Based on Timed DES Supervisory Control
2) Real-time scheduling of non-preemptive sporadic tasks on uni processor systems using Supervisory Control of timed DES
3) Exact Task Completion Time Aware Real-Time Scheduling Based on Supervisory Control Theory of Timed DES
4) Fault-Tolerant Scheduling for Real-Time Scientific Workflows with Elastic Resource Provisioning in Virtualized Clouds
5) Scheduling of fog networks with optimized knapsack by symbiotic organisms search
6) Fault-Tolerant Scheduling for Hybrid Real-Time Tasks Based on CPB Model in Cloud
7) Energy-efficient fault-tolerant scheduling algorithm for real-time tasks in cloud-based 5G network
8) Real-time workflows oriented online scheduling in uncertain cloud environment
9) Towards energy-efficient scheduling for real-time tasks under uncertain cloud computing environment
10) Petri net based decision system modeling in real-time scheduling and control of flexible automotive manufacturing systems
11) DRS: Dynamic Resource Scheduling for Real-Time Analytics over Fast Streams
12) DRS: Auto-scaling for real-time stream analytics
13) Energy-Efficient Scheduling for Real-Time Systems Based on Deep Q-Learning Model
14) Towards energy-efficient scheduling for real-time tasks under uncertain cloud computing environment
15) SEATS: smart energy-aware task scheduling in real-time cloud computing
16) Optimal Semi-Partitioned Scheduling in Soft Real-Time Systems
17) A framework for scheduling DRAM memory accesses for multi-core mixed-time critical systems
18) Dynamic Multiple-Period Reconfiguration of Real-Time Scheduling Based on Timed DES Supervisory Control
19) Fault-tolerant scheduling for real-time scientific workflows with elastic resource provisioning in virtualized clouds

20) Hybrid edf packet scheduling for real-time distributed systems
21) Scheduling real-time communication in IEEE 802.1 Qbv time sensitive networks
22) Global scheduling not required: Simple, near-optimal multiprocessor real-time scheduling with semipartitioned reservations
23) Scheduling algorithms to reduce the static energy consumption of real-time systems
24) Self-suspension real-time tasks under fixed-relativedeadline fixed-priority scheduling
25) Re-Stream: Real-time and energy-efficient resource scheduling in big data stream computing environments
26) Energy-adaptive scheduling of imprecise computation tasks for QoS optimization in real-time MPSoC systems
27) Energy-efficient scheduling of real-time tasks with shared resources
28) Energy minimization for on-line real-time scheduling with reliability awareness
29) Comparison of real time task scheduling algorithms
30) Heuristic Approach for Scheduling Dependent Real-Time Tasks

Appendix B
Scheduling Approaches

1) Fuzzy logic Approach
2) Deep Q-Learning model
3) Polynomial Remainder Sequence (PRS)
4) Earliest Deadline First (EDF-os)
5) Dynamic Resource Scheduler (DRS)
6) FASTER Algorithm
7) novel fault tolerance algorithm ARCHER

VII. References

[1] *X. Wang, Z. Li and W. M. Wonham, "Dynamic Multiple-Period Reconfiguration of Real-Time Scheduling Based on Timed DES Supervisory Control," in IEEE Transactions on Industrial Informatics, vol. 12, no. 1, pp. 101-111, Feb. 2016*

[2] *R. Devaraj, A. Sarkar and S. Biswas, "Real-time scheduling of nonpreemptive sporadic tasks on uniprocessor systems using Supervisory Control of timed DES," 2017 American Control Conference (ACC), Seattle, WA, 2017, pp. 3212-3217.*

[3] *Devaraj, Rajesh, Arnab Sarkar, and Santosh Biswas. "Exact Task Completion Time Aware Real-Time Scheduling Based on Supervisory Control Theory of Timed DES." 2018 European Control Conference (ECC). IEEE, 2018.*

[4] *X. Zhu, J. Wang, H. Guo, D. Zhu, L. T. Yang and L. Liu, "Fault-Tolerant Scheduling for Real-Time Scientific Workflows with Elastic Resource Provisioning in Virtualized Clouds," in IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 12, pp. 3501-3517, 1 Dec. 2016.*

[5] *Rahbari, Dadmehr, and Mohsen Nickray. "Scheduling of fog networks with optimized knapsack by symbiotic organisms search." 2017 21st Conference of Open Innovations Association (FRUCT). IEEE, 2017.*

[6] *H. Han, W. Bao, X. Zhu, X. Feng and W. Zhou, "Fault-Tolerant Scheduling for Hybrid Real-Time Tasks Based on CPB Model in Cloud," in IEEE Access, vol. 6, pp. 18616-18629, 2018. doi: 10.1109/ACCESS.2018.281021*

[7] *Guo, Pengze, et al. "Energy-efficient fault-tolerant scheduling algorithm for real-time tasks in cloud-based 5G networks." IEEE Access 6 (2018): 53671-53683.*

[8] *Chen, Huangke, et al. "Real-time workflows oriented online scheduling in uncertain cloud environment." The Journal of Supercomputing 73.11 (2017): 4906-4922.*

[9] *Chen, Huangke, et al. "Towards energy-efficient scheduling for real-time tasks under uncertain cloud computing environment." Journal of Systems and Software 99 (2015): 20-35.*

[10] *Bas¸ak, Ozkan, and Y. Esra Albayrak. "Petri net based decision system¨ modeling in real-time scheduling and control of flexible automotive manufacturing systems." Computers Industrial Engineering 86 (2015): 116-126.*

[11] *T. Z. J. Fu, J. Ding, R. T. B. Ma, M. Winslett, Y. Yang and Z. Zhang, "DRS: Dynamic Resource Scheduling for Real-Time Analytics over Fast Streams," 2015 IEEE 35th International Conference on Distributed Computing Systems, Columbus, OH, 2015, pp. 411-420.* [12] *Fu, Tom ZJ, et al. "DRS: Auto-scaling for real-time stream analytics." IEEE/ACM Transactions on Networking 25.6 (2017): 3338-3352.*

[13] *Zhang, Qingchen, et al. "Energy-efficient scheduling for real-time systems based on deep Q-learning model." IEEE Transactions on Sustainable Computing 4.1 (2017): 132-141.*

[14] *Chen, Huangke, et al. "Towards energy-efficient scheduling for real-time tasks under uncertain cloud computing environment." Journal of Systems and Software 99 (2015): 20-35.*

[15] *Hosseinimotlagh, Seyedmehdi, Farshad Khunjush, and Rashidaldin Samadzadeh. "Seats: smart energy-aware task scheduling in real-time cloud computing." The Journal of Supercomputing 71.1 (2015): 45-66.*

[16] *Anderson, James H., et al. "Optimal semi-partitioned scheduling in soft real-time systems." Journal of Signal Processing Systems 84.1 (2016): 323.*

[17] *Hassan, Mohamed, Hiren Patel, and Rodolfo Pellizzoni. "A framework for scheduling DRAM memory accesses for multi-core mixed-time critical systems." 21st IEEE Real-Time and Embedded Technology and Applications Symposium. IEEE, 2015.*

[18] *Wang, Xi, ZhiWu Li, and W. M. Wonham. "Dynamic multiple-period reconfiguration of real-time scheduling based on timed DES supervisory control." IEEE Transactions on Industrial Informatics 12.1 (2015): 101111.*

[19] *Zhu, Xiaomin, et al. "Fault-tolerant scheduling for real-time scientific workflows with elastic resource provisioning in virtualized clouds." IEEE Transactions on Parallel and Distributed Systems 27.12 (2016): 35013517.*

[20] *Qian, Tao, Frank Mueller, and Yufeng Xin. "Hybrid edf packet scheduling for real-time distributed systems." 2015 27th Euromicro Conference on Real-Time Systems. IEEE, 2015.*

[21] *Craciunas, Silviu S., et al. "Scheduling real-time communication in IEEE 802.1 Qbv time sensitive networks." Proceedings of the 24th International Conference on Real-Time Networks and Systems. ACM, 2016.*

[22] *Brandenburg, Bjorn B., and Mahircan G¨ ul. "Global scheduling not¨ required: Simple, near-optimal multiprocessor real-time scheduling with semi-partitioned reservations." 2016 IEEE Real-Time Systems Symposium (RTSS). IEEE, 2016.*

[23] *Legout, Vincent, Mathieu Jan, and Laurent Pautet. "Scheduling algorithms to reduce the static energy consumption of real-time systems." RealTime Systems 51.2 (2015): 153-191.*

[24] *Huang, Wen-Hung, and Jian-Jia Chen. "Self-suspension real-time tasks under fixed-relative-deadline fixed-priority scheduling." 2016 Design, Automation Test in Europe Conference Exhibition (DATE). IEEE, 2016.*

[25] *Sun, Dawei, et al. "Re-Stream: Real-time and energy-efficient resource scheduling in big data stream computing environments." Information Sciences 319 (2015): 92-112.*

[26] *Zhou, Junlong, et al. "Energy-adaptive scheduling of imprecise computation tasks for QoS optimization in real-time MPSoC systems." Proceedings of the conference on design, automation test in Europe. European Design and Automation Association, 2017.*

[27] *Wu, Jun. "Energy-efficient scheduling of real-time tasks with shared resources." Future Generation Computer Systems 56 (2016): 179-191.*

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 December 2019 doi:10.20944/preprints201912.0072.v1

[28] Fan, Ming, Qiushi Han, and Xiaokun Yang. "Energy minimization for on-line real-time scheduling with reliability awareness." Journal of Systems and Software 127 (2017): 168-176.

[29] Shinde, Vijayshree, and Seema C. Biday. "Comparison of real time task scheduling algorithms." Int. J. Comput. Appl 158.6 (2017): 37-41.

[30] Awadalla, Medhat Hussein Ahmed. "Heuristic Approach for Scheduling Dependent Real-Time Tasks." Bulletin of Electrical Engineering and Informatics 4.3 (2015): 217-230.