A Survey on Schedulability Analysis of Rate-Adaptive Tasks

Prashant Giridhar Shambharkar
Department of Computer Engineering Delhi Technological University
prashant.shambharkar@dtu.ac.in

Siddhant Bhambri
Department of Computer Engineering Delhi Technological University
siddhant_bt2k16@dtu.ac.in

Arnav Goel
Department of Computer Engineering Delhi Technological University
arnav_bt2k16@dtu.ac.in

M.N. Doja
Department of Computer Engineering, Jamia Millia Islamia New Delhi, India
mdoja@jmi.ac.in

***Abstract*— In automotive real-time systems applications, Multiprocessor Control Units (MCU) handle the engine particular tasks which depend on the specific crankshaft rotation angles and are also a function of the instantaneous angular velocity of the engine. The timing behavior of Adaptive Variable Rate (AVR) tasks has been analyzed in several papers, whose results allow verifying the behavior of engine control applications under different sets of assumptions. This survey paper highlights the major scheduling policies that have been studied in past. In particular, scheduling of such tasks at an arbitrary mode is derived for static as well as dynamic state of the engine.**

***Keywords*— adaptive variable tasks, schedulability, crankshaft**

I. Introduction

Real-time systems play a crucial role in engine control applications which involve scheduling of adaptive variable-rate tasks. Before a comprehensive theory was available, the rate-adaptive tasks were modelled on the lines of elastic scheduling. However, the limitation of the earlier approaches in scheduling of fuel injection systems has led to the need for new types of task models such as fixed priority scheduling. Effective scheduling of these tasks leads to improvement in the computational parameters such as engine output torque, air-fuel ratio and engine efficiency.

The direct relationship between AVR tasks and the rotation of the crankshaft inside the engine of an automobile, and their handling, is presented as a problem which has been discussed and taken forward by a lot of researchers involved in this field. The paper also provides an overview of the major scheduling policies that have been studied in the past in lieu of rate variable angular tasks.

The study of the working mechanism of crankshaft is of concern and thus, has been covered under this paper in brief. The literature survey in the end consists of intensive comparative studies made on the scheduling policies and also on the work done in the past with respect to AVR tasks.

The paper is divided into six sections. Section I contains the introduction; Section II illustrates the Crankshaft working Mechanism. The Adaptive Variable Rate tasks are explained in Section III. Dynamic Priority Based Scheduling is presented in Section IV.Literature Survey is illustrated in Section V and the paper is concluded with the discussion presented in Section VI.

II. Crankshaft Working Mechanism

Inside a fuel-injection system [1], the primary objective is to know the point(s) in time and quantity of fuel that needs to be injected in the engine cylinders, which is relative to the position of each piston. The angular position of the crankshaft defines each piston's position. When the dead center is farthest from the crankshaft, it is known as the top dead center (TDC). When this dead center is nearest to the crankshaft, it is called the bottom dead center (BDC). Table 1 lists down the important and frequently used notations in the paper.

TABLE I. Defined Notations and their Descriptions

Symbol	Description
R_i	A r_i periodic task is a software entity defined by set of parameters to be executed in a defined job and mode.
ω_i^m	Instantaneous speed for mode m and i^{th} task.
D	A mode change deadline is a temporal time constraint (maximum) for the job activation of the task.
C_i	Worst Case execution time is longest time executed in job of i^{th} task as function of instantaneous speed.
T_i	Time period between j^{th} and $(j+1)^{th}$ jobs instances or two adjacent mode change phases.
U_i	Utilization factor is the ratio of WCET/Time Period of the i^{th} task i.e. C_i/T_i
$U_n = \sum U_i$	Total processor utilization factor is the fraction of processor time utilized by the periodic task set.

This is depicted in Figure 1. The first case is of an engine consisting of four cylinders in which the pistons are coupled in opposite phases, such that, when two of them are in a top-dead center, the others are in a bottom-dead center. The top-dead center is generally considered to be the reference point. Also, it is crucial to check if the combustion process occurred properly or not. The implementation of these tasks thus depends on the engine structure.

The tasks may be triggered at each top-dead center i.e. either twice during every crankshaft rotation which is termed as pseudo-cycle or more frequently, which is called half-top-dead center. At the TDC, the time between two activations is neither fixed nor varying, which is a problem. They depend on the engine rotation speeds which are varying. The variation takes place within given ranges in a defined interval of maximum acceleration.

III. AVR TASKS

Tasks are activated using different mechanisms in automotive applications [2]. AVR tasks are one of them and these are directly linked to the crankshaft rotation and get activated at specific rotation angles. Thus, they are also termed as angular tasks [3]. The tasks' activation rate is proportional to the engine's speed. The problem associated with this application is that higher the speeds of the engine, more will be the increase in the system utilization beyond a pre-defined limiting value which may further generate a condition of overloading on the Electronic Control Unit's (ECU) processor which will be executing the application. Such type of an overload may cause disruption to the controlled system. One possibility is the introduction of unbounded delays or may even result into a complete loss of functionality [4].

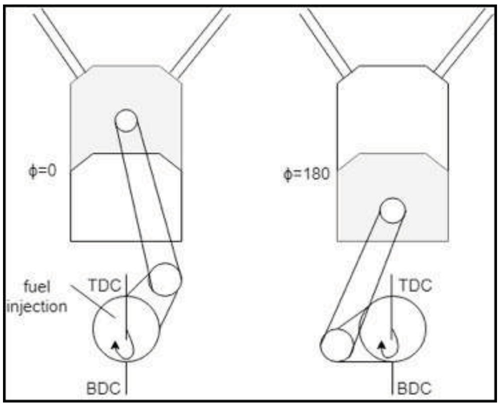

Fig. 1. Different angular phases in the rotation of the crankshaft [1].

It is thus important to change the behavior of angular tasks [5]. These tasks are executed as temporal constrained jobs within an interval of execution modes that work within a pre-defined interval of angular speeds. They also have a self-adaptive behavior which is implemented through a set of mode change.

IV. FIXED/DYNAMIC PRIORITY BASED SCHEDULING

The classical approaches based on processes such as Elastic/timeline scheduling [6], [7] although driven a synchronized method of analysis but failed to introduce significant validation for runtime overhead, jitter in task scheduling. Considerably such problems were solved using more reformed priority-based approach [8].

For a defined set of modes and jobs, priority-assignment is done for each task and a schedule is generated according to the current priority value [9]. Liu and Layland (1973) [10], were among the first to analyse the properties and parameters of the two primary priority assignment policies:

A. Rate Monotonic Scheduling Policy

- Under the Rate Monotonic policy, fixed priority assignment is done for the tasks based on the length of time periods. In simple words, the task with the minimum period is assigned the highest priority and then this task is executed in the given job sequence.
- It is the most commonly used algorithm for assigning priorities in engine controlled real-time application due to its easiness and feasibility in implementation.

B. Earliest Deadline First Scheduling Policy [11]

- It's a dynamic priority scheduler assignment. Dynamic assignment of priorities is done and the priority is inversely proportional to the absolute deadline of the active jobs.
- Only optimal for the jobs which are scheduled else it degrades for overloaded runtime overhead.

Figure 2 represents the Gantt chart when drawn using the rate monotonic and earliest deadline first scheduling algorithms.

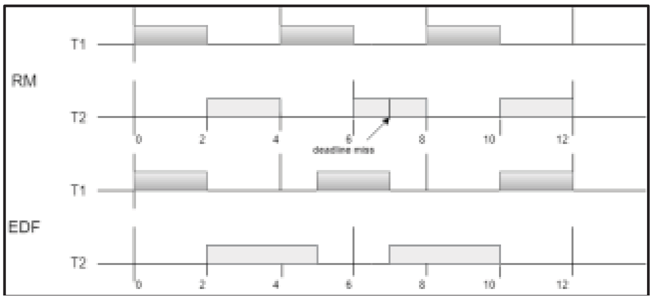

Fig. 2. Tasks scheduled by RM and EDF for U_p=1 [3]

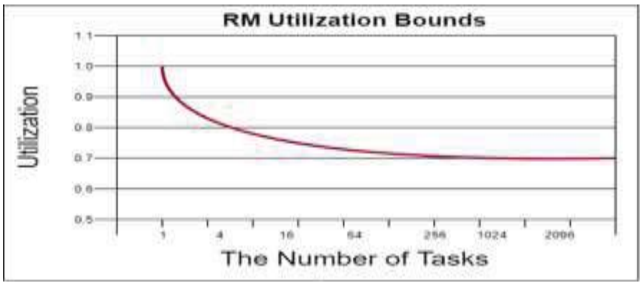

Fig. 3. Utilization vs number of tasks 'n' curve of system processor scheduled under rate monotonic policy.

The proposed mechanism by Liu and Layland were derived on a defined set γ of n periodic task under the assumption that all tasks begin simultaneously at say time t=0.the tasks were independent to each other and didn't imposed on each other. A set of tasks can be scheduled by applying Rate Monotonic algorithm if the following condition holds [12]:

$$\sum_{k=1}^{n} U_i \leq n(2^{1/n} - 1) \quad (1)$$

$$\lim_{n\to\infty} n\left(2^{1/n} - 1\right) = \ln 2 = 0.69 \quad (2)$$

Hence for n tasks to be scheduled by Rate Monotonic algorithm policy the maximum no. of tasks that could be scheduled if $U \leq 0.69$.

V. Literature Survey

A comparative study of the execution of the engine controlled angular (AVR) tasks is presented with respect to-

1) Table III illustrates about the past literature with respect to analysis of AVR tasks.

2) Table II illustrates about different modes of scheduling used for analysis of AVR tasks.

TABLE I: Different Modes of Scheduling used for Analysis of AVR Tasks

ALGORITHM	ASSUMPTIONS	ADVANTAGE	DISADVANTAGE
Rate Monotonic Policy	It is implemented using (ILP) method. Critical or quantized speeds were only taken into consideration.	It is a commonly used algorithm. It is simple to implement, by defining set of execution modes (ranges) in reference to angular variable speeds.	At high engine speeds, this algorithm could introduce large delays and eventually runtime overhead.
Earliest Deadline First Policy	Period of each task changes continuously in an arbitrary range. Dynamic state of engine is considered.	Under exact analysis and arbitrary state variables, linear complexity for AVR task is procured.	Uncertainty in the occurrence of task from same rotation source.
Elastic Scheduling Policy	Activation rate of AVR task is assumed constant. Tasks are assumed as elastic springs with constant elastic coefficients.	Computation time is fixed. Execution period of task could be changed elastically within a given range.	Overload condition by the task is not controlled independently but by global processor scheduler. This reduces the utilization of other task by increasing the execution time.

TABLE II: Past Literature with Respect to Analysis of AVR Tasks and Task Scheduling

REFERENCE PAPER	KEY HIGHLIGHT	DESCRIPTION
[1]	Analysis of fuel-injection	Knowledge of point(s) in time and quantity of fuel that is to be injected in cylinders which is relative to the position of each piston.
[2]	Task activation	In automotive applications, different mechanisms are presented to activate AVR tasks.
[3]	Angular tasks	As tasks are dependent on crankshaft rotation angles, they are specifically termed as angular tasks.
[4]	Introduction of unbounded delays	Possible solution presented to prevent overloading on ECU but may also result into a complete loss of functionality.
[5]	Behavioral change of angular tasks	Tasks are executed as temporal constrained jobs along with having self-adaptive behavior.
[6], [7]	Analysis of elastic/timeline scheduling	Scheduling is represented using synchronized method of analysis but failed to justify significant runtime overhead.
[8], [9]	Priority-based approach	Possible solution presented to prevent runtime overhead in job scheduling. Priorities are assigned and schedule is generated according to current priority value.
[10], [11]	Analysis of scheduling policies	Liu and Layland analyzed parameters for rate monotonic and earliest deadline first scheduling policy.
[12]	Analysis of rate monotonic scheduling policy	Conditions are presented for scheduling tasks based on rate monotonic scheduling policy based on the value of utilization factor.
[13]	Analysis of AVR task problems with respect to different engine speeds.	Automotive applications of AVR tasks presented to adapt the requirements for computation of tasks controlled by engine and their functionalities.
[14]	Analysis of a single AVR task.	Preliminary schedulability results derived and their model presents analysis of a single AVR task based on Rate Monotonic Scheduling policy.
[15]	Analysis of angular tasks based on their schedulability.	The focus has been on Earliest Deadline First (EDF) for the schedulability analysis.
[16]	Quantizing instantaneous rotation speed of the crankshaft.	Integer Linear Programming (ILP) model is discussed.
[17]	Triggering of worst-case interference by rate variable tasks.	A number of simulation experiments are shown and compared with the suggested analysis based on Integer Programming (ILP) model. This analysis computes the worst-case interference processed by non-periodic tasks in dynamic conditions under different fixed priorities processor.
[18], [19], [20]	Mode change analysis.	This analysis is not suitable for describing rate-adaptive tasks, because of their continuously changing activation periods.
[21]	Issues faced in analyzing and designing AVR tasks.	To avoid the processor's overloading, computational requirements are adjusted by the tasks on their own. For each rate-adaptive task, the task utilization should be kept below a required value for which a set of switching speeds is to be determined.

[22]	Validated another way for the exact computation of the worst-case interference of Variable Rate tasks	It further reduced the complexity in the execution of different tasks modes and relaxation in runtime analysis. Assumed the magnitude of the acceleration and deceleration to have same modulus (i.e. $\|\alpha_+\|=\|\alpha_-\|$).
[23]	Formulated tests for the analysis of AVR/Sporadic tasks under Fixed Priority Policy.	Assumed acceleration to be unbounded making any execution mode sequence rational. The schedulability test were validated using ILP and simple linear upper bound method for different execution modes.
[24]	Formulated new Schedulability test of task and multi-modal level systems in polynomial time.	Proved that a Utilization factor of $2 - \sqrt{2} \approx 0.5857$ can be achieved in implicit deadline multi-modal systems for Rate Monotonic Scheduling Policy.
[25]	Analysis of AVR tasks using refined DRT (Digraph Real Time) model.	Formulated a method for the construction of DRT model of a rate variable task with a given sub-ranges of the angular velocity ranges, as a function of control unit (acceleration).
[26]	Introduced sufficient schedulability test	Speed range of rate-dependent task was quantized and maximum response time were determined for each specific angular position.

VI. Conclusion

This paper summaries the major work done in the past on the subject of schedulability analysis of Adaptive Variable Rate Tasks. The study of these scheduling mechanisms is critical as they affect the engine workload management in terms of stability and overloading. The classical approach of elastic and monotonic scheduling laid a foundation for the study of these tasks, though the Rate Monotonic and Earliest Deadline First scheduling policies highlighted the global processor utilisation of such tasks to a greater extent. The in-depth analysis of such tasks is of primary concern when dealing with engine control applications and has contributed to a better simulation on Engine Control Unit (ECU) and its performance.

It has been concluded that relative deadlines of i^{th} task in an arbitrary mode is the most important factor to determine the worst-case utilization U_i which in turn improves the performance of engine-controlled applications, minimizing the risk of runtime overhead, overloading of processor etc. The adoption of schedulability policies for the rate adaptive task under RM and EDF have led for a faster algorithm for controlling crankshaft dominant speeds while avoiding the loss of system functionality. Furthermore, at higher rpm of the shaft, since the activation rate is high and processor usage is maximum, EDF could be a better choice for the scheduling of such tasks as it considers dynamic state of engine as well as sustains a linear complexity of AVR tasks. It also considers the change of priority of each task with respect to their variable deadline.

As for existing advancement in automotive industry, the schedulability test of rate variable tasks can play a vital role in improving fuel efficiency of engine through the fuel injection systems. The fuel injection in Electronic Control Unit (ECU) of an automobile depends on the crankshaft position or their dead centers (BDC and TDC). For the optimal engine performance, the relative deadline, execution and period of a set of tasks must be determined by specific position of crankshaft at an instantaneous speed.

References

[1] Alessandro Biondi, Alessandra Melani, Mauro Marinoni, Marco Di Natale, Giorgio Buttazzo. Exact Interference of Adaptive Variable-Rate Tasks under Fixed-Priority Scheduling. Proceedings of the 26th Euromicro Conference on Real-Time Systems (ECRTS 2014), Madrid, Spain, July 8-11, 2014.

[2] Alessandro Biondi, Marco Di Natale and Giorgio Buttazzo. Response Time Analysis for Real-Time Tasks in Engine Control Applications. Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems, Seattle, Washington — April 14 - 16, 2015.

[3] E. Bini and G. C. Buttazzo. Measuring the performance of schedulability tests. Real-Time Systems, 30(1-2):129–154, 2005.

[4] A. Biondi, A. Melani, M. Marinoni, M. D. Natale, and G. Buttazzo. Exact interference of adaptive variable-rate tasks under fixed-priority scheduling. In Proceedings of the 26th Euromicro Conference on Real-Time Systems (ECRTS 2014), Madrid, Spain, July 8-11, 2014.

[5] Alessandro Biondi and Giorgio Buttazzo. Engine Control: Task Modelling and Analysis. Proceedings of the International Conference on Design, Automation and Test in Europe (DATE 2015), Grenoble, France, March 9-13, 2015.

[6] Giorgio Buttazzo, Luca Abeni. “Adaptive Workload Management through Elastic Scheduling”. Real-Time systems, vol.23, no.1-2, pp.7-24, July-September, 2002.

[7] Giorgio C. Buttazzo, Giuseppe Lipari, and Luca Abeni, “Elastic Task Model for Adaptive Rate Control”, IEEE Real-Time Systems Symposium, Madrid, Spain, December, 1998

[8] Paolo Pazzaglia, Alessandro Biondi, Marco Di Natale and Giorgio Buttazzo, “How Does Task Scheduling Affect Engine Control Performance?” Proceedings of the 7th International Real-Time

Scheduling Open Problems Seminar (RTSOPS 2016), Toulouse, France, July 5, 2016.

[9] Frank Singhoff, Real-Time Scheduling Analysis, 2014.

[10] C. Liu and J. Layland, “Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment.” Journal of ACM, vol. 20, no.1, pp.45-61, Jan 1973.

[11] Zhishan Guo and Sanjoy K. Baruah, “Uniprocessor EDF Scheduling of AVR Task Systems.” Proceeding ICCPS '15 Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems, pp.159-168, April 14-16, 2015.

[12] Giorgio C. Buttazzo, “Rate Monotonic vs. EDF: Judgment Day”, Real-Time Systems, vol. 29, no.1, pp.5-26, Jan. 2005.

[13] D. Buttle. Real-time in the prime-time. Keynote speech given at the 24th Euromicro Conference on Real-Time Systems (ECRTS 2012), Pisa, Italy, July 12th, 2012.

[14] J. Kim, K. Lakshmanan, and R. Rajkumar. Rhythmic tasks: A new task model with continually varying periods for cyber-physical systems. In Proceedings of the Third IEEE/ACM International Conference on Cyber-Physical Systems (ICCPS 2012), pages 28–38, Beijing, China, April 17-19, 2012.

[15] G. Buttazzo, E. Bini, and D. Buttle. Rate-adaptive tasks: Model, analysis, and design issues. In Proceedings of the International Conference on Design, Automation and Test in Europe (DATE 2014), Dresden, Germany, March 24-28, 2014

[16] R. I. Davis, T. Feld, V. Pollex, and F. Slomka. Schedulability tests for tasks with variable rate-dependent behaviour under fixed priority scheduling. In Proceedings 20th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2014), Berlin, Germany, April 15-17, 2014.

[17] A. Biondi, A. Melani, M. Marinoni, M. D. Natale, and G. Buttazzo. Exact interference of adaptive variable-rate tasks under fixed-priority scheduling. In Proceedings of the 26th Euromicro Conference on Real-Time Systems (ECRTS 2014), Madrid, Spain, July 8-11, 2014.

[18] L. Sha, R. Rajkumar, J. P. Lehoczky, and K. Ramamritham, “Mode change protocols for priority-driven pre-emptive scheduling,” Real-Time Systems, vol. 1, no. 3, pp. 243–264, December 1989.

[19] J. Real and A. Crespo, “Mode change protocols for real-time systems: A survey and a new proposal,” Real-Time Systems, vol. 26, no. 2, pp. 161–197, March 2004.

[20] N. Stoimenov, S. Perathoner, and L. Thiele, “Reliable mode changes in real-time systems with fixed priority or EDF scheduling,” in Proceedings of the Design, Automation and Test Conference in Europe (DATE 2009), Nice, France, April 20-24, 2009

[21] Giorgio C. Buttazzo, Enrico Bini, Darren Buttle. Rate-Adaptive Tasks: Model, Analysis, and Design Issues. Proc. of the Int. Conf. on Design, Automation & Test in Europe (DATE 2014), Dresden, Germany, March 24-28, 2014.

[22] Exact Interference of Tasks with Variable Rate-Dependent Behaviour Timo Feld-Frank Slomka - IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems – 2017.

[23] Schedulability tests for tasks with Variable Rate-dependent Behaviour under fixed priority scheduling Robert Davis-Timo Feld-Victor Pollex-Frank Slomka - 2014 IEEE 19th Real-Time and Embedded Technology and Applications Symposium (RTAS) – 2014.

[24] Techniques for Schedulability Analysis in Mode Change Systems under Fixed-Priority Scheduling Wen-Hung Huang-Jian-Jia Chen - 2015 IEEE 21st International Conference on Embedded and Real-Time Computing Systems and Applications - 2015.

[25] Mohaqeqi,M., Abdullah, S.M.J., Ekberg, P., Yi,W: Refinement of workload models for engine controllers by state space partioning. In: Proceedings of ECRTS 2017, Croatia, pp. 11:1-11:22 (2017)

[26] Sufficient Response Time Analysis Considering Dependencies Between Rate-Dependent Tasks Timo Feld-Frank Slomka - Design, Automation & Test in Europe Conference & Exhibition (DATE), 2015 –2016