Optimal Preemptively Scheduling for Real–Time Reconfigurable Uniprocessor Embedded Systems

Hamza Gharsellaoui
INSAT Institute, Tunisia & Al Jouf Technical College, Saudi Arabia

INTRODUCTION

Nowadays, due to the growing class of portable systems, such as personal computing and communication devices, embedded and real-time systems are addressing new criteria such as flexibility and agility (Gharsellaoui et al., 2012). For these reasons, there is a need to develop tools, methodologies in embedded software engineering and reconfigurable embedded control systems as an independent discipline. By response for this requirement of developing reconfigurable systems, many interesting academic and industrial studies have been made in recent years. Feasibility Conditions (FC) for the dimensioning of a real-time system enables a designer to grant that timeliness constraints associated to an application run by the system are always met for all possible configurations. A reconfiguration can be decided either off-line or on-line. In the first case, the goal is to check if several hardware platforms or several hardware configurations can be used to run a specific application while preserving the timeliness constraints of the tasks. In the second case, a reconfiguration might result from a system mode change to adapt the system to the context of its execution or to handle hardware or software faults. We study in this work many dimension sensitivity analysis (more than one task parameter can evolve). A reconfiguration scenario means the addition, removal or update of tasks in order to save the whole system on the occurrence of hardware/software faults, or also to improve its performance when random disturbances happen at run time.

We consider in this article the problem of preemptively scheduling that mixed a workload on uniprocessor to optimize the response time and while ensuring that all off-line requests meet their deadlines and to accept as many sporadic requests which must be guaranteed to meet their deadlines after a reconfiguration scenario by using the earliest deadline first (EDF) scheduling algorithm. Indeed, many real-time systems rely on the EDF scheduling algorithm. This algorithm has been shown to be optimal under many different conditions. For example, for independent, preemptable tasks, on a uniprocessor EDF is optimal in the sense that if any algorithm can find a schedule where all tasks meet their deadlines, then EDF can meet the deadlines (Dertouzos, 1974). Also, Jackson's rule (Jackson, 1955) says that ordering a set of tasks by deadline will minimize the maximum lateness. Further, it has also been shown that EDF is optimal under certain stochastic conditions (Towsley & Panwar, 1991). In spite of these advantageous properties, EDF has one major negative aspect. That is, when using EDF in a dynamic system, if overload occurs, tasks may miss deadlines in an unpredictable manner, and in the worst case, the performance of the system can approach zero effective throughputs (Locke, 1986). The main contribution of this article is the development and the performance evaluation of an optimal version of the EDF algorithm. We propose an intelligent agent-based architecture in which a software agent is deployed to dynamically adapt the system to its environment by applying reconfiguration scenarios.

Before any reconfiguration scenario, the initial real-time embedded control system is assumed to be feasible with a rapid response time. If the system is reconfigured by adding some OS tasks, the processor utilization factor U will be increased and/or some tasks of the new execution model will violate corresponding deadlines. The agent dynamically provides technical solutions for users where the system becomes unfeasible (e.g deadlines are violated), by sending sporadic tasks to idle times, by modifying the deadlines of tasks, the worst case execution times (WCETs), the activation time, by tolerating some non critical tasks m among

n according to the (m,n) firm and a reasonable cost (for more details on (m,n) firm see (Hamdaoui & Ramanathan, 1995), or in the worst case by removing some non hard (soft) tasks according to predefined heuristic. A tool RT-Reconfiguration is developed at INSAT institute in university of Carthage, Tunisia to support all the services offered by the agent. The minimization of the response time is evaluated after each reconfiguration scenario to be offered by the agent.

The organization of this article is as follows. Section 2 presents the related work of the pro- posed approach, gives the basic guarantee algorithm and presents the efficient calculation of system load and the state of the art. In Section 3 we state, some preliminaries concepts, our new contribution and the proposed algorithm for optimal scheduling theory. Section 4 discusses experimental results and the highlights of the proposed approach research. Section 5 summarizes the main results and describes the intended future works.

BACKGROUND

We present related works dealing with reconfigurations and real-time scheduling of embedded systems. Baruah et al. (1991) present a necessary and sufficient feasibility test for synchronous systems with pseudopolynomial complexity. In order to motivate and more easily understand the following definitions, proofs, and subsequent additions to the basic algorithm, we begin by presenting an overview of the dynamic operation of the system. First, we assume that there are n known tasks given by τ_1, τ_2, ..., τ_n where each has a worst case computation time, a deadline relative to when the task is activated, and a specified deadline tolerance indicator which indicates the possibility or not of missing deadline for a given task (may be one or zero) and its value I_i.

- ξ denotes a set of active sporadic tasks τ_i ordered by increasing deadline in a linked list, τ_1 being the task with the shortest absolute deadline.
- a_i denotes the arrival time of task τ_i, i.e., the time at which the task is activated and becomes ready to execute.
- C_i denotes the maximum computation time of task τ_i, i.e., the worst case execution time

(WCET) needed for the processor to execute task $\tau_{i,k}$ without interruption.

- c_i denotes the dynamic computation time of task τ_i, i.e., the remaining worst case execution time needed for the processor, at the current time, to complete task $\tau_{i,k}$ without interruption.
- d_i denotes the absolute deadline of task τ_i, i.e., the time before which the task should complete its execution, without causing any damage to the system.
- D_i denotes the relative deadline of task τ_i, i.e., the time interval between the arrival time and the absolute deadline.
- S_i denotes the first start time of task τ_i, i.e., the time at which task τ_i gains the processor for the first time.
- s_i denotes the last start time of task τ_i, i.e., the last time, before the current time, at which task τ_i gained the processor.
- f_i denotes the estimated finishing time of task τ_i, i.e., the time according to the current schedule at which task τ_i should complete its execution and leave the system.
- L_i denotes the laxity of task τ_i, i.e., the maximum time task τ_i can be delayed before its execution begins.
- R_i denotes the residual time of task Γ_i, i.e., the length of time between the finishing time of τ_i and its absolute deadline.

Notation

- m_i denotes the deadline tolerance of task τ_i, i.e., the task τ_i may executes after its deadline if $m_i = 1$, otherwise it couldn't if $m_i = 0$.
- I_i denotes the task value, i.e, the relative importance of task τ_i with respect to the other tasks in the set. Moreover, we split real-time tasks into two classes: In many applications, there are as well aperiodic as periodic tasks.
- A sporadic task Γ_i (C_i, T_i, D_i) is an infinite collection of jobs having their request times constrained by a minimum inter-arrival time T_i, a WCET C_i and a relative deadline D_i.

Related Content

A Grounded Theory Study of Enterprise Systems Implementation: Lessons Learned from the Irish Health Services
John Loonamand Joe McDonagh (2009). *Information Systems Research Methods, Epistemology, and Applications (pp. 58-72).*
www.irma-international.org/chapter/grounded-theory-study-enterprise-systems/23468

Recognition of Odia Handwritten Digits using Gradient based Feature Extraction Method and Clonal Selection Algorithm
Puspalata Pujariand Babita Majhi (2019). *International Journal of Rough Sets and Data Analysis (pp. 19-33).*
www.irma-international.org/article/recognition-of-odia-handwritten-digits-using-gradient-based-feature-extraction-method-and-clonal-selection-algorithm/233595

Radio Frequency Identification Systems Within a Lean Supply Chain in a Global Environment
Alan D. Smith, Terry Stringer Damron, Susan Cockrelland Amye M. Melton (2018). *Encyclopedia of Information Science and Technology, Fourth Edition (pp. 5516-5526).*
www.irma-international.org/chapter/radio-frequency-identification-systems-within-a-lean-supply-chain-in-a-global-environment/184253

A Comparative Analysis of a Novel Anomaly Detection Algorithm with Neural Networks
Srijan Das, Arpita Dutta, Saurav Sharmaand Sangharatna Godboley (2017). *International Journal of Rough Sets and Data Analysis (pp. 1-16).*
www.irma-international.org/article/a-comparative-analysis-of-a-novel-anomaly-detection-algorithm-with-neural-networks/186855

The Use of Structural Equation Modeling in IS Research: Review and Recommendations
Kun S. Imand Varun Grover (2004). *The Handbook of Information Systems Research (pp. 44-65).*
www.irma-international.org/chapter/use-structural-equation-modeling-research/30342