Chapter 7

Laurent George
University of Paris-Est – Val de Marne, France

Pierre Courbin
LACSC – ECE Paris, France

ABSTRACT
In this chapter the authors focus on the problem of reconfiguring embedded real-time systems. Such reconfiguration can be decided either off-line to determine if a given application can be run on a different platform, while preserving the timeliness constraints imposed by the application, or on-line, where a reconfiguration should be done to adapt the system to the context of execution or to handle hardware or software faults. The task model considered in this chapter is the classical sporadic task model defined by a Worst Case Execution Time (WCET), a minimum inter-arrival time (also denoted the minimum Period) and a late termination deadline. The authors consider two preemptive scheduling strategies: Fixed Priority highest priority first (FP) and Earliest Deadline First (EDF). They propose a sensitivity analysis to handle reconfiguration issues. Sensitivity analysis aims at determining acceptable deviations from the specifications of a problem due to evolutions in system characteristics (reconfiguration or performance tuning). They present a state of the art for sensitivity analysis in the case of WCETs, Periods and Deadlines reconfigurations and study to what extent sensitivity analysis can be used to decide on the possibility of reconfiguring a system.

1. INTRODUCTION
Real-time scheduling has been extensively studied over the last forty years. Feasibility Conditions (FCs) for the dimensioning of a real-time system enable a designer to ensure that timeliness constraints associated to an application run by the system are always met for all possible configurations. The goal of FCs is thus to ensure a deterministic respect of the timeliness constraints.

DOI: 10.4018/978-1-60960-086-0.ch007
Classical feasibility conditions do not consider possible deviations resulting from the reconfiguration of real-time systems. A reconfiguration can be decided either off-line or on-line. In the first case, the goal is to check whether 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.

It could be interesting to study the validity of FCs in the case of such reconfigurations. Sensitivity analysis aims at studying the ability to introduce greater flexibility in the specifications. In this paper, we study only one-dimension sensitivity analysis (one task parameter can evolve, the other task parameters are assumed to be constant).

**Definition 1:** The classical methodology to solve a real-time scheduling problem in the dimensioning phase (more formally described in LeLann (1996)) is as follows:

- (a) Identify the class of scheduling problem to solve, defined by the task model, the timeliness constraints and the scheduling models.
- (b) Identify for this class the possible scenarios of task activation request times.
- (c) Identify from those possible scenarios the subset of scenarios leading to the worst case conditions for the respect of the timeliness constraints.
- (d) Express for this subset the associated Feasibility Conditions (task parameters are used as variables).
- (e) Check that the FCs are met on a targeted architecture (the values of the task parameters are only used at this phase).

This methodology is valid for a given architecture. Sensitivity analysis aims at providing more robust feasibility conditions, valid for a set of hardware platforms or able to tolerate on-line deviations in the specifications of a problem.

In this paper, we consider a preemptive uniprocessor scheduling. We focus on two scheduling policies: Fixed Priority highest priority first (FP) scheduling and Earliest Deadline First scheduling (EDF). The task model is the classical sporadic task model. A sporadic task set \( \tau = \{ \tau_1, \ldots, \tau_n \} \) is composed of \( n \) sporadic tasks, where a sporadic task \( \tau_i \) is defined by:

- \( C_i \): its worst-case execution time (WCET) on a given architecture.
- \( T_i \): its minimum inter-arrival time (also called the minimum period).
- \( D_i \): its relative deadline (any job of task \( \tau_i \) with a request time \( t_i \) must be executed by its absolute deadline \( t_i + D_i \)).

We therefore consider timeliness constraints expressed as late deadlines on the worst case response time of a task, defined as the maximum possible time between the request time of a task and its termination time.

In classical FCs, the task parameters are assumed to be constant for all tasks instances at run time. In the following we discuss the problems that may occur in real systems.

- The WCET depends on the underlying architecture and on the conditions of execution (e.g.: type of memory or cache, frequency of the processor) leading to WCET that depend on the conditions of execution.
- Considering constant task parameters might not be suitable for every application and should be adapted to the situations at run time (e.g. a process should be run more frequently in a given situation to obtain more precision). This has therefore an impact on the periods of the tasks chosen at run time.
- Furthermore, new architectures propose variable speed processors to scale the per-
Related Content

First Steps Towards a Wise Development Environment for Behavioral Models
[www.igi-global.com/article/first-steps-towards-a-wise-development-environment-for-behavioral-models/170517?camid=4v1a](www.igi-global.com/article/first-steps-towards-a-wise-development-environment-for-behavioral-models/170517?camid=4v1a)

Secure Key Generation for Static Visual Watermarking by Machine Learning in Intelligent Systems and Services
[www.igi-global.com/article/secure-key-generation-static-visual/39098?camid=4v1a](www.igi-global.com/article/secure-key-generation-static-visual/39098?camid=4v1a)

Identifying the Opinion Orientation of Online Product Reviews at Feature Level: A Pruning Approach
[www.igi-global.com/article/identifying-the-opinion-orientation-of-online-product-reviews-at-feature-level/199005?camid=4v1a](www.igi-global.com/article/identifying-the-opinion-orientation-of-online-product-reviews-at-feature-level/199005?camid=4v1a)

Combining Static Code Analysis and Machine Learning for Automatic Detection of Security Vulnerabilities in Mobile Apps