Chapter 17
Feasible Automatic Reconfigurations of Real-Time OS Tasks

Hamza Gharsellaoui
National Institute of Applied Sciences and Technology, Tunisia

Atef Gharbi
National Institute of Applied Sciences and Technology, Tunisia

Mohamed Khalgui
Martin Luther University, Germany

Samir Ben Ahmed
National Institute of Applied Sciences and Technology, Tunisia

ABSTRACT

This research deals with reconfigurable uniprocessor embedded real-time systems to be implemented by different OS tasks that should be independent, synchronous, and periodic, and that should meet functional and temporal properties described in user requirements. The authors define two forms of automatic reconfigurations assumed to be applied at run-time: addition-removal of tasks or just modifications of their temporal parameters; WCET and/or periods. The authors define a new semantic of reconfigurations where a crucial criterion to consider is the automatic improvement of the system's feasibility at run-time.

INTRODUCTION

We define an Intelligent Agent that automatically checks the system's feasibility after any reconfiguration scenario to verify if all tasks meet the required deadlines. It provides otherwise, for users to remove some tasks, to change temporal parameters of tasks that violate corresponding constraints by new ones, or to take a hybrid real-time scheduling approach that combines the Rate Monotonic (RM) scheduling algorithm and the Earliest Deadline First (EDF) scheduling algorithm. To handle all possible reconfiguration solutions, the proposed agent-based architecture applies automatic reconfigurations, in order to re-obtain the system's feasibility and to satisfy

DOI: 10.4018/978-1-4666-0294-6.ch017

user requirements. We developed the tool *RT-Reconfiguration* to support these contributions that we apply to a running example system, and apply the Real-Time Simulator, *Cheddar* to check the whole system's behavior. We present in the chapter some simulations of this architecture where we evaluate the agent that we have implemented.

Starting as an aid for industrial embedded applications, Real Time Operating Systems (RTOSs) are now common in a large variety of commercial products. Application areas are, for example telecommunications, automotives, defense industry, medical equipments, and consumer electronics. Common denominators for these embedded systems are real-time constraints. These systems are often safety critical and should immediately react after any environment's evolution. Imagine, for example the airbag of a car not going off instantly as a crash occurs; reaction time delay would be disastrous. Several interesting academic and industrial research works have been made in recent years to develop reconfigurable systems (A.-L. Gehin and M. Staroswiecki, 2008). We distinguish, in these works, two reconfiguration policies: static and dynamic reconfigurations such that static reconfigurations are applied off-line to apply changes before the system cold starts (C. Angelov, K. Sierszecki, and N. Marian, 2005), whereas dynamic reconfigurations are dynamically applied at run-time. Two cases exist in the last policy: manual reconfigurations to be applied by user (M. N. Rooker, C. Sunder, T. Strasser, 2007), and automatic reconfigurations to be automatically applied by Intelligent Agents (M. Khalgui, 2010); (Al-Safi and V. Vyatkin, 2007).

In this book chapter, we are interested in the automatic reconfiguration of embedded real time systems. We define at first time a new semantic of this type of reconfiguration where a crucial criterion to consider is the automatic improvement of the system's feasibility at run-time. We propose thereafter an agent-based architecture to handle all possible reconfiguration scenarios. Nowadays in industry, the new generations of embedded real time systems are addressing new criteria as flexibility and agility. To reduce their cost, these systems should be changed and adapted to their environment without any disturbances. It might be interesting therefore to study the temporal robustness of real-time systems when automatic reconfigurations are dynamically applied at run-time to change parameters of tasks: WCET, deadline and period. This new reconfiguration semantic is considered in our work and we will present its benefits. We suppose in this chapter that a reconfigurable real-time system is implemented by sets of tasks that we assume independent, periodic and synchronous (e.g. they are simultaneously activated at time $t = 0$ time units). We assume also that the deadline of each task is equal to the corresponding period. According to (Liu and Layland, 1973), we characterize each task by a period to be denoted by T equal to the deadline denoted by D and by a Worst Case Execution Time (WCET) denoted by C. We define an automatic reconfiguration as any operation allowing additions-removals or updates of tasks at run-time. Therefore the system's implementation is dynamically changed and should meet all considered deadlines of the current combination of tasks. Nevertheless, when a reconfiguration is applied, the deadlines of new and also old tasks can be violated. We define therefore an agent-based architecture that checks the system's evolution and defines useful solutions when deadlines are not satisfied after each reconfiguration scenario. The Intelligent Agent handles the system's resources in such way that, meeting deadlines is guaranteed. Five cases of suggestions are possible to be provided by the agent: removal of some tasks from the new list, modification of periods (equal to deadlines), modification of worst case execution times of tasks, changing the system's execution model to be assumed as only one task that should support all required functionalities or finally to take a hybrid real-time scheduling

Related Content

Hybrid Algorithms for Manufacturing Rescheduling: Customised vs. Commodity Production
Luisa Huaccho Huatucoand Ani Calinescu (2013). *Industrial Engineering: Concepts, Methodologies, Tools, and Applications (pp. 1488-1516).*
www.irma-international.org/chapter/hybrid-algorithms-manufacturing-rescheduling/69351

Modeling and Simulation of Discrete Event Robotic Systems Using Extended Petri Nets
Gen'ichi Yasuda (2013). *Industrial Engineering: Concepts, Methodologies, Tools, and Applications (pp. 577-593).*
www.irma-international.org/chapter/modeling-simulation-discrete-event-robotic/69304

Fault Tolerant Control of Nonholonomic Mobile Robot Formations
T. Dierks, B. T. Thumatiand S. Jagannathan (2010). *Intelligent Industrial Systems: Modeling, Automation and Adaptive Behavior (pp. 50-83).*
www.irma-international.org/chapter/fault-tolerant-control-nonholonomic-mobile/43629

An Empirical Study to Evaluate the Impact of Demographic Variables to Complaint Behavior of Customers in a Dine-In Restaurant Industry: A Case of Graduate Students
Tiffany Adelaine Gan Tan (2017). *International Journal of Applied Industrial Engineering (pp. 19-32).*
www.irma-international.org/article/an-empirical-study-to-evaluate-the-impact-of-demographic-variables-to-complaint-behavior-of-customers-in-a-dine-in-restaurant-industry/182721

Design and Development of Hybrid Stir Casting Process
Abhishek Kamboj, Sudhir Kumarand Hari Singh (2012). *International Journal of Applied Industrial Engineering (pp. 1-6).*
www.irma-international.org/article/design-and-development-of-hybrid-stir-casting-process/93011