Chapter 2
Scheduling of Aperiodic Tasks on Multicore Systems

Shruti Jadon
Motilal Nehru National Institute of Technology, India

Rama Shankar Yadav
Motilal Nehru National Institute of Technology, India

ABSTRACT

For a hard real-time multicore system, the two important issues that are required to be addressed are feasibility of a task set and balancing of load amongst the cores of the multicore systems. Most of the previous work done considers the scheduling of periodic tasks on a multicore system. This chapter deals with scheduling of aperiodic tasks on a multicore system in a hard real-time environment. In this regard, a multicore total bandwidth server (MTBS) is proposed which schedules the aperiodic tasks with already guaranteed periodic tasks amongst the cores of the multicore processor. The proposed MTBS algorithm works by computing a virtual deadline for every aperiodic task that is arriving to the system. Apart from schedulability of aperiodic tasks, the MTBS approach also focuses on reducing the response time of aperiodic tasks. The simulation studies of MTBS were carried out to find the effectiveness of the proposed approach, and it is also compared with the existing strategies.

INTRODUCTION

Real time systems have been widely used in a variety of devices across a wide range of applications such as mobile phones, electronic game devices, motor vehicles, medical equipments, avionic products, etc. In real time systems, there are two types of tasks or events: periodic events and aperiodic events. The periodic events repeat themselves after a particular interval of time; whereas, the aperiodic events never repeat themselves. The main concern of any real time system is to schedule the periodic as well as aperiodic events with no miss of their deadlines. In this regard, several algorithms had been proposed previously for scheduling of periodic tasks or inclusion of aperiodic tasks together with periodic tasks

DOI: 10.4018/978-1-7998-2491-6.ch002

on single core systems. However, very few research works have been proposed to accommodate the aperiodic tasks together with the periodic tasks for the case of multicore systems.

This chapter is an extension of the previous work reported (Jadon et al. 2018) to accommodate the case of aperiodic tasks when the periodic tasks are already allocated on the cores of the multicore system. In the previous reported work, the periodic tasks for multicore systems were considered and they are applicable for the applications which work only on periodic arrival pattern tasks. However, there are applications that have combination of both periodic as well as aperiodic ones. For example, in an aircraft control system, the instances of parameters like vibration levels on wings, altitude, atmospheric pressure and temperature, engine fuel burn rate and thrust etc. are measured periodically after regular intervals of time. These parameters are checked on the basis of signals accomplished by periodic events. However, if the system signals an air pressure drop, then the jobs that are executed to raise such randomly occurred events are aperiodic jobs. The periodic tasks are scheduled in offline approach using different approaches and algorithms, whereas online approach is required to check the feasibility of aperiodic tasks. The scheduling of aperiodic tasks over periodic ones is the main concern of this chapter.

Based on the above concern, this chapter deals with scheduling of aperiodic tasks that may arrive randomly in the system such that they can be feasibly scheduled on the cores of the system without disturbing the feasibility of already guaranteed periodic tasks on the cores. In this chapter, hard real time system is assumed therefore, the aperiodic tasks with hard deadlines are considered as those were considered like in case of periodic tasks. Thus, it is important for the system to accommodate all periodic as well as aperiodic tasks' within their respective deadlines.

The present chapter deals the case of periodic task set where each task has relative deadline not lesser than its period is considered for inclusion of aperiodic tasks. Further, while dealing with the periodic tasks are scheduled using earliest deadline first approach and the aperiodic tasks are scheduled using total bandwidth server (D. Duy et al. 2017, K. Tanaka 2013, S Kato et al. 2008). The rest of the chapter is organized as follows. The next section provides the study related to the basic understanding of periodic tasks in real time systems. Further sections discuss the preliminaries for this chapter and deals with the scheduling of aperiodic tasks with periodic tasks having relative deadlines not lesser than their respective periods, through total bandwidth server followed by an example to illustrate the effectiveness of proposed multicore total bandwidth server. The simulation experiments for this part of work along with its result analysis are also being carried out in this section. Finally, the chapter concludes. The next section discusses about periodic tasks in real time systems.

PERIODIC TASKS IN REAL TIME SYSTEMS

Real Time Embedded Systems (RTES) have become pervasive and indispensable in our daily life that is, from mobile phones to digital recorder, transportation to industry controls and medical instruments to home appliances, as such these systems affect almost every aspects of our day to day routine. Typical applications of real time embedded systems include smart phones, electronic gaming devices, robotics, space navigation and guidance, weapon monitoring and delivery, aviation and aircraft engine control, nuclear power plant control, medical monitoring and equipments, multimedia systems, anti-lock breaking systems etc. (Liu 2000, Burns 1991, Burns 1995, Haritsa et al. 1993, Kuo et al. 2002, Baruah 1998, Hoyme et al. 1992, Philippou et al. 1998, Buttazzo 2006, Agrawal 2008, Kuo et al. 2017, Macchelli et al. 2002). In Real Time Systems (RTS), the correctness of the system depends not only on the logical