Chapter 13
Formal Analysis of Real-Time Systems

Osman Hasan
National University of Science and Technology (NUST), Pakistan

Sofiène Tahar
Concordia University, Canada

ABSTRACT

Real-time systems usually involve a subtle interaction of a number of distributed components and have a high degree of parallelism, which makes their performance analysis quite complex. Thus, traditional techniques, such as simulation, or state-based formal methods usually fail to produce reasonable results. The main limitation of these approaches may be overcome by conducting the performance analysis of real-time systems using higher-order-logic theorem proving. This chapter is mainly oriented towards this emerging trend and it provides the details about analyzing both functional and performance related properties of real-time systems using a higher-order-logic theorem prover (HOL). For illustration purposes, the Stop-and-Wait protocol, which is a classical example of real-time systems, has been considered as a case-study.

INTRODUCTION

Real-time systems can be characterized as systems for which the correctness of an operation is dependant not only on its functional correctness but also on the time taken. Some commonly used real-time system applications include embedded systems, digital circuits with uncertain delays, communication protocols and dynamic reconfigurable systems.

Until the last decade, real-time systems were analyzed using traditional techniques, such as paper-and-pencil proof methods or simulation. The paper-and-pencil based proof techniques usually have some risk of an erroneous analysis due to the human-error factor. Similarly, accuracy of analysis cannot be guaranteed in computer simulation as well since the fundamental idea in this approach is to approximately answer a query by analyzing a large number of samples. These inaccuracy limitations of paper-and-pencil proof methods and simulation techniques may lead to
disastrous consequences in today’s world, where real-time systems are extensively being used in safety critical and extremely sensitive applications such as medicine, military and transportation. In fact, some unfortunate incidents have already happened in this regard. One of the well-known incidents is the loss, in December 1999, of the Mars Polar Lander; a $165 million NASA spacecraft launched to survey Martian conditions. The Mars Polar Lander is believed to be lost mainly because of its engine shutdown while it was still 40 meters above the Mars surface. The engine shutdown happened due to the vibrations caused by the deployment of the Lander’s legs, i.e., a probabilistic behavior that gave false indication that spacecraft had landed. Some other such incidents related to inaccurate or inadequate analysis of real-time systems include the loss of $125 million Mars Climate Orbiter in 1998 and the performance degradation of Microsoft’s IIS indexing service DLL due to a buffer overflow problem caused by the “Code Red” worm in 2001, which resulted in a loss of over $2 billion to the company. A more recent incident is the faulty operation of the fly-by-wire primary flight control real-time software of a Boeing 777, operated by the Malaysia Airlines, in August 2005, which could have resulted in the loss of 177 passenger lives if the pilot had not manually taken over the autopilot program in time. All these incidents happened because the erroneous conditions were not caught during the analysis phase, due to the imprecise nature of the analysis techniques, and thus bugs appeared in the original product. Therefore, techniques like paper-and-pencil proof methods and simulation should not be relied upon for the analysis of real-time systems especially when they are used in safety or financial critical domains.

Formal methods are capable of conducting precise system analysis and thus overcome the above mentioned limitations. The main principle behind formal analysis of a system is to construct a computer based mathematical model of the given system and formally verify, within a computer, that this model meets rigorous specifications of intended behavior. A number of elegant approaches for the formal analysis of real-time systems can be found in the open literature using state-based or theorem proving techniques (e.g., Alur, 1992; Cardell-Oliver, 1992; Amnell, 2001; Beyer, 2003; Kwiatkowska, 2002; Bucci, 2005; Kwiatkowska, 2007; Hasan, 2009). However, some of these existing formal verification tools are only capable of specifying and verifying hard deadlines, i.e., properties where a late response is considered to be incorrect. Whereas, in the case of performance analysis of real-time systems, soft deadlines, i.e., properties that provide the quality of service in terms of probabilistic quantities or averages, play a vital role. Also, the above mentioned state-based approaches are limited by reduced expressive power of their automata based or Petri net based specification formalism. On the other hand, the higher-order-logic theorem proving based technique [Hasan, 2009] tends to overcome the above mentioned limitations of existing formal real-time system analysis techniques.

The main principle behind the higher-order logic theorem proving based approach is to leverage upon the high expressiveness of higher-order logic to formally specify and reason about the temporal properties and random behaviors of the present age complex real-time systems. This approach is primarily based upon the previous work reported for the functional verification of hard real-time systems [Cardell-Oliver, 1992], the formalization of random variables [Hurd, 2002] and the verification of expectation properties for discrete random variables [Hasan, 2008]. The idea is to formally specify the given real-time system as a logical conjunction of higher-order-logic predicates [Cadell-Oliver, 1992], whereas each one of these predicates defines an autonomous component or process of the given real-time system, while representing the unpredictable or random elements in the system as formalized random variables [Hurd, 2002]. The functional correctness and the performance related proper-