Sādhanā, Vol. 17, Part 1, March 1992, pp. 1–28. © Printed in India.

Modelling real-time systems: Issues and challenges

R K SHYAMASUNDAR and S RAMESH*

Computer Science Group, Tata Institute of Fundamental Research, Homi Bhabha Road, Bombay 400 005, India
*Department of Computer Science and Engineering, Indian Institute of Technology, Powai, Bombay 400 076, India

Abstract. In this paper, we discuss the issues and challenges that lie in the specification, development, and verification of real-time systems. In our presentation, we emphasize on the issues underlying modelling of real-time distributed concurrency.

Keywords. Real-time; reactive systems; concurrency; bisimulation; trace equivalence; scheduling.

1. Introduction

Real-time systems are designed to cater to many applications ranging from home appliances or laboratory instruments to process control systems, flexible manufacturing, flight control and tactical control in military applications. Flexible manufacturing is a special kind of real-time application where the behaviour of each manufacturing machine can be adapted *instantaneously* to continuously changing working conditions while still satisfying a global optimality criterion. In flight control systems *real-time* automatic manoeuvering is used to achieve significant reduction of fuel consumption and also for tactical control over the target. In these systems, the timely execution of requests and responses by the computers is critical to the successful operation of both the physical systems and the computer itself. That is, in addition to the normal functional requirements, it is necessary that responses to inputs (from the environment) must happen in a given interval of time. We refer to these systems as real-time systems and the specified intervals of time as *deadlines.* We use the qualification *reactive* to refer to the fact that the system has to respond to environment stimuli continuously. In such systems one can distinguish two kinds of deadlines.

- *Hard deadlines*: Here, it is important that the deadline must be met; otherwise the result is useless; in other words, *what is needed is the right output at the right time.*
- *Soft deadlines*: In these deadlines, not meeting the deadlines results in the degradation of the system performance.

One of the common concepts that counter a majority of the process control systems is that of providing continual feedback to an unintelligent environment. The continual demands of an unintelligent environment cause these systems to have relatively rigid and urgent performance requirements, such as real-time response requirements and *fail-safe* reliability requirements. It seems that this emphasis on performance requirements is what really characterizes time-critical systems, and causes us to be more aware of their roles in their environments than we are for other types of systems. The interface between a process control system and its environment tends to be complex, asynchronous, highly parallel and distributed. This is another direct result of the *process control* concept, because the environment is likely to consist of a number of objects which interact with the system and each other asynchronously in a parallel fashion. Furthermore, it is probably the complexity of the environment that necessitates computer support in the first place. This characteristic makes the requirements difficult to specify in a way that is both precise and comprehensible. Finally, embedded systems can be extraordinarily hard to test. The complexity of the system/environment interface is one obstacle, and the fact that these programs often cannot be tested in their operational environments is another. It is not feasible to test flight-guidance software by flying with it, nor to test ballistic-missile-defence software under battle conditions. Further, embedded systems are especially likely to have stringent resource requirements. These are requirements on the resources, mainly physical in this case, from which the system is constructed. This is because embedded systems are often installed in places (such as satellites) where the weight, volume or power consumption must be limited, or where temperature, humidity, pressure and other factors cannot be as carefully controlled as in the traditional machine room. It is important to note that a failure quite often results in economic, human and ecological catastrophes. Thus, safety and reliability are extremely important for time-critical process control systems. Various parameters one has to cope up with in building such systems can be seen from some of the main characteristics of real-time systems given below.

(a) The system tends to be large, complex and can be extraordinarily hard to test.
(b) The environment that the system interacts with is nondeterministic. That is, most of the times, there is no way to anticipate in advance the precise order of external events.
(c) High speed external events (perhaps in parallel), must be able to affect the flow of control in the system easily.
(d) The requests must be responded and handled within certain bounded time limits.
(e) The system is a coordinated set of asynchronous distributed units.
(f) The mission time is long. The system not only must deal with ordinary situations but also must be able to recover from some extraordinary ones.

It must be quite evident from the above characteristics that the design of complex real-time systems poses a serious challenge since many of the requirements and restrictions are often conflicting with one another. Thus, one of the most important needs is to design sound methodologies for the specification, verification and development of real-time systems that would support the common requirements of flexibility and predictability of systems. This would certainly go a long way in bridging the thin line between acceptable and unacceptable systems.

In this paper, we discuss the issues and challenges that lie in the specification, development, and verification of real-time systems with an emphasis on the modelling of distributed real-time concurrency. The rest of the paper is organized as follows: § 2 discusses aspects of real-time systems that make it different from other systems and the notion of time; § 3 surveys the issues of modelling real-time reactive systems in some detail as the study provides a basis for observation-based specifications. The challenges in the design of real-time systems are highlighted in § 4 followed by a discussion in § 5.