A Modelling and Simulation Based Approach to Dependable System Design

Miriam Zia, Sadaf Mustafiz, Hans Vangheluwe, and Jörg Kienzle

School of Computer Science, McGill University
Montreal, Quebec, Canada
{mzia2, sadaf, hv, joerg} @cs.mcgill.ca

Abstract. Complex real-time system design needs to address dependability requirements, such as safety, reliability, and security. We introduce a modelling and simulation based approach which allows for the analysis and prediction of dependability constraints. Dependability can be improved by making use of fault tolerance techniques. The de-facto example in the real-time system literature of a pump control system in a mining environment is used to demonstrate our model-based approach. In particular, the system is modelled using the Discrete EVent system Specification (DEVS) formalism, and then extended to incorporate fault tolerance mechanisms. The modularity of the DEVS formalism facilitates this extension. The simulation demonstrates that the employed fault tolerance techniques are effective. That is, the system performs satisfactorily despite the presence of faults. This approach also makes it possible to make an informed choice between different fault tolerance techniques. Performance metrics are used to measure the reliability and safety of the system, and to evaluate the dependability achieved by the design. In our model-based development process, modelling, simulation and eventual deployment of the system are seamlessly integrated.

1 Introduction

Model-based approaches are used to represent the structure and behaviour of systems, which are becoming increasingly complex and involve a large number of components and domain-specific requirements [1][2]. Dependable systems, in particular, must satisfy a set of functional requirements, and in addition, must adhere to constraints which ensure correct behaviour of the system. Safety, security and reliability are a few such dependability requirements. The necessity for accomplishing these constraints has spawned new fields of research. The most prominent area is that of fault-tolerant systems, and the introduction of fault tolerance design in the software development process is an emerging topic.

We are interested in developing the model-based process illustrated in Fig. 1 for designing a dependable system. The process allows us to predict the behaviour of a specific system, and compare it to the behaviour of a fault-tolerant implementation of the same system. This is done through a sequence of manual activities. First, from functional requirements, a model is derived which represents the structure of a chosen system. A fault injection mechanism is also modelled as a means to generate faulty behaviour of the system. Simulation results indicate how the system performs in the
presence of faults, and whether it conforms to the specified requirements. Secondly, from dependability constraints, a fault-tolerant model is created which includes techniques designed to improve on the initial system. A fault-tolerant simulation model is derived and simulated to gather performance data. This data reflects the dependability constraints that must be satisfied by the system.

Although research has been done in formal modelling and analysis of fault tolerance properties \cite{3} \cite{4}, either using natural language description of models, probabilistic models, figures of fault-trees or Markov models, we suggest using the formalism DEVS (Discrete Event System specification). In our case study, the initial system as well as the fault tolerant system are translated into DEVS.

The paper is structured as follows. Section 2 presents essential background concepts relating to the DEVS formalism and to fault tolerance. Section 3 describes the real-time Pump Control System (PCS) chosen to demonstrate our process. We introduce its functional requirements and dependability constraints and briefly discuss why modelling and simulation is an appropriate approach, and why DEVS is an suitable modelling formalism. Section 4 introduces the model of the PCS, and the means by which fault injection is introduced in the system. A PCS failure situation is described in Section 5, and a fault-tolerant model is presented that counteracts this failure. Furthermore, safety and reliability are defined as the dependability constraints that are threatened by failure of the PCS. In Section 6, implementation-specific and experimental simulation framework details are outlined. Mathematical equations are presented to quantify the safety and reliability of the PCS, and results of the simulations are analyzed to compare the performance of the PCS in the two models. Finally, some general conclusions about our model-based process are drawn in Section 7.