Journal of Applied Logic 18 (2016) 71–104

Contents lists available at ScienceDirect

Journal of Applied Logic

www.elsevier.com/locate/jal

Formal assessment of reliability specifications in embedded cyber-physical systems

Aritra Hazra[a,*], Pallab Dasgupta[b], Partha Pratim Chakrabarti[b]

[a] *Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai, Tamilnadu 600036, India*
[b] *Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, West Bengal 721302, India*

ARTICLE INFO

Article history:
Received 31 August 2015
Accepted 7 July 2016
Available online 5 September 2016

Keywords:
Formal methods
Temporal logic
Functional reliability
Fault-tolerant design
Verification
Cyber-physical systems
AND/OR graph search

ABSTRACT

Reliability has become an integral component of the design intent of embedded cyber-physical systems. Safety-critical embedded systems are designed with specific reliability targets, and design practices include the appropriate allocation of both spatial and temporal redundancies in the implementation to meet such requirements. With increasing complexity of such systems and considering the large number of components in such systems, redundancy allocation requires a formal scientific basis. In this work, we profess the analysis of the redundancy requirement upfront with the objective of making it an integral part of the specification. The underlying problem is one of synthesizing a formal specification with built-in redundancy artifacts, from the formal properties of the error-free system, the error probabilities of the control components, and the reliability target. We believe that upfront formal analysis of redundancy requirements is important in budgeting the resource requirements from a cost versus reliability perspective. Several case-studies from the automotive domain highlight the efficacy of our proposal.

1. Introduction

In embedded control of cyber-physical systems (CPS), intermittent failures may result out of various uncertainties in the computation platform, such as delay in receiving sensor data or posting actuation messages, stack overflows, memory or data corruption errors, and system crashes. These have been a major irritant towards the design of reliable software controlled embedded systems, and system failures arising out of such errors have been widely reported [25,23,31].

The traditional approaches towards design of reliable control systems are broadly divided into two categories. In the first category, control engineers design the control law in a way that guarantees robust control

[*] Corresponding author.
E-mail addresses: aritrah@cse.iitm.ac.in (A. Hazra), pallab@cse.iitkgp.ernet.in (P. Dasgupta), ppckak@cse.iitkgp.ernet.in (P.P. Chakrabarti).

http://dx.doi.org/10.1016/j.jal.2016.09.001
1570-8683/© 2016 Elsevier B.V. All rights reserved.

performance under minor disturbances. This category of approaches typically assume the computational platform to be ideal in certain ways. The second category of approaches, which are germane to the design of cyber-physical control, profess the use of redundancy in the application of control through the computing platform. This includes redundancy in sampling, processing the sensor data, execution of the control law and delivery of actuation messages. Since the modern computational platform is quite complex with multiple electronic control units (ECUs), various types of communication buses, real time operating systems, the task of redundancy allocation and assessing its impact on reliability is not obvious.

We believe that the task of creating the reliability blue-print of a CPS must begin by formalizing the reliability requirement at the behavioral level of abstraction. At this level, it is relatively natural to specify the redundancy in terms of repetitions of control actions and analyze their expected impact on the overall reliability of control. Such an early analysis helps in budgeting the reliability requirements of the components and their actions, which in turn falls in line with the existing component based design practices prevalent in the industry.

Formal specification development has become a part of the design process in various domains, notable among which is the domain of integrated circuits and systems. The use of formal specifications has been recommended in several recent international safety standards, including aeronautics (DO-178C), automotive (ISO 26262), industrial process automation (IEC 61508), nuclear (IEC 60880), railway (EN 50128) and space (ECSS-Q-ST-80C). Formal methods have been widely used in functional verification [5,7,6,8,22,29], and more recently in the analysis of performance attributes such as timing [1,2,10,11,27] and power [14,15]. The aspect which separated our work from that of existing literature is in formalizing reliability requirements in terms of behavior and methods for analyzing the realization of formal reliability specifications in terms of redundancy artifacts over such behaviors.

We consider the use of spatial and temporal redundancies, which has been studied in the context of implementation [16,33], but has not so far been abstracted for analytical purposes in the specification. Intuitively, spatial redundancy means that a component is physically replicated (or the control is repeated from different components) with the understanding that the probability with which both the replicas will fail at the same time is less than the probability of failure of each [17,20,26,32,3,24]. On the other hand, temporal redundancy alludes to repetitions in control actions (such as re-execution of some control component) [12,21,18,19].

Our primary goal in this work is to formally determine what form of behavioral redundancy is needed at the component-level so that an end-to-end feature is implemented with a desired level of reliability. Our main contribution is to establish that the proposed formal analysis can be leveraged to come up with a formal layout of the redundancy requirements in the components' behavior.

An intuitive description of our intent is as follows. In our level of abstraction, an *action* represents a discrete control task which is enabled by a logically defined pre-condition and achieves a logically specified consequent when executed successfully. Suppose φ is a consequent that is guaranteed by the successful execution of some non-trivial sequence of actions. Given that these actions, which may relate to different components of the system, are not necessarily reliable, our concern is to guarantee φ with some specified reliability, θ. In order to achieve this level of reliability, we may have to replicate (sequentially or in parallel) one or more of these actions. The goal is to optimize the redundancy so as to guarantee the desired level of reliability of one or more consequents like φ. Finally, our intention is to model the redundancy artifacts at the specification level, so that the required redundancy directives may be carried down into the implementation.

Therefore, formally our goal is to find/verify a modified specification ψ, such that any design that realizes ψ also realizes φ *with the desired reliability* θ. In essence, we elevate the reliability analysis to the early phases of design and incorporate the desired redundancy attributes in the specification itself, thereby alleviating many of the reliability concerns which appear today at the later stages of design. Including the redundancy requirement in the formal specification provides the scientific basis for designing reliable systems in a top–down manner.