Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Reliability Engineering and System Safety 92 (2007) 395–407

RELIABILITY ENGINEERING & SYSTEM SAFETY

www.elsevier.com/locate/ress

Fault-tolerant embedded system design and optimization considering reliability estimation uncertainty

Naruemon Wattanapongskorn[a,*], David W. Coit[b]

[a]*Department of Computer Engineering, King Mongkut's University of Technology Thonburi, 91 Suksawad 48, Ratburana, Tung-Kru, Bangkok 10140, Thailand*

[b]*Department of Industrial and Systems Engineering, Rutgers University, 96 Frelinghuysen Rd., Piscataway, NJ 08854, USA*

Received 3 March 2005; received in revised form 21 November 2005; accepted 19 December 2005
Available online 20 February 2006

Abstract

In this paper, we model embedded system design and optimization, considering component redundancy and uncertainty in the component reliability estimates. The systems being studied consist of software embedded in associated hardware components. Very often, component reliability values are not known exactly. Therefore, for reliability analysis studies and system optimization, it is meaningful to consider component reliability estimates as random variables with associated estimation uncertainty. In this new research, the system design process is formulated as a multiple-objective optimization problem to maximize an estimate of system reliability, and also, to minimize the variance of the reliability estimate. The two objectives are combined by penalizing the variance for prospective solutions. The two most common fault-tolerant embedded system architectures, *N*-Version Programming and Recovery Block, are considered as strategies to improve system reliability by providing system redundancy. Four distinct models are presented to demonstrate the proposed optimization techniques with or without redundancy. For many design problems, multiple functionally equivalent software versions have failure correlation even if they have been independently developed. The failure correlation may result from faults in the software specification, faults from a voting algorithm, and/or related faults from any two software versions. Our approach considers this correlation in formulating practical optimization models. Genetic algorithms with a dynamic penalty function are applied in solving this optimization problem, and reasonable and interesting results are obtained and discussed.

Keywords: Estimation uncertainty; Reliability analysis; Fault tolerance; Embedded system; Genetic algorithm

1. Introduction

Determination of a recommended system design architecture involves the selection of available software and hardware components with the goal of maximizing system reliability, given constraints on the system. In the determination of optimal system designs, component reliability is not known exactly but must be estimated with some uncertainty. If the selected components have very high-reliability estimation uncertainty, then this can result in a system design which also has very high, and perhaps unacceptable, system reliability estimation uncertainty. This is undesirable because system designers and users seek an optimal design with high-predicted reliability, but also one with low-estimation uncertainty.

There is existing research [1–6] on system reliability optimization considering component reliability/failure rate uncertainty. Rekab [1] considers reliability estimation uncertainty of a series system assuming all components in the system function s-independently, and the system is not fault-tolerant. The assumption of independent component failures has been used in many published research. However, there is evidence showing that such assumption is not realistic [7,8]. Rubinstein et al. [2] consider a redundancy allocation problem with uncertain component reliability, by maximizing the expected value of system reliability using genetic algorithms (GAs). Their approach,

[*] Corresponding author. Tel.: +662 470 9089; fax: +662 872 5050.
E-mail address: naruemon@cpe.kmut.ac.th (N. Wattanapongskorn).

0951-8320/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ress.2005.12.011

considering only the expected values of reliability, is not sufficient for many decision makers because it ignores undesirable high uncertainty or risk associated with reliability estimation. In practice, system designers and users desire a designed system with a high-reliability estimate, associated with low-estimated variability. Coit et al. [3–5] solve the problem by considering variance of system reliability estimates in addition to the expected system reliability value. Zafiropoulos et al. [6] perform reliability and cost optimization of an electronic system but does not consider failure dependencies.

There has also been published research [9–15] considering failure dependencies in the system design and optimization. Unlike these results, the models presented in this paper consider both component reliability estimation uncertainty and redundancy with heterogeneous software components, hardware components, and failure dependencies in multiple software versions.

The new models combine the approach of considering both the system reliability estimate and its variance, with the embedded system optimization approach. This is an extension of work from Wattanapongsakorn and Levitan [16] where component reliability is known with certainty. This results in practical reliability optimization models for the design of fault-tolerant embedded systems.

An embedded system consists of both hardware and software components where software components are embedded in (and running on) hardware components. To make it fault-tolerant, redundancy techniques can be applied to obtain fault-tolerant architectures. In this paper, *N*-Version Programming (NVP) architectures and Recovery Block (RB) architectures are considered. The detailed description of these architectures is discussed in Section 2. The fault-tolerant systems are capable of tolerating software faults and/or hardware faults. For many systems, it is known that the majority of system failures are related to software faults. Therefore, optimal design of software fault-tolerance is often more critical than hardware fault-tolerance optimization. The fault-tolerant embedded system architectures result from different strategies of integrating software and hardware redundancy, together with some decision algorithms such as voting, acceptance test and comparison [9,13,17].

Similar to Wattanapongsakorn et al. [16], we consider a system where each subsystem is connected in series. Each subsystem consists of both software and hardware components. The software components are application software modules, and the hardware components are processing units (with operating system, disk, etc.) or network elements. The systems that we model are series-parallel fault-tolerant systems. The redundancy allocation problem for series-parallel systems is known to be difficult (i.e., NP-hard). Many researchers have proposed a variety of approaches to solve this problem using, for example, integer programming, dynamic programming, mixed integer and nonlinear programming. Recent optimization approaches [16,18–20] are based on heuristic search algorithms (or meta-heuristics) such as simulated annealing, (GAs), and Tabu Search (TS). All of these approaches were developed for either optimizing reliability for software or hardware systems. Here, we consider systems consisting of both software and hardware components.

Optimization models have been developed to select both software and hardware components and redundancy levels given a total system cost constraint. In the system, there are a specified number of subsystems in series. For each subsystem, there are several choices of heterogeneous hardware and software components/versions to be selected. The system is designed using components, each with estimated reliability, but with known cost. Additionally, the variance or standard deviation of the estimated system reliability is known or can be approximated using standard statistical methods.

GAs are used as the optimization approach. The term 'genetic' derives from the roughly analogous natural reproduction of new populations by crossover and mutation operators. There are competitions among the population; the stronger ones will survive to the next generation and the weak ones will soon die out. GA is a heuristic optimization model that has been applied effectively to solve many difficult problems in different fields such as scheduling, facility layout, and graph coloring/graph partitioning problems. It is a stochastic algorithm with performance depending on the solution encoding, crossover breeding operator, elitist selection and mutation operator.

In Section 2, a description of the fault-tolerant system architectures are presented. Section 3 presents the concept of reliability estimation variability for each of the system architecture models, including the higher-order information of component reliability estimates. Section 4 presents four optimization models to maximize reliability considering uncertainty. The first model does not consider component redundancy, while the other three models each do consider a specific fault-tolerant architecture type. Section 5 explains the GA and its parameter settings. In Section 6, the effectiveness of our optimization models is demonstrated using numerical examples. Lastly, in Section 7, the paper ends with a summary and conclusions.

1.1. Assumptions

1. Each software component, hardware component and the system has 2 states: functional or failed.
2. Reliability of each software or hardware component is unknown, but it can be estimated.
3. There is no repair for each component or the system.
4. Hardware redundancy is in active mode (i.e., hot spares).
5. Failures of individual hardware components are *s*-independent.

1.2. Notation

$X/i/j$ system architecture X (NVP or RB) with i hardware faults tolerated and j software faults tolerated