The Journal of Systems and Software 84 (2011) 835–846

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Reliability-driven deployment optimization for embedded systems

Indika Meedeniya[a], Barbora Buhnova[b], Aldeida Aleti[a], Lars Grunske[a,*]

[a] Faculty of ICT, Centre for Complex Software Systems and Services (CS3), Swinburne University of Technology, Hawthorn, VIC 3122, Australia
[b] Faculty of Informatics, Masaryk University, 60200 Brno, Czech Republic

ARTICLE INFO

Article history:
Received 29 September 2009
Received in revised form 7 December 2010
Accepted 3 January 2011
Available online 28 January 2011

Keywords:
Reliability evaluation
Optimization
Embedded systems
System deployment

ABSTRACT

One of the crucial aspects that influence reliability of embedded systems is the deployment of software components to hardware nodes. If the hardware architecture is designed prior to the customized software architecture, which is often the case in product-line manufacturing (e.g. in the automotive domain), the system architect needs to resolve a nontrivial task of finding a (near-)optimal deployment balancing the reliabilities of individual services implemented on the software level.In this paper, we introduce an approach to automate this task. As distinct to related approaches, which typically stay with quantification of reliability for a specific deployment, we target multi-criteria optimization and provide the architect with near-optimal (non-dominated) deployment alternatives with respect to service reliabilities. Toward this goal, we annotate the software and hardware architecture with necessary reliability-relevant attributes, design a method to quantify the quality of individual deployment alternatives, and implement the approach employing an evolutionary algorithm.

1. Introduction

Reliability is one of the quality attributes that should drive important decisions in embedded-system design (e.g. in the automotive domain (Broy, 2006; Pretschner et al., 2007)). System deployment, i.e. mapping of software components to hardware hosts (see Fig. 1 which shows two mapping alternatives), is a typical example of a design task, which can be optimized to maximize the reliability of a system or its parts. In embedded systems, the software level typically implements a set of high-level services (such as the *ABS* or *Cruise Control* service in a car), whose reliabilities cannot be maximized jointly, but need to be understood as conflicting objectives. Since the deployment-optimization task is known to be NP hard Garey and Johnson (1979), finding good solutions becomes infeasible for both human and computerized exact methods already for relatively small settings (tens of software and hardware components) Garey and Johnson (1979), and requires employment of sophisticated methods and heuristics.

The problem of automatic exploration of deployment alternatives, and reporting a set of near-optimal solutions has already been addressed with respect to various system quality attributes, including latency, performance, security, or resource usage (Mikic-Rakic et al., 2004; Medvidovic and Malek, 2007; Malek, 2007; Sharma et al., 2005; Fredriksson et al., 2005). The approaches dealing with reliability are however very scarce, and typically address slightly different problems, like clustering of software systems (Suri et al., 1998; Bastarrica et al., 1998; Islam et al., 2006), or redundancy allocation (Grunske, 2006a; Liang and Smith, 1999; Coit and Smith, 1996), rarely taking the hardware reliability into account.

Our approach of optimize software deployments, with respect to conflicting service reliabilities, is inspired by both sides. We follow the direction started by Medvidovic and Malek (2007) and Malek (2007), who propose an approach to solve the deployment-optimization problem under very general settings (irrespective of the specific quality attribute), and hence cannot directly benefit from the specifics related to a specific quality attribute domain. In our approach, which is also inspired by the AUTomotive Open System ARchitecture (AUTOSAR) initiative (Heinecke et al., 2004), we adopt the reliability model introduced by Kubat (1989), and extend it in two directions. First, we define the propagation of hardware-level reliabilities to the reliabilities of software-level services. Second, we extend the model also to inter-component communication by viewing component links as first-class entities in the probabilistic model.

To achieve the aim, three main tasks are completed in this paper: (1) The identification of the reliability-relevant attributes of embedded systems with respect to system deployment and their formulation in terms of a modeling language. (2) The formalization of the propagation of hardware-level reliability measures to the software level, and the definition of the reliability-driven quality of a single deployment alternative over the implemented services. (3) The automatic exploration of the problem space by using evolutionary algorithms to solve the specific deployment problem.

* Corresponding author.
E-mail addresses: imeedeniya@swin.edu.au (I. Meedeniya), buhnova@fi.muni.cz (B. Buhnova), aaleti@swin.edu.au (A. Aleti), lgrunske@swin.edu.au (L. Grunske).

0164-1212/$ – see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2011.01.004

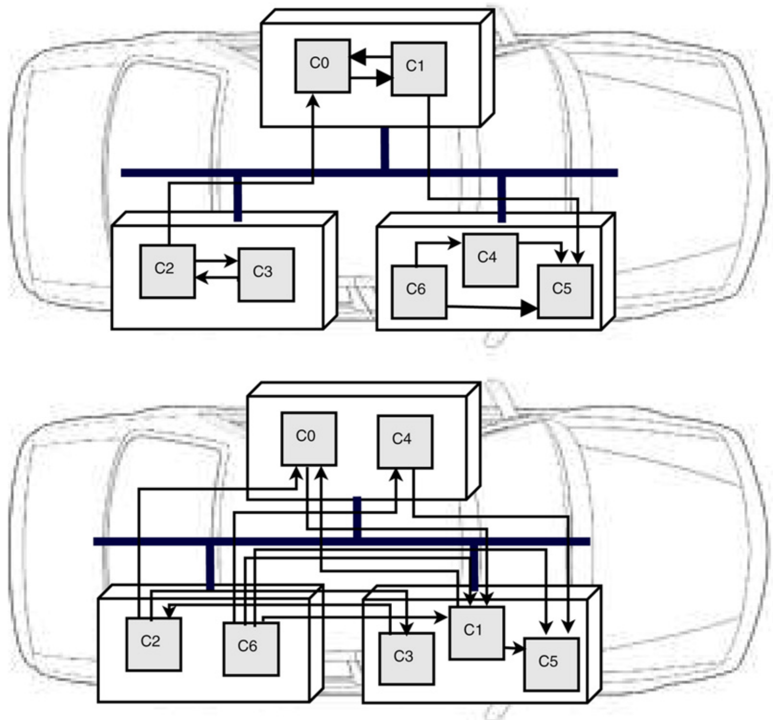

Fig. 1. Two alternative deployments of the same system, with presumably very different reliability characteristics.

Paper outline: After reviewing the related work in Section 2, we identify the reliability-relevant properties of embedded systems with respect to the deployment problem in Section 3. The main part of the paper is contained in Section 4, which formalizes the problem, and introduces our approach to address it. Section 5 presents tool support for our approach, and Section 6 illustrates the approach on a case study of an *Automotive Control System*. In Section 7 we present the assumptions and limitations of our approach. Section 8 concludes the paper and presents our directions for future work.

2. Related work

This section overviews the approaches related to ours with respect to the goal of reliability-driven deployment optimization. The main characteristics, in terms of targeted problem, quality attributes, design-space exploration technique, and employed search algorithms and heuristics, of closely related approaches are summarized in Table 1.

Deployment: The influence of system deployment to system quality has been observed by many authors, in both software and embedded systems domain (Sharma et al., 2005; Malek, 2007; Nicholson, 1998; Papadopoulos and Grante, 2003; Pace et al., 2008). To evaluate the quality of a system with respect to a selected deployment alternative, new models propagating hardware quality attributes to the software level and/or user interface have been introduced (Bastarrica et al., 1998; Bondarev et al., 2005; Islam et al., 2006; Grunske et al., 2005, 2006; Assayad et al., 2004; Calinescu and Kwiatkowska, 2009). Besides the quantification of system quality taking system deployment into account, methods have been designed to automatically search the space of possible deployment options and reporting promising candidates. Some of the methods are aiming at any deployments satisfying predefined constraints or user requirements (Kichkaylo et al., 2003; Calinescu and Kwiatkowska, 2009; Martens and Koziolek, 2009), others are targeting methods to find an optimal deployment or at least candidates being close to it (Medvidovic and Malek, 2007; Sharma et al., 2005; Fredriksson et al., 2005; Nicholson, 1998; Blickle et al., 1998), often in combination with given constraints (Mikic-Rakic et al., 2004; Medvidovic and Malek, 2007; Fredriksson et al., 2005; Blickle et al., 1998; Lukasiewycz et al., 2008).

Reliability: The aforementioned approaches focus on different system quality attributes, including latency, performance, security, or resource usage (Mikic-Rakic et al., 2004; Medvidovic and Malek, 2007; Fredriksson et al., 2005; Blickle et al., 1998; Malek, 2007; Sharma et al., 2005). Reliability, however, is not getting much attention in the specific context of deployment optimization. On the other hand, related approaches can be identified in the domains of clustering of software systems (Bastarrica et al., 1998; Islam et al., 2006; Suri et al., 1998), network design (Glaß et al., 2008), redundancy allocation (Coit and Smith, 1996; Grunske, 2006a; Liang and Smith, 1999). These approaches are inspiring in the identification of reliability-relevant properties of systems, but miss the incorporation of hardware architectures into the picture, which is crucial for the deployment problem.

Our approach bridges this gap by connecting the experience of deployment-targeting approaches (namely of Medvidovic and Malek (2007) and Malek (2007)), and existing reliability models (namely the Kubat (1989) model), which results in a new reliability-driven approach to the system deployment problem. An additional novel aspect that we consider is optimizing the deployment with respect to reliability estimates of different services.

3. Reliability-driven modeling

Embedded systems, such as avionic or automotive control systems, closely interact with the physical environment, typically via sensors and actuators. They are deployed on compact self-contained computational units (containing both processing and memory part), known as *Electronic Control Units (ECUs)*. All the hardware modules (ECUs, sensors, actuators) are connected