Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 83 (2016) 417 – 424

Procedia Computer Science

The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)

An Online Self-Adaptive System Management Technique for Multi-Core Systems

Hyunwoo Kim and Hoeseok Yang*

Department of Electrical and Computer Engineering, Ajou University, Suwon, 16499 Republic of Korea

Abstract

This paper presents a light-weight online system adaptation technique for multi-core embedded systems running multiple applications. Thus far, online system adaptation techniques are restricted to reconfiguring resource management schemes such as operating frequency scaling or task-to-processor assignment. Additionally, in this paper, we enable to adapt the algorithm at runtime. That is, we selectively apply a suitable algorithm considering the system status when more than one algorithm candidates exist in the target application. We propose a generic and extensible self-adaptive framework with multiple applications in mind by providing generic programming interface that requires minimal changes in the legacy software code. It is shown that the proposed framework adaptively optimizes both resource management and algorithm selection with negligible performance overhead. The effectiveness of the proposed framework is experimentally proven with real-life examples.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Self-Adaptive; Algorithm Reconfiguration; Multi-Core; Dynamic Voltage Frequency Scaling

1. Introduction

Traditionally, embedded systems are designed and optimized with a static and fixed execution scenario in mind. In architectural synthesis, for example, execution time of each operation is assumed to be constant, then the optimized schedule is computed based on that[1]. However, as the design level of abstraction continues to get higher, such fixed and static execution scenarios become impractical as reliable grounds for compile-time optimization. Indeed, today's embedded systems have to deal with many uncertainties including fluctuating task execution time and workload variation. Satish et al.[2] showed that execution times of P-macroblock processing in H.264 decoder vary from 10 to 70 ms, making traditional scheduling techniques incompetent. Marculescu et al.[3] reported that workload variations of modern embedded systems are closely related with physical processes, which cannot be easily captured at design-time. Therefore, in order to stay efficient in such dynamic cases, it is necessary to let the system reform its configuration in a self-adaptive way at runtime, which is the main focus of this paper.

* Corresponding author. Tel.: +82-31-219-2361 ; fax: +82-31-219-2361.
E-mail address: hyang@ajou.ac.kr

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.04.204

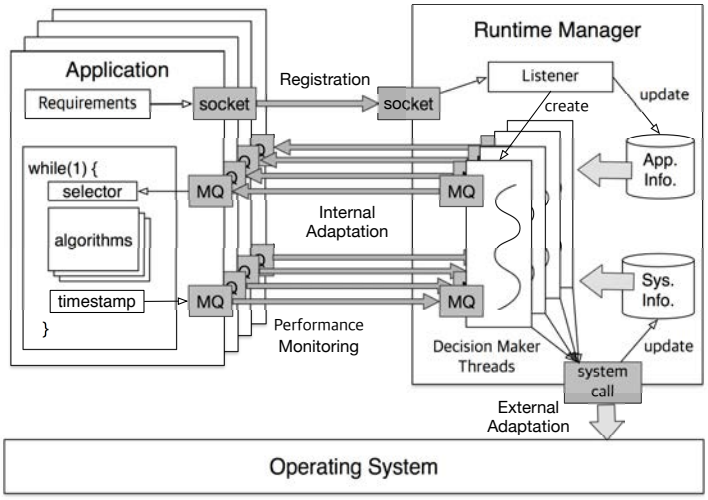

Fig. 1: Overview of the proposed self-adaptive framework.

Thus far, most self-adaptive computing techniques have focused solely on efficient managements of computing resources, such as optimal processor frequency modulation[4], task scheduling[5,6], task-to-processor assignment[7], and so forth. In this paper, reconfiguring the resource usage or management is called *external adaptation* as the control knobs are outside the application. On the other hand, there also exist adaptable parameters inside applications themselves. Altering such software parameters or algorithms is termed *internal adaptation* as opposed to the external adaptation. Hoffmann et al.[8] took internal adaptations into consideration by enabling dynamic reconfiguration of software parameters in various applications. However, it required in-depth understandings on the target applications due to the lack of generic programming APIs, thus limited the applicability of the technique. Moreover, it is not clear how they support multiple applications in their framework.

In this paper, we propose an online self-adaptive algorithm adaptation technique for multi-core embedded systems with multiple applications, which is able to adaptively manage the computing resources as well as software/algorithm knobs. The system gets reconfigured when the performance is unallowably degraded. In order to detect such performance degradations, we continuously monitor the performance of target applications. When necessary, internal or external adaptations are triggered through the provided APIs as will be explained in Section 3.

The contributions of this paper can be summarized as follows:

- we propose an unified self-adaptive framework that can deal with both internal and external adaptations;
- we propose an extensible self-adaptive framework that can allow spawning new applications at runtime;
- we facilitate applying self-adaptive computing technique by providing generic APIs for internal and external adaptations.

2. Overall Framework

Fig.1 illustrates the overview of the proposed self-adaptive computing framework. In the proposed technique, it is the duty of the programmer to insert the code for performance monitoring (time stamp) and internal adaptation (selector) using the provided API as depicted in Fig. 1. A system-wide runtime manager daemon is active at all times, waiting for the arrival of a new application. Once an application is initialized, using the provided initialization API, the event is automatically notified to the runtime manager. Note that the runtime manager is implemented in a multi-threaded way in order to be able to adaptively manage multiple applications independently and simultaneously. To be more precise, the runtime manager has the following three tasks: