Offline Phase Analysis and Optimization for Multi-Configuration Processors

Frederik Vandeputte, Lieven Eeckhout, and Koen De Bosschere

Ghent University, Electronics and Information Systems Department,
Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium
{fgvdeput, leeckhou, kdb}@elis.UGent.be

Abstract. Energy consumption has become a major issue for modern microprocessors. In previous work, several techniques were presented to reduce the overall energy consumption by dynamically adapting various hardware structures. Most approaches however lack the ability to deal efficiently with the huge amount of possible hardware configurations in case of multiple adaptive structures. In this paper, we present a framework that is able to deal with this huge configuration space problem. We first identify phases through profiling and determine the optimal hardware configuration per phase using an efficient offline search algorithm. During program execution, we inspect the phase behavior and adapt the hardware on a per-phase basis. This paper also proposes a new phase classification scheme as well as a phase correspondence metric to quantify the phase similarity between different runs of a program. Using SPEC2000 benchmarks, we show that our adaptive processing framework achieves an energy reduction of 40% on average with an average performance degradation of only 2%.

1 Introduction

Energy dissipation is a major design issue for modern microprocessors both in the embedded, the general-purpose as well as the high performance market segments. To address this issue several researchers have proposed to dynamically tune or resize several hardware resources without affecting overall performance, thereby reducing energy consumption.

Generally speaking, we can identify three major classes of adaptive processing: resource-driven, positional and temporal adaptation. In resource-driven adaptation [1][2], the various hardware components tune themselves according to their current use. In positional adaptation [3], particular architectural configurations are associated with particular code sections, for example at the level of subroutines. Each time the processor enters one of those sections, the corresponding architectural configuration is installed. In the temporal approach [4][5][6], the program execution is partitioned into fixed-length intervals, recurring phases are identified and energy-efficient processor configurations are associated per phase. Phase identification and hardware adaptation is all done dynamically. A common problem is how to deal efficiently with a large number of hardware configurations in case of multiple adaptive structures. Finding the optimal configuration through enumeration (as is typically done in previously proposed dynamic optimization schemes) obviously is not an option.