

FAST BENCHMARKING FOR PROCESSOR DESIGN

Afzal Hossain,* Daniel J. Pease,** and James S. Burns***

Abstract

Architecture simulations may take days during design when silicon is not available. Long simulation time is impractical; yet the cost of early design mistake is high. Hence, fast architectural parameter exploration without rigorous simulation is an important problem. Analytical models can execute fast. However, few analytical models can produce benchmark performance of a design. The presented method, termed *Fast Benchmarking*, based on analytical models, can produce benchmark performance of a processor in a few hundred milliseconds. With accurate model, performance data produced by the tool is accurate. For example, instruction fetch results differ by $\pm 7\%$ on average with simulations. This article further extends the concept by presenting a new method for analytically guided adaptive architecture simulation. To our knowledge, no other work has presented a full methodology to use analytical models for the study of processor performance during design. The method gives birth to a valuable tool that can be used in the industry for designing high-performance multi-core, multi-threaded processors.

Key Words

Processor, architecture, benchmarking, design, modelling, simulation

1. Introduction

Processors and applications are becoming ever larger and more complex. Designers are challenged with a maze of entangled and competing design goals. Many expensive mistakes are made during early stage of architecture definition. Early stage mistakes are also difficult to fix later. Simulation has remained a primary vehicle to cope with this challenge. But timing-accurate full system simulations can take even years [1].

Benchmarking is a primary tool for studying performance of different design choices. A software model of the

* Nanova Corporation, 4962 El Camino Real, Suite 104, Los Altos, California 94022, USA; e-mail: afzal.hossain@nanova.com

** Department of Electrical Engineering and Computer Science, CST 3-191, Syracuse University, Syracuse, NY 13244; e-mail: dapease@syr.edu

*** Intel Corporation, 2200 Mission College Blvd, Santa Clara, CA 94054; e-mail: james.s.burns@intel.com

Recommended by Prof. M.D. Rossetti

(DOI: 10.2316/Journal.205.2012.1.205-5392)

architecture, popularly known as architecture model, executes the benchmarks. The process, known as architecture simulation, can produce data and charts to help with design decisions. Among others, the SPEC CPU2006 benchmarks stress processor, memory, and compiler performances [2].

Architecture simulation requires a carefully crafted simulator that must model the architecture accurately along with realistic workloads. The details take toll on simulation speed. The CPU2000 includes a suit of 26 programs with a combined total of 8 trillion instructions [1]. To make matters worse, hundreds of simulations are often needed to produce meaningful data. The analysis of massive amount of simulation data is also a daunting task. Developing insight about a design to guide hundreds of simulations is even more challenging.

This is a big problem and there are conferences and journals for this subject. Even so, a good solution to avoid lengthy simulations has remained illusive. The problem, frustrations, and the state of the art can be found concisely in a special issue of *IEEE Micro* [1], [3]–[5]. In the guest editors' words, "the desire for faster and more accurate simulation is by no means satiated".

The problem can be observed in the research design of many dissertations on computer architecture. For example, Burns in his dissertation on on-chip SMT processors presented large number of performance graphs. He wrote the POSM simulator and ran hundreds of simulations in each configuration of POSM [6]. Hossain tried to alleviate the architecture simulation problem by using analytical models [7]. In other works, Binkert, Dreslinski, Hsu, Lim, Saidi, and Reinhardt presented the M5 simulator. M5 provides a capability to simulate multiple systems in a network [4]. In their work on hybrid-compiled simulation, Reshadi, Mishra, and Dutt improve interpretive simulation performance by applying compiled simulation [8].

There are efforts to avoid full benchmark simulations by taking samples of executions only, while still trying to maintain accuracy. SimFlex and SimPoint are two good examples [3], [5]. A challenge in realizing sampling lies in constructing the correct initial state for a large number of fine-grained measurements. For what is known as the warming problem, Wenisch, Wunderlich, Ferdman, Ailamaki, Falsafi, and Hoe present a solution in Livepoints, which allows simulation in 91 seconds [3]. To reduce randomness in SimPoint, Van Biesbrouck, Calder, and