Guest Editors' Introduction: Approximate Computing

Qiang Xu
The Chinese University of Hong Kong

Nam Sung Kim
University of Illinois at Urbana-Champaign

Todd Mytkowicz
Microsoft Research

Classical technology scaling, also known as Dennard's scaling has tremendously improved computer's performance over the past decades, which in turn has enabled countless innovative applications benefiting our daily lives today. However, as the classical technology scaling has approached its fundamental physical limit, the improvement in computer's performance has been sluggish, whereas the amount of information for computers to process exponentially grows.

Meanwhile, computers are increasingly used to process, comprehend, and use a large amount of information originated from interaction with the physical world, as computers and their applications become pervasive. We expect them to be context-aware and present natural human interfaces. Consequently, many applications that involve nontraditional input sources (e.g., sensors), commonly referred to as recognition, mining, and synthesis (RMS) applications, have emerged and are rapidly gaining prominence.

These applications process noisy data sets and/or involve a human interface with limited perceptual capability, and there is usually no specific "golden" output value that must be computed. By exploiting the inherent error resilience in such applications, approximate computing is an approach that seeks to relax the numerical equivalence between the specification and implementation of error-tolerant applications and trades off computational effort (e.g., energy) with computation quality (e.g., accuracy), thus achieving significant improvement in energy efficiency and/or performance.

While approximate computing has gained significant traction in recent years, it is still in its infancy and the research and industrial communities need new design methodologies and innovative concepts to address the challenges in this area before it becomes mainstream energy-efficient computing solution.

In response to such need, it is our great pleasure for us to introduce this Special Issue on Approximate Computing, which highlights recent investigations in this field and the selected articles include one survey article prepared by the guest editors and four regular articles.

In the first article "Approximate Computing: A Survey," the guest editors present a survey of the literature, which puts recent approximate computing work in circuit, architecture, and software in perspective and offer various insights and research challenges in this field.

In "An eDRAM-based Approximate Register File for GPUs," Jeong et al. first note that SRAM-based register file will become a scalability bottleneck due to its area and power overhead, as GPUs demand larger register file for higher performance. As an alternative, the authors propose a register file architecture based on eDRAM which can offer higher density and lower leakage power than SRAM. Especially, observing that eDRAM often suffers from frequent refresh operations, the authors propose to approximate low-order bits of register file by refreshing them at a much lower rate and significantly improve energy efficiency without notably impacting the compute accuracy.

Digital Object Identifier 10.1109/MDAT.2015.2509607
Date of current version: 18 January 2016.

Copublished by the IEEE CEDA, IEEE CASS, IEEE SSCS, and TTTC

In the next article "Mitigating the Memory Bottleneck with Approximate Load Value Prediction," Yazdanbakhsh et al. aim to tackle two fundamental memory bottlenecks for GPUs: limited off-chip bandwidth and long access latency with a novel architectural technique, rollback-free value prediction (RFVP). More specifically, the authors propose to predict the values of some safe-to-approximate load operations upon cache misses while evading costly recovery operations upon mispredictions, considerably improving both performance and energy efficiency.

In "Quality Control for Approximate Accelerators by Error Prediction," Khudia et al. highlight the importance of quality control in approximate computing. Then the authors propose Rumba, a lightweight online technique that detects and corrects large errors in an approximate accelerator-based computing environment. Rumba significantly reduces computing error without notably degrading the performance gain attained by approximate computing.

In the last article "Exploring the Precision Limitation for RRAM-Based Analog Approximate Computing," Li et al. first take resistive-switching random access memory (RRAM) as a promising substrate for analog approximate computing, and demonstrate the impact of limited precision in RRAM bit levels and analog-to-digital (AD) and digital-to-analog (DA) conversions. Then, the authors jointly analyze the impact of RRAM bit levels and AD/DA resolution.

It was a remarkable experience for the three guest editors with complementary expertise to successfully prepare this special issue, and we would like to acknowledge the help we received from many dedicated individual contributors that made this special issue possible. Especially, we thank the authors who submitted articles to this special issue, as well as the reviewers for their precious time and dedicated effort to provide insightful comments to the authors. We also thank Prof. Andre Ivanov, the former Editor-in-Chief of *IEEE Design & Test*, for helping us to create this special issue, and for his support and patience during the entire process, which involved soliciting articles, inviting reviewers, and gathering their feedback. Last, we thank the editorial staff of the IEEE Publishing Operations for their excellent job in editing and assembling this issue. ■

Qiang Xu is an Associate Professor in Computer Engineering at the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong. His current research interests include fault-tolerant computing and trusted computing. His research has received the IEEE/ACM Design Automation and Test in Europe (DATE) Best Paper Award in 2004. He is a Member of the IEEE.

Todd Mytkowicz is a Researcher at Microsoft Research, Redmond, WA, USA, working at the intersection of programming languages and systems, with a specific focus on abstractions for parallelism and performance. His research interests span program analysis, probabilistic programming, verification, and concurrency. His research received three SIGPLAN research highlights nominations, was chosen as an IEEE Micro Top Pick, and is used in production systems at Microsoft.

Nam Sung Kim is an Associate Professor at the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, USA. His current research interests span circuit, architecture, and system for energy-efficient computing. His research received the IEEE/ACM International Symposium on Microarchitecture (MICRO) Best Paper Award and was chosen as an IEEE Micro Top Pick. He is a Fellow of the IEEE.

■ Direct questions and comments about this article to Qiang Xu, Computer Science & Engineering Dept., The Chinese University of Hong Kong, Shatin, Hong Kong; qxu@cse.cuhk.edu.hk.