J Sign Process Syst (2015) 80:1–2
DOI 10.1007/s11265-014-0964-2

Data Intensive Computing: From Modeling to Implementation

Pekka Jääskeläinen · Jani Boutellier

Published online: 10 January 2015

Data intensive computing encompasses applications that mostly perform data processing in regular patterns. Traditionally, such applications have been found, e.g., in image processing, audio processing, telecommunications and radar signal processing. With the immense growth of the mobile consumer electronics market, data intensive computing is present in billions of devices.

In contrast to general-purpose computing, the regularity of data intensive computing enables intelligent design choices to be made in specification, modeling, design space exploration, software design, compilation and hardware design. However, unlike classic signal processing, contemporary data intensive applications, such as a H.265 video decoding or cognitive radio radio systems cannot be described with fully static representations. This dynamic nature of contemporary data intensive applications has its implications to all layers of system design, from higher-level software to hardware implementation.

This special issue covers topics from the whole spectrum of design aspects for data intensive computing. Included are three articles that address dataflow specification and analysis of applications, four articles that address programmable processing hardware for data intensive applications and one publication that deals with both software and hardware design aspects.

Article *Parameterized Sets of Dataflow Modes and Their Application to Implementation of Cognitive Radio Systems (doi:10.1007/s11265-014-0938-4)* presents a novel dataflow modeling technique named parameterized sets of modes (PSM) that addresses the important issue of modeling reconfigurable data intensive systems. After presenting a formal definition of the new concept, a cognitive radio application example is presented.

Memory Analysis and Optimized Allocation of Dataflow Applications on Shared-Memory MPSoCs (doi:*10.1007/s11265-014-0952-6*) is an article that concentrates on the memory usage analysis and optimization of applications that target MPSoC platforms. The application is modeled with the well-known Synchronous Dataflow formalism, based on which the memory usage analysis and optimization is performed. Experiments are performed with several data intensive applications on Intel and Texas Instruments DSP multicores.

The paper *Modeling Resolution of Resources Contention in Synchronous Data Flow Graphs (doi:10.1007/s11265-014-0923-y)* addresses interprocessor communication contention on multicore platforms. The proposed approach uses Synchronous Dataflow graphs to model applications and to perform the resource contention analysis. Experiments consist from random application graphs to well-known data intensive applications, such as video and audio codecs.

When going towards the hardware implementation level, issues such as processor instruction-set architectures suitable for data intensive computing become of interest. To this end, *Code Density and Energy Efficiency of Exposed Datapath Architectures (doi:10.1007/s11265-014-0924-x)* gives an overview of an instruction-set design direction where non-traditional aspects of the datapath are exposed to the control of the programmer in order to improve energy efficiency and instruction-level parallelism. The article looks at several proposed variations of the so called *exposed datapath architectures* with focus on their instruction word width overheads.

A Low Energy Wide SIMD Architecture with Explicit Datapath (doi:10.1007/s11265-014-0950-8) proposes a data-oriented processor architecture which exposes the bypass

P. Jääskeläinen (✉)
Tampere University of Technology, Tampere, Finland
e-mail: pekka.jaaskelainen@tut.fi

J. Boutellier
University of Oulu, Oulu, Finland

network to the programmer. The authors measure major reductions in energy consumption in comparison to alternatives where the register file is bypassed using hardware in the context of an SIMD instruction-set. Strongly related to this work is *A Co-Design Framework with OpenCL Support for Low Energy Wide SIMD Processor (doi:10.1007/s11265-014-0957-1),* which discusses hardware-software co-design and compilation aspects for processors that utilize the idea of explicit bypasses in SIMD datapaths. The OpenCL parallel programming standard is used as a basis for programming the devices designed using the described co-design flow. The code generator of the compiler utilizes an iterative process optimized for increasing the exploitation of the programmable bypass network while reducing the general purpose register pressure.

Among the other interesting features of the tiled homogeneous manycore architecture, Loki, presented in *Exploiting Tightly-Coupled Cores (doi:10.1007/s11265-014-0944-6),* is its instruction set which exposes the interconnection network between the relatively small RISC-style cores to the programmer. Using this idea, mesh-interconnected tiles and cores inside the tiles can communicate efficiently to form entities of collaborating cores, thus being capable of adapting to different parallelism patterns in applications. An implication of this that is topical to this special issue is the ability to use these features to support dataflow style processing by means of core local instruction storages and blocking network accesses.

Finally, *Embedded multi-core systems dedicated to dynamic dataflow programs (doi:10.1007/s11265-014-0953-5),* utilizes the benefits of high-level dataflow descriptions combined with processing platform customization using low power exposed datapath cores. It proposes a methodology to implement data flow execution platforms consisting of distributed local memories and exposed datapath cores. In addition, the authors propose several software optimizations to improve the performance, and exemplify the techniques using real word video decoders as case studies.

The articles in this issue cover the topics from high-level optimizations of data flow descriptions down to computing platforms that are suitable for implementing data intensive applications. Therefore, the issue should contain interesting reading for audiences with background in high-level modeling, as well as for those with interests in the lower level implementation issues.

Pekka Jääskeläinen (Dr. Tech.) has worked with exposed datapath processor architecture customization and programming since 2002. He has led the development work of the TTA-based Co-design Environment (TCE), a toolset for rapid customization of VLIW-style parallel processors based on the Transport-Triggered Architecture. He received his master's degree in 2005, and doctor's degree in 2012, both from Tampere University of Technology. His current research interests include methods for rapid design of programmable parallel platforms, and techniques for enhancing performance portability of heterogeneous parallel programs.

Jani Boutellier received the M.Sc. and Ph.D. degrees from the University of Oulu, Finland, in 2005 and 2009, respectively. In 2007–2008, 2013, he was working as a visiting researcher with the EPFL, Switzerland. Currently, he is Adjunct Professor at the Computer Science and Engineering Department at University of Oulu. His research interests include design automation, dataflow programming, signal processing, hardware-software codesign and embedded multiprocessing. He is a member of the IEEE Signal Processing Society and an affiliate member of Design and Implementation of Signal Processing Systems Technical Committee.