Architecture, Languages, Compilation and Hardware support for Emerging ManYcore systems (ALCHEMY):
Preface

Johanna Sepúlveda¹, Vania Marangozova-Martin², and Jeronimo Castrillon³

¹ Institute for Security in Information Technology, Technical University of Munich, Germany
johanna.sepulveda@tum.de
² IMAG Grenoble, France
vania.marangozova-martin@imag.fr
³ Technical University of Dresden, Germany
Jeronimo.Castrillon@tu-dresden.de

Abstract
Manycore systems are one of the key enabler technologies for most of current computational paradigms, including Internet of things, data-centers on chip and big data processing. These paradigms are characterized by tight and demanding requirements such as code portability, dynamicity, high performance, usability, predictability, reliability, low power and security.

This combination of requirements has led to heterogeneous manycore systems which are extremely challenging to design and to program. As a result, a large body of research has focused on development of languages, simulation environments and analysis tools that allow to model and predict the behavior of this type of systems from early design stages.

ALCHEMY 2017 presents five research works addressing the challenges of code portability, high performance, usability, security and reliability in manycore systems. These are namely:

1. "An OpenMP backend for the Sigma-C streaming language, addresses the software portability challenge of manycore architectures. It proposes an implementation of an OpenMP backend for the SigmaC language, a cycle-static data flow abstraction to program manycore embedded platforms. Its compilation scheme allows for utilization of future manycore embedded systems such as Kalray’s MPPA.

2. A multi-level optimization strategy to improve the performance of the stencil computation combines manual vectorization, space tiling and stencil composition for achieving high performance of stencil kernels on manycore systems. The evaluation with three compilers (Intel, Clang and GCC) and two target multi-core platforms (Intel Broadwell and Ivybridge) reports better results compared to the state of the art.

3. A Distributed Shared Memory Model and C++ Templated Meta-Programming Interface for the Epiphany RISC Array Processor addresses the usability challenge. It proposes techniques for data layout and parallel loop order abstraction as a parallel programming API targeting the Epiphany architecture. This results into a transparent distributed shared memory (DSM) model for Epiphany that eliminates the need to manage local data movement between cores.
4. *Towards Protected MPSoC Communication for Information Protection against a Malicious NoC* deals with vulnerabilities on Network-on-Chip (NoC). The authors propose a security protocol which allows the secure communication among the cores of the system, even in the presence of Trojan insertions at the NoC whose aim is to modify and steal data.

5. *GPU-Accelerated Real-Time Path Planning and the Predictable Execution Model* addresses the reliability challenge and tackles the important problem of ensuring reliable Worst Case Execution Time for Real-Time and Cyber Physical Systems. While considering heterogeneous (CPU/GPU), the idea is to separate memory and processor operations through Time-Division Multiplexing (TDM).

© 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the International Conference on Computational Science

*Keywords:* Manycore, code portability, high performance, usability, security and reliability.
دریافت فوری
متن کامل مقاله

امکان دانلود نسخه تمام متن مقالات انگلیسی
امکان دانلود نسخه ترجمه شده مقالات
پذیرش سفارش ترجمه تخصصی
امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
امکان دانلود رایگان ۲ صفحه اول هر مقاله
امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
دانلود فوری مقاله پس از پرداخت آنلاین
پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات