Topic 4: High Performance Architectures and Compilers

Koen de Bosschere*, Ayal Zaks*, Michael C. Huang*, and Luis Piñuel*

This topic deals with architecture design and compilation for high performance systems – the discovery and support of parallelism at all levels. The areas of interest range from microprocessors to large-scale parallel machines; from general-purpose platforms to specialized hardware (e.g., graphic coprocessors, low-power embedded systems); and from hardware design to compiler technology. On the compilation side, topics of interest include language aspects, program analysis, transformation, automatic extraction of parallelism at all levels, and the interaction between compiler and the rest of the system. On the architecture side, the scope spans system architectures, processor micro-architecture, memory hierarchy, multi-threading, and the impact of emerging trends.

Out of the 23 paper submitted to this topic, 7 were accepted for presentation at the conference.

The paper "Reducing the Number of Bits in the BTB to Attack the Branch Predictor Hot-Spot" by Noel Tomás, Julio Sahuquillo, Salvador Petit, and Pedro Lopez proposes two techniques to store less data in the Branch Target Buffer (BTB): (i) less tag bits and (ii) less target address bits, in order to tackle the power consumption in the BTB. The authors show that up to 35% of power can be saved without performance loss.

The paper "Low-Cost Adaptive Data Prefetching" by Luis Ramos, Jose Briz, Pablo Ibañez and Victor Viñals explores different prefetch distance-degree combinations and very simple, low-cost adaptive policies on a superscalar core with a high bandwidth, high capacity on-chip memory hierarchy. The authors show that sequential prefetching can be tuned to outperform state-of-the-art hardware data prefetchers and complex filtering mechanisms.

The paper "Stream Scheduling: A Framework to Manage Bulk Operations in Multi-level Memory Hierarchies" by Abhishek Das presents an extension to the Sequoia compiler to schedule data transfers and computation kernels on a parallel computer with multiple levels of memory, in such a way that computation and data transfer overlap and performance is maximized. The authors show a performance that is comparable to the best known (hand-tuned) performance; some compute-intensive applications improve by 15%-35%.

The paper "Interprocedural Speculative Optimization of Memory Accesses to Global Variables" by Lars Gesellensetter presents a register promotion technique for global variables. The authors propose a method to speculatively promote global

* Topic Chairs.

E. Luque, T. Margalef, and D. Benítez (Eds.): Euro-Par 2008, LNCS 5168, pp. 315–316, 2008.

variables to register across function calls by exploiting the Advanced Load Address Table (ALAT) present in Itanium processors.

The paper "Efficiently Building the Gated Single Assignment Form in Codes with Pointers in Modern Optimizing Compilers" by Manuel Arenaz, Pedro Amoedo, and Juan Touriño presents a simple and fast Gated Single Assignment (GSA) construction algorithm that takes advantage of the infrastructure for building the SSA form available in modern optimizing compilers. An implementation on top of the GIMPLE-SSA intermediate representation of GCC is described and evaluated in terms of memory consumption and execution time using the UTDSP, Perfect Club and SPEC CPU2000 benchmark suites.

The paper "Inter-Block Scoreboard Scheduling in a JIT Compiler for VLIW Processors" by Benoit Dupont de Dinechin presents a post-pass instruction scheduling technique suitable for use by Just-In-Time (JIT) compilers targeted to VLIW processors. The technique is implemented in a Common Language Infrastructure JIT compiler for the ST200 VLIW processors and the ARM processors.

The paper "Global Tiling for Communication Minimal Parallelization on Distributed Memory Systems" by Liu Lei presents some strategies to select the matrices for "semi-oblique tiling" considering both iteration space and data space. In addition it proposes a strategy to select among the possible local solutions the optimal ones to achieve a global optimal solution. The experimentations with NPB2.3-serial SP and LU on Qsnet connected cluster achieves an average parallel efficiency of 87% and 73% respectively.