

Topic 7

Parallel Computer Architecture and Instruction Level Parallelism

Eduard Ayguadé, Wolfgang Karl, Koen De Bosschere, and Jean-Francois
Collard

Topic Chairs

We welcome you to the two Parallel Computer Architecture and Instruction Level Parallelism sessions of Euro-Par 2006 conference being held in Dresden, Germany. The call for papers for this Euro-Par topic area sought papers on all hardware/software aspects of parallel computer architecture, processor architecture and microarchitecture. This year 12 papers were submitted to this topic area. Among the submissions, 5 papers were accepted as full papers for the conference (41% acceptance rate).

Three of the accepted papers cover the hardware aspects of this Euro-Par topic. Ro and Gaudiot present and evaluate the design of hierarchically distributed dispatch queues, as an alternative to the traditional centralized dispatch queue. Authors show how their proposal can be designed with small-sized, distributed dispatch queues which consequently can be implemented with low hardware complexity and lead to high clock rates. Rui, Zhang and Hu present and describe the necessary hardware infrastructure on chip multiprocessors to support a hybrid strategy for prefetching that includes dynamic prefetching threads, automatically constructed, triggered, spawn and managed by hardware, and static prefetching threads, statically constructed by a binary-level optimization tool with the guide of profiling information. Finally, De Dios, Sahelices, Ibáñez, Viñals and Llabería attack in their paper one of the major performance bottlenecks in parallel programs: synchronization. Authors present and show an inexpensive implementation of a novel hardware mechanism, named Request Bypass, to speed-up lock-based synchronizations in DSM multiprocessors.

The two other papers are related with code generation and architecture simulation. Bednarski and Kessler evaluate and compare two methods for optimal integrated VLIW code generation that fully integrate all steps of code generation (instruction selection, register allocation and instruction scheduling). The techniques are based on integer linear programming and dynamic programming, both previously proposed by the same authors. Colmenar, Garnica, Lanchares, Hidalgo and Miñana present an architectural simulator able to model asynchronous superscalar architectures, with the aim of studying different architectural proposals for asynchronous processors. The novelty resides in the use of distribution functions to describe the probability of delays.

We are grateful to our referees for lending us their expertise and providing rigorous reviews. We hope that this collection of papers will prove to be inter-