MEmory performance: DEaling with Applications, systems and architecture

S. Bartolini*, P. Foglia+, R. Giorgi*, C.A. Prete+

*Department of Information Engineering, University of Siena, Siena, Italy
{bartolini,giorgi}@dii.unisi.it
+ Department of Information Engineering, University of Pisa, Pisa, Italy
{foglia prete}@iet.unipi.it

Members of the HiPEAC[1]

In this issue of ACM SigArch Newsletter, we present ten papers from MEDEA-2005 Workshop held in conjunction with the IEEE/ACM International Conference on Parallel Architectures and Compilation Techniques (PACT-2004) [1], [2].

Due to the ever-increasing gap between CPU and memory speed, the interest in evaluating and proposing architectures dealing with "memory wall" problems is still alive. In particular, the problem of hiding/tolerating memory latencies is exacerbated by wire-delay and power consumptions issues. Memory is far slower than processor, and its access is becoming even slower due to wire delay, which is not anymore negligible also in caches, especially large and high performance ones. On the other side, the need of maintaining low the overall chip energy dissipation pushes towards the reconsideration of the whole memory subsystem architecture. One of the most interesting approaches is based on decoupling resources, both on-chip and off-chip. In addition, modular high-level design is needed to reduce design costs.

In this scenario, system design issues should be addressed taking into account the relationship between the system architecture and the run-time behavior of the particular application domain. In fact, it is the interaction between the static/dynamic features of the application and the system on which it executes that stresses the memory subsystem and pushes towards specific solutions.

The presented papers can be split into three general areas:

1) *Caching in general purpose and embedded system*, in which A. Friedman et al., examine the effectiveness of write-back cache policies in decreasing write traffic towards memory for garbage collection applications; Ramaswamy et al. propose a data trace-cache design to exploit the locality of reference of tree data structures, which are responsible for a significant component of the memory traffic in several applications. Kavi et al. show that their previously proposed split data cache can also be advantageous in embedded applications; W. Zhang, et al. propose to make use of replication cache in order to improve the performance of multiple-issue superscalar microprocessors by improving the cache read bandwidth.

2) *Multiprocessors and network processors*, in which Monchiero, Silvano et al. explore optimization techniques for synchronization mechanisms in MPSoCs relying on complex interconnection (Network-on-Chip), targeted to future mobile systems. Khunjush et al. introduce architectural extensions comprising specialized network caches and instructions. The objective is to reduce the copying overhead needed to transfer and bind a received message to the consuming process/thread in message passing environments.

[1] HiPEAC is a Network of Excellence funded by European Union; the acronym stands for High Perfmance Embedded-system Architecture and Compilation.

3) *Energy, memory bandwidth and system optimizations*, in which Yue et al. present a benchmark suite specifically designed to evaluate cryptographic performance on Network Processors, focusing mainly on processor and memory subsystem. López-Lagunas et al. advocate the use of stream descriptors to keep processors busy in applications for computer vision and image processing. Chiyonobu and Sato propose a new instruction scheduling method utilizing cache miss information in a processor architecture where only critical instructions are executed on power-hungry functional units. In this way, the total energy consumption can be reduced without severe performance degradation. Bardine et. al. present a methodology for a system-level design space exploration leveraging on fast, high level simulation, and a case study based on a H.264 video coder.

We wish to thank the MEDEA Program Committee (Erik Altman, Fumio Arakawa,Alessio Bechini, Mats Brorsson, Ali Hurson, David Kaeli, Krishna Kavi, Stephen Keckler, Hiroaki Kobayashi, David M. Koppelman, Sally McKee, Enrico Martinelli, Alexander Milencovich, Veljko Milutinovic, Sanjay Patel, Toshinori Sato, Naohiko Shimizu, Alan J. Smith, Mateo Valero, Theo Ungerer, Stamatis Vassiliadis, Wei Zhang) for their excellent work in disclosing the workshop, referring activity and suggestions to increase the quality of the accepted papers. We want also to thank T. Cucinotta, A. Naz, D. Vercelli, R. Canal and P. Bennati for their help in the reviewing process.

References:

[1] ACM-IEEE International Conference on Parallel Architectures and Compilation Techniques (PACT) 2005, http://pact05.ce.ucsc.edu/, Saint Louis, Missouri (USA), September 17-21, 2005.
[2] PACT Conferences, http://www.pactconf.org.
[3] MeDEA-2005 Workshop, http://garga.iet.unipi.it/medea05/.