IEEE RTSI 2018 – Track 2 – 2.11

COMPUTING IN HETEROGENEOUS, AUTONOMOUS 'N' GOAL-ORIENTED ENVIRONMENTS

As the push for parallelism continues to increase the number of cores on a chip, system design has become incredibly complex; optimizing for performance and power efficiency is now nearly impossible for the application programmer. To assist the programmer, a variety of techniques for optimizing performance and power at runtime have been developed, but many employ the use of speculative threads or performance counters. These approaches result in stolen cycles, or the use of an extra core, and such expensive penalties can greatly reduce the potential gains.

Within this context imagine a revolutionary computing system that can observe its own execution and optimize its behavior around a user's or application's needs. Imagine a programming capability by which users can specify their desired goals rather than how to perform a task, along with constraints in terms of an energy budget, a time constraint, or simply a preference for an approximate answer over an exact answer. Imagine further a computing system that performs better according to a user's preferred goal the longer it runs an application. Such an architecture will enable, for example, a handheld radio or a cell phone that can run cooler the longer the connection time. Or, a system that can perform reliably and continuously in a range of environments by tolerating hard and transient failures through self healing.

Self-aware computer systems are the key technology to succeed in doing this. They will be able to configure, heal, optimize, improve interaction and protect themselves without the need for human intervention, exploiting abilities that allow them to automatically find the best way to accomplish a given goal with the resources at hand. Within this context, imagine a revolutionary computing system that can observe its own execution and optimize its behavior around the external environment, user's and application's needs. The Self-Aware computing research leverages the new balance of resources to improve performance, utilization, reliability and programmability. Within this context, the proposed workshop is intended to present innovative works describing:

- Self-aware Operating Systems

- Autonomous self-aware computer architecture

- Adaptive algorithm and distributed self-training algorithms

- Biologically inspired systems

Marco Domenico Santambrogio

Politecnico di Milano, Italy

marco.santambrogio@polimi.it

Short Curriculum Vitae

Dr. Marco Domenico Santambrogio is an Assistant professor at Politecnico di Milano. He received his laurea (M.Sc. equivalent) degree in Computer Engineering from the Politecnico di Milano (2004), his second M. Sc. degree in Computer Science from the University of Illinois at Chicago (UIC) in 2005 and his PhD degree in Computer Engineering from the Politecnico di Milano (2008). Marco, since 2009, is an adjunct professor at the University of Illinois at Chicago and he has been an MIT Research Affiliate (2010-2015). Dr. Santambrogio was, in 2009-2010, a postdoc fellow at CSAIL, MIT, and he has also held visiting positions at the Department of Electrical Engineering and Computer Science of the Northwestern University (2006 and 2007) and Heinz Nixdorf Institut (2006).
Marco D. Santambrogio is a senior member of the IEEE (since 2011) and ACM (since 2013). Since 2013 he is a member of the IEEE Italy Section as Student Activities Coordinator, and starting in 2015 he is Vice-Chairs of the IEEE Italian Chapter of the Computer Society.
He has been with the NECST Lab at the Politecnico di Milano, where he founded the Dynamic Reconfigurability in Embedded System Design (DRESD) project in 2004 and the CHANGE (self-adaptive computing system) project in 2010. He conducts research and teaches in the areas of reconfigurable computing, self-aware and autonomic systems, hardware/software co-design, embedded systems, and high performance processors and systems.

Henry Hoffmann

University of Chicago, IL, USA

hankhoffmann@uchicago.edu

Henry Hoffmann has been an Assistant Professor in the Department of Computer Science at the University of Chicago since January 2013 where he leads the Self-aware computing group (or SEEC project) and conducts research on adaptive techniques for power, energy, and performance management in computing systems.

He has spent the last 13 years working on multicore architectures and system software in both academia and industry. He completed a PhD in Electrical Engineering and Computer Science at MIT where his research on self-aware computing was named one of the ten "World Changing Ideas" by Scientific American in December 2011. He received his SM degree in Electrical Engineering and Computer Science from MIT in 2003. As a Masters student he worked on MIT's Raw processor, one of the first multicores.

Along with other members of the Raw team, he spent several years at Tilera Corporation, a startup which commercialized the Raw architecture and created one of the first manycores. His implementation of the BDTI Communications Benchmark (OFDM) on Tilera's 64-core TILE64 processor still has the highest certified performance of any programmable processor.

Prior to his graduate studies, he served as an Associate Staff member of MIT Lincoln Laboratory where his research produced the Parallel Vector Library (PVL), which forms the foundation of the VSIPL++ standard, an interface for parallel signal and image processing. Henry was appointed as a Lincoln Masters Scholar in 2001 and a Lincoln Doctoral Scholar in 2004. In 1999, he received his BS in Mathematical Sciences with highest honors and highest distinction from UNC Chapel Hill.