

**Program for the Workshop on Theory and Many-Core (T&MC), May 29, 2009**

Kim Engineering Building, Room 1110

University of Maryland, College Park, Maryland

8:30 Light breakfast

9:00 Welcome and brief introduction

9:15 Parallel Algorithm Issues in TBB, Arch Robison, Intel

10:00 Coffee Break

10:30 A Bridging Model for Multi-Core Computing, Leslie Valiant, Harvard University

11:15 Theory: Asleep at the Switch to Many-Core, Phillip B. Gibbons (Intel Research Pittsburgh)

12:00 Lunch

1:30 Opportunities in Many Core Algorithms, Guy Blelloch, Carnegie-Mellon University

2:15 The PRAM-On-Chip Proof-of-Concept, Uzi Vishkin, University of Maryland

3:00 Coffee break

3:30 Optimal Speedup on a Low-Degree Multi-Core Parallel Architecture (LoPRAM), Reza Dorrigiv, Alejandro López-Ortiz, and Alejandro Salinger, University of Waterloo

3:40 Algorithm design for multicore processors, Theory and applications, Peter Krusche and Alexander Tiskin, The University of Warwick.

3:50 A Model of Computation for Map Reduce, Howard Karloff, AT&T Labs, Siddharth Suri, and Sergei Vassilvitskii, Yahoo! Research

4:00 Parallel Phase Model for Manycore Parallel Machines, Zhaofang Wen, Sandia National Labs, and Junfeng Wu, Syracuse University

4:10 Parallel External Memory Model, Lars Arge, University of Aarhus, Michael Goodrich, and Nodari Sitchinava, University of California - Irvine

4:20 Short break

4:30 Panel discussion: What will be the role of parallel algorithmic thinking in getting many core computing to stable state, and once it gets there? What will be the similarities and differences in the role that parallel algorithms will play for parallel programming relative to serial algorithms and serial programming? What are the key algorithmic/theory questions to address? What will become (or already are) the theory "classics" worth teaching for the long term and what are temporary artifacts of day or obsolete assumptions? How can we get reignite interest in parallelism in the theory community?