Keynote: Cache-Aware Scheduling and Analysis for Multicores

Prof. Wang Yi, Uppsala University, Sweden

About the keynote speaker

Wang Yi received his Ph.D. in computer science from Chalmers University of Technology, Sweden in 1991. He is a chair professor (Embedded Systems) at Uppsala University, and a professor in computer science at North Eastern University, China. He has been a program chair of TACAS, FORMATS, EMSOFT and HSCC, and a track chair of RTSS. He is or has been an editorial board member of Elsevier Journal of computer architectures, the journal of Computer Science and Technology, and IEEE transactions on computers.

His research interests include modeling and verification of real-time and embedded systems. He initiated (1993) and co-founded the UPPAAL tool (1995) and also the company UP4ALL (2007). He also co-founded the TIMES tool for scheduling and analysis of timed systems. His current interests are in the development of embedded and real-time applications on multi-core processors. He is directing the Swedish strategic research program CoDeR-MP: Computationally Demanding Real-Time Applications on Multi-core Platforms, in collaboration with ABB and SAAB. He is also a principle investigator of UPMARC: Uppsala Programming for Multi-core Architectures Research Center.

Summary:

The major obstacle to use multicores for real-time applications is that we may not predict and provide any guarantee on real-time properties of embedded software on such platforms; the way of handling the on-chip shared resources such as L2 cache may have a significant impact on the timing predictability. In this talk, we propose to use cache space isolation techniques to avoid cache contention for hard real-time tasks running on multicores with shared caches. We present a scheduling strategy for real-time tasks with both timing and cache space constraints, which allows each task to use a fixed number of cache partitions, and makes sure that at any time a cache partition is occupied by at most one running task. In this way, the cache spaces of tasks are isolated at run-time. As technical contributions, we present solutions for the scheduling analysis problem. For simplicity, the presentation will focus on non-preemptive fixed-priority scheduling. However, our techniques can be easily adapted to deal with other scheduling strategies like EDF. We have developed a sufficient schedulability test for non-preemptive fixed-priority scheduling for multicores with shared L2 cache, encoded as a linear programming problem. To improve the scalability of the test, we then develop our second schedulability test of quadratic complexity, which is an over approximation of the first test. To evaluate the performance and scalability of our techniques, we use randomly generated task sets. Our experiments show that the first test which employs an LP solver can easily handle task sets with thousands of tasks in minutes using a desktop computer. It is also shown that the second test is comparable with the first one in terms of precision, but scales much better due to its low complexity, and is therefore a good candidate for efficient schedulability tests in the design loop for embedded systems or as an on-line test for admission control.