Abstract:
Many applications exhibit the property that small, localized changes to the input result in only minor changes to the output. Incremental computations seek to exploit this observation to speed-up successive runs of an application by tracking control and data dependencies, and re-executing only those parts of the computation that are affected by a given input change. To realize the benefits of incremental computation, researchers and practitioners are developing new systems where the programmer can provide efficient update mechanisms for changing application data. Unfortunately, most of the existing solutions are limiting because they either depart from existing programming models, or require programmers to devise an incremental update mechanism (or a dynamic algorithm) on a per-application basis.

The high-level goal of my work has been to enable practical, automatic, and efficient incremental computations in parallel and distributed systems. My approach neither requires a radical departure from current models of programming nor complex, application-specific dynamic algorithms.

In this talk, I will first present a high-level description of my research work. Thereafter, as a concrete example I will touch upon two projects: iThreads. iThreads is a POSIX-compliant threading library to support parallel incremental computation targeting unmodified C/C++ multithreaded programs.

Lastly, I will conclude the talk with a brief overview of my on-going projects.

Host: Yongluan Zhou