Distributed Cooperation in Dynamically Changing Groups

Speaker: Chryssis Georgiou

Abstract:

The problem of cooperatively performing a set of t tasks in a decentralized computing environment subject to failures is one of the fundamental problems in distributed computing. The setting with partitionable networks is especially challenging, as algorithmic solutions must accommodate the possibility that groups of communicating processors become disconnected (and, perhaps, reconnected) during the computation. The efficiency of task-performing algorithms is often assessed in terms of work: the total number of tasks, counting multiplicities, performed by all of the processors during the computation. In general, the scenario where the processors are partitioned into g disconnected components causes any task-performing algorithm to have work Omega (tg) even if each group of processors performs no more than the optimal number of Theta(t) tasks.

Given that such pessimistic lower bounds apply to any scheduling algorithm, we pursue a competitive analysis. Specifically, in the talk I will present a simple randomized scheduling algorithm for p asynchronous processors, and compare its performance to that of an omniscient off-line algorithmwith full knowledge of the future changes in the communication medium. I will describe the notion of computation width, which associates a natural number with a history of changes in the communication medium, and I will show both upper and lower bounds on work-competitiveness in terms of this quantity. Specifically, we will see that the simple randomized algorithm obtains the competitive ratio (1+ cw / e), where cw is the computation width and e is the base of the natural logarithm (e=2.7182...); this competitive ratio is tight. Finally, I will explain how the algorithm could be implemented using a Group Communication Service implementation while maintaining its work-competitiveness.

Joint work with Alex A. Shvartsman and Alexander Russell.