Combinatorial Structures for Bonded-Memory Computing

**Goals**: Characterize computability in shared memory models of bounded capacity using tools of combinatorial topology.

**Tools**: Logic, mathematics, algorithmic reasoning.

**Prerequisites**: Maturity in math and algorithms, curiosity and rigor.

Practically all computing systems, from fire alarms to Internet-scale services, are nowadays distributed: they consist of a number of computing units performing independent computations and communicating with each other to synchronize their activities. Our dependence on performance and reliability of the distributed computing becomes more and more imminent. Therefore, understanding fundamentals of distributed computing is of crucial importance.

The main complication here is the existing immense diversity of distributed applications, models of distributed computations, and performance metrics, combined with the lack of mathematical tools to handle this complexity.

Recently, an impressive attempt to address this challenge was made: some long-standing open questions in distributed computability were resolved using advanced branches of modern mathematics, such as combinatorial and algebraic topology. More precisely, a set of possible concurrent executions can be represented as a geometrical structure, called simplicial complex, and all possible ways the concurrent system can evolve can be seen as a transformation of the simplicial complex in space.

For example, it turns out that the simplicial complex modelling the reachable states of a wait-free system (imposing neither synchrony assumptions nor bounds on the number of failures) is always contractible (connected in all dimensions) and, thus, there is no way to solve non-trivial set agreement (imposing an odd number of “holes”) [1, 4, 5, 7]. More generally, task computability in read-write shared-memory systems has been characterized via the celebrated Asynchronous Computability Theorem (ACT) [3, 5], relating the ability if solving a task with the existence of a specific simplicial map from the task’s input simplicial complex to the task’s output simplicial complex.

ACT and its more recent generalizations [2, 6] implicitly assumes unbounded shared-memory distributed systems. Indeed, these characterizations are based on full-information protocols in which processes use the shared memory to exchange complete information about their states, which might require no bounds on both the local and shared memory. In this project, we intend to characterize bounded computations. This may potentially give rise to new time and space complexity bounds for shared-memory algorithms.
Contact
Prof. Petr Kuznetsov
http://www.infres.enst.fr/~kuznetso/
petr.kuznetsov@telecom-paristech.fr
INFRES, Télécom ParisTech
Office C213-2, 46 Rue Barrault

References


