Self-Oscillations in Population Protocols

Anissa Lamani
Department of Informatics, Kyushu University, Japan
Joint work with Masafumi Yamashita and Yukiko Yamauchi

The PP model was introduced by Angluin at al. to model passive distributed systems, in which a collection of finite-state agents interact with each other in order to accomplish a common task. The agents are assumed to be identical and uniform i.e., they are not identified and all execute the same protocol. The computations are performed through pairwise interactions i.e., when two agents interact, they exchange their local information and update their state according to a common protocol. The interaction pattern is unpredictable i.e., the agents have no control on which agent they will interact with.

Many investigations have considered the PP model, different problems have been addressed as computing a function, electing a leader, counting, coloring and naming [1, 2, 3, 4, 6, 8]. Most of these problems are concerned with the convergence to a specific configuration that contains the answer to the problem that is considered. These problems are hence said to be static. Only few investigations have considered dynamic problems such as the self-stabilizing token circulation problem on rings [4], the self-stabilizing mutual exclusion and the group mutual exclusion problems [5] and recently the self-stabilizing oscillation problem [7].

About the self-stabilizing oscillation problem, it has been shown that under a deterministic scheduler, the self-stabilizing leader election (SS-LE) and the self-stabilizing oscillation problem (SS-OSC) are equivalent [7], in the sense that an SS-OSC protocol is constructible from a given SS-LE protocol and vice versa, which unfortunately implies that (1) resorting to a leader is inevitable (although we seek a decentralized solution) and (2) n states are necessary to create an oscillation of amplitude n, where n is the number of agents (although we seek a memory-efficient solution).

Under the probabilistic scheduler, we investigate both the self-stabilizing synchronization problem and the self-stabilizing oscillation problem. We then aim at designing a PP that represent any given periodic function f by a non empty set of populations that exhibit an oscillatory behavior.

References

[1] Dana Angluin, James Aspnes, Melody Chan, Michael J. Fischer, Hong Jiang, and René Peralta. Stably computable properties of network graphs. In *DCOSS*, volume 3560, pages 63–74, 2005.

[2] Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. In *PODC*, pages 290–299, 2004.

[3] Dana Angluin, James Aspnes, and David Eisenstat. Fast computation by population protocols with a leader. *Distributed Computing*, 21(3):183–199, 2008.

[4] Dana Angluin, James Aspnes, Michael J. Fischer, and Hong Jiang. Self-stabilizing population protocols. *TAAS*, 3(4), 2008.

[5] Joffroy Beauquier and Janna Burman. Self-stabilizing mutual exclusion and group mutual exclusion for population protocols with covering. In *OPODIS*, volume 7109, pages 235–250, 2011.

[6] Shukai Cai, Taisuke Izumi, and Koichi Wada. How to prove impossibility under global fairness: On space complexity of self-stabilizing leader election on a population protocol model. *Theory Comput. Syst.*, 50(3):433–445, 2012.

[7] Colin Cooper, Anissa Lamani, Giovanni Vigiletta, Masafumi Yamashita, and Yukiko Yamauchi. Constructing self-stabilizing oscillators in population protocols. In *Stabilization, Safety, and Security of Distributed Systems, (SSS)*, volume 9212, pages 187–200, 2015.

[8] Keigo Kinpara, Tomoko Izumi, Taisuke Izumi, and Koichi Wada. Improving space complexity of self-stabilizing counting on mobile sensor networks. In *OPODIS*, volume 6490, pages 504–515, 2010.