Towards Efficient Verification of Population Protocols

Stefan Jaax, Michael Blondin, Javier Esparza, and Philipp J. Meyer

Technische Universität München
{jaax, blondin, esparza, meyerphi}@in.tum.de

Description.

Population protocols are a model of computation by anonymous, identical finite state agents. Verification of population protocols is decidable, but EXPSPACE-hard. We introduce a fully expressive class of population protocols that is suitable for efficient verification.

Abstract.

This talk introduces the first efficient approach to parameterized verification of population protocols. Population protocols (Angluin et al., PODC, 2004) are a model of distributed computation by many anonymous finite-state agents. They were initially introduced to model networks of passively mobile sensors, but are now also used to describe chemical reaction networks. In each computation step of a population protocol, a fixed number of agents are chosen nondeterministically, and their states are updated according to a joint transition function. Since agents are anonymous and identical, the global state of a protocol is completely determined by the number of agents at each local state, called a configuration. A protocol computes a boolean value b for a given initial configuration C_0 if in all fair executions starting at C_0, all agents eventually agree to b — so, intuitively, population protocols compute by reaching consensus under a certain fairness condition.

Designing correct population protocols is a difficult task. Consequently, there is a need for automatic verification of population protocols. However, deciding whether a protocol computes a given predicate is hard, as the number of agents in a population is fixed, but arbitrarily large. Earlier approaches only attacked the verification problem for populations up to a constant size. In our talk we present a truly *parametric* approach, which is able to prove correctness for *all* of the infinitely many initial configurations: We introduce a class of population protocols that is suitable for automatic verification, while preserving the expressive power of general well-specified protocols. Moreover, the membership test for our class is of moderate complexity, as it reduces to solving boolean combinations of polynomially bounded linear constraints over the natural numbers.

We implemented our decision procedure on top of the constraint solver Z3. We present benchmarks showing how our approach has been successfully applied to standard protocols from the literature: Our tool proves correctness for all inputs of protocols with up to 20 states in less than one second, and protocols with 70 states and 2500 transitions in less than one hour. In particular, we can automatically prove correctness for *all* inputs in less time than previous tools needed to check *one single large input*.