SAT-based minimization of omega-automata
(Research Master Intership)

Alexandre Duret-Lutz (LRDE/EPITA) adl@lrde.epita.fr
Souheib Baarir (LIP6/UPMC) souheib.baarir@lip6.fr

Deterministic Büchi automata (DBA) are used in several domains like probabilistic model checking or synthesis of distributed systems. DBAs are commonly obtained from Büchi automata (BA) by applying Safra's construction or, when it works, some powerset-based construction. Since applications are usually more efficient when dealing with small automata, it makes sense to try to minimize these DBA (i.e., find an equivalent DBA such that no smaller equivalent DBA exist).

Among other things, the Spot library (`http://spot.lip6.fr/`) offers numerous algorithms to simplify Büchi automata. A recently added algorithm (available in Spot 1.2) uses a SAT-based approach to minimize DBA.

To minimize an n-state DBA, an NP-complete problem, we encode its equivalence with a deterministic $(n-1)$-state automaton as a SAT problem. If the problem is satisfiable, it means there exists an equivalent automaton with $n-1$ states or less. The procedure can therefore be repeated to check for an equivalent $(n-2)$-state automaton, and so on…

Our encoding also works for a generalized variant of DBA called Deterministic Transition-based Generalized Büchi Automata (DTGBA).

It would be perfectly sensible to imagine a similar procedure to minimize deterministic Rabin automata (DRA). It even makes more sense, as any omega-regular property can be expressed by a DRA, while this is not the case with DBA or DTGBA.

The objectives of this internship are:

- to optimize the actual SAT encoding of the DBA minimization by using the structural characteristics of the automaton,
- to generalize the approach to deterministic Rabin automata,
- to optimize the implementation of these procedures using advanced SAT-based techniques such as incremental SAT-solving.

This internship targets people:

- who want to delve the wonderful world of SAT-problems and omega-automata,
- are familiar with C++ and Unix development.

References:

- On the use of SAT for DBA minimization:
  - `http://www.react.uni-saarland.de/publications/sat2010.pdf`
  - `http://www.react.uni-saarland.de/people/ehlers/slides_sat2010.pdf`

  (Although the technique implemented in Spot is more evolved, it is inspired by this one.)
- On the use of SAT in Spot: `http://spot.lip6.fr/userdoc/satmin.html`