**PhD subject**

Reachability in Petri Nets

**Supervisors**

- **Alain Finkel**
  *Téléphone*: Alain Finkel : 01 47 40 75 69
  *Email*: finkel@lsv.ens-cachan.fr

- **Serge Haddad**
  *Téléphone*: 01 47 40 75 24
  *Email*: haddad@lsv.ens-cachan.fr

**Laboratoire Spécification et Vérification**

*Ecole Normale Supérieure de Cachan*

61, avenue du Président Wilson
94235 Cachan CEDEX

**Key words:**

Reachability, verification, decidability, logic, well structured transition systems, efficient algorithms.

**General Context**

The theory of *Well Structured Transition Systems*, (WSTS) allows the automatical verification of safety properties of infinite-state systems, such that parts of reachability sets can be finitely represented [7, 11, 10]. Termination, boundedness and coverability are decidable for WSTS [4, 5, 9].

As Petri nets are WSTS, the previous properties are decidable.

For complete WSTS [10], the Karp and Miller procedure [13, 10] computes the finite set of maximal elements of the downward closure of the reachability set. This procedure logs a state space exploration of the reachability set with a finite tree allowing to decide some other reachability problems like the recurrent control-state reachability problem. The class of very-WSTS in which this procedure terminates has been determined very recently in [2] and, still, Petri nets are very-WSTS. When the Ideal Karp Miller algorithm terminates, LTL is decidable on very-WSTS under natural but new effective conditions that are also verified on Petri nets [2].

**Objective:**

The first main objective is to construct an efficient coverability graph algorithm and to construct an efficient LTL model checker for Petri nets.
The second main objective is to construct an efficient reachability tool for Petri nets. Surprisingly, there does not exist such a tool for solving this fundamental problem. Even if the decidability of reachability for Petri nets is known since 1981, the different known algorithms [16, 14, 15] have not been yet implemented.

**Objective 1 : construct an efficient LTL model checker for Petri nets.**

1. Analyse the three following minimal coverability algorithms of Gilles Geeraerts and Jean-François Raskin and Laurent Van Begin in [12], of Pierre-Alain Reynier and Frédéric Servais in [18], and of Artturi Piipponen and Antti Valmari in [17].
2. Compare these three different coverability algorithms.
3. Compare the three different tools.
4. From the previous survey on existing algorithms, construct an efficient implementation of the minimal coverability graph algorithm based on the original minimal coverability set procedure [8].
5. Extend the decidability of LTL to temporal logics beyond LTL (see, for instance, bounded Model Checking on WSTS [6]).

**Objective 2 : construct an efficient reachability tool for Petri nets.**

Design an heuristic that begins with the most faster non-reachability algorithms: on the positive rationals, the integers, compute over-approximations (Q-COVER),...

1. Define a logics, called cov-reach, allowing to express both coverability and reachability properties. Define the cov-reachability set/tree/graph associated with such formula.
2. Make a polynomial cov-reach algorithm for Petri nets ruling on positive rationals [3].
3. Make a non-deterministic polynomial cov-reach algorithm for Petri nets on integers.
4. Use the minimal coverability tool (built in Objective 1) to decide whether the downward approximation of the reachability set is sufficient to conclude the non-coverability (hence the non-reachability).
5. Construct an algorithm deciding whether there are less than two simultaneous unbounded counters on each reachability path. Then extend the result in [1] for computing the reachability set, and for computing the cov-reachability set.
6. Combine pre and post computations.

8. Make an efficient prototype with case studies.

9. Determine classes of Petri nets on which the cov-reach algorithm is efficient.

Location

This PhD will be supervised at the Ecole Normale Supérieure Paris-Saclay.

Qualifications and Connections

This PhD is opened to strongly motivated and excellent Master students who like discrete mathematics, theoretical computer science and algorithms.

Ideally, the candidate holds a Master degree in Computer Science (with courses in formal verification, theoretical computer science and mathematical structures for CS) or equivalently is graduated from a Computer Science Engineering School with a strong background in theoretical computer science.

This research program is directly connected to [MPRI C2-9 course] on Mathematical foundations of the theory of infinite transition systems. It should suit a theoretically-minded student with some taste for theoretical and algorithmic constructions. The internship is an ideal opportunity for starting a PhD thesis (possible collaborations with Bordeaux and Montréal).

Références


