On the Expressive Power of Timed Automata and Time Petri Nets

B. Bérard
LAMSADE, Paris, France
E-mail: berard@lamsade.dauphine.fr

Abstract

This presentation is based on joint work with F. Cassez*, S. Haddad**, D. Lime* and O. H. Roux* (*IRCCyN, Nantes and **LSV, Cachan), published in the proceedings of ATVA’05, CONCUR’05, FSTTCS’05, with an extended version of the third paper to appear in Theoretical Computer Science. The aim of this work was to compare the expressive power of Timed Automata (TA) and Time Petri Nets (TPN).

**Time Petri Nets.** Adding explicit time to classical models of dynamic systems was first done in the seventies for Petri nets [17,19], with the aim to verify quantitative properties of systems. Since then, various timed models based on Petri nets were proposed. Among them, the two most prominent ones are Timed Petri Nets (TdPN) [19,2,1,10] and Time Petri Nets (TPN) [17,8], which we consider here. In TPNs, a time interval is associated with each transition and a transition can fire if its enabling duration belongs to its interval. Since time elapsing must not disable transitions, TPNs naturally model urgency requirements. Efficient verification methods have been designed for bounded TPNs and implemented in several tools like ROMEO or TINA [13,9].

**Timed Automata.** Timed Automata (TA), introduced in the seminal paper [4] have yielded a significant breakthrough in the theory of modelling and analysis of timed systems. The most commonly used variant of TA, called Safety TA has been defined in [15] and is the one considered here (and called TA for brevity). A TA is a finite automaton equipped with a set of clocks which evolve synchronously with time. Elementary constraints on clock values restrict the time spent in a location and the firing of transitions. In addition, transitions may involve some clock reset. In this model, verification is based on a finite partition of clock values [12], leading to a so-called region automaton [4], and is supported by various tools, for instance KRONOS or UPPAAL [20,18]. Many extensions of TA have also been proposed, often leading to undecidability for verification problems.

**Comparing expressiveness of timed models.** Due to the diversity of time mechanisms involved in these models, comparing their expressiveness, with respect to some equivalence relation, is a natural issue. A first interesting equivalence is the weak timed bisimilarity of the models. Recall that there are unbounded TPNs for which no bisimilar TA exists. This is a direct consequence
of the following observation: the untimed language of a TA is regular which is not necessarily the case for TPNs. It was proved in [11] that bounded TPNs form a strict subclass of the class of timed automata, in the sense that for each bounded TPN \( \mathcal{N} \), there exists a TA which is weakly timed bisimilar to \( \mathcal{N} \) but the converse is false. A similar result can be found in [16], where it is obtained by a completely different approach. On the other hand, we prove in [6] that there are TA for which no bisimilar TPN exists. This strict inclusion result leads to investigate the relations between these models in two directions: 1) considering the equivalence based on timed language acceptance and 2) characterizing the maximal subclass \( TA^{wtb} \) of timed automata which admit a weakly timed bisimilar TPN.

Language equivalence. In [14], the authors compare Timed State Machines (TSM, a restricted version of TA) and TPNs, giving a translation from TSM to TPN that preserves timed languages. In [6], we have designed a more general translation between TA and TPNs with better complexity. As a consequence of these results, we obtain that TA, bounded TPNs and 1-safe TPNs are equally expressive with respect to timed language acceptance.

Weak timed bisimilarity. We first noticed that the memory policy of the original model selects what we call intermediate semantics (I): time is reinitialised for the transitions disabled after consuming the tokens, as well as for the firing transition. Since this semantics is not appropriate for some applications, we proposed in [5] two alternative semantics: the atomic (A) and the persistent atomic (PA) ones, and we compared the expressiveness of these three semantics for weak timed bisimilarity. We first prove the inclusion result \( (I) \subseteq (A) \subseteq (PA) \), with some examples for strict inclusion. However, we show that for bounded TPNs with upper-closed intervals, the three semantics are equivalent.

We then investigate the second question: given a TA, can we decide whether there exists a bisimilar TPN. First note that we obtain in [6] a syntactical subclass of TA which is as expressive as bounded and 1-safe TPNs. In [7], using persistent atomic semantics, we give a characterization of the maximal subclass \( TA^{wtb} \) of timed automata which admit a weakly timed bisimilar TPN. The condition is not intuitive and relates to the topological properties of the region automaton associated with a TA. To prove that the condition is necessary, we introduce the notion of uniform bisimilarity, which is stronger than weak timed bisimilarity. Conversely, when the condition holds for a TA, we provide two effective constructions of bisimilar TPNs: the first one with rational constants has a size linear w.r.t. the TA, while the other one, which uses only integer constants has an exponential size. From this characterization, adapting a proof in [3], we deduce that given a TA, the problem of deciding whether there is a TPN bisimilar to it, is PSPACE-complete. Thus, we obtain that the membership problem is PSPACE-complete. Finally we also prove that the reachability problem is PSPACE-complete.
References


