Partial Order Based Model Checking with Data Types

HANS FLEISCHHACK & ELISABETH PELZ

Since 20 years, model checking has been developed as a method for verification of logical properties of concurrent systems Σ. To avoid unnecessary exponential explosion caused by interleaving semantics Esparza invented a model checking with partial order semantics of Σ (cf. [1]). The method is based on computing a complete finite prefix $\beta_{ll}(\Sigma_{ll})$ of the maximal branching process of Σ, where Σ usually is given by a finite low level (ll) Petri net Σ_{ll}.

For the specification of 'real world' problems which may require the use of large—possibly infinite—data types, high level (hl) Petri nets (Σ_{hl})have been introduced and offer a compact representation. Model checking of these systems unfortunately requires preliminary unfolding $\Sigma_{hl} \to \Sigma_{ll}$, generating possibly exponentially larger or even infinite nets.

We propose a substantial improvement of this approach by calculating

(i) a finite prefix β_{hl} directly from the the hl-model Σ_{hl};

(ii) using a symbolic representation of data in the –now high level– maximal branching process;

(iii) and by extending slightly Esparza's logic $\mathcal{L}$ by atomic formulas of the form $v \in A$, where v denotes a value and A a set as data type.

All three aspects are new in this area of research.

More formally, let t be an event of Σ_{hl} labeled by a formula $\phi(y_1, ..., y_n) = \phi(\overline{y})$, M a state, and $A_i = type(y_i)$, for $1 \leq i \leq n$.

Usually t is unfolded into as many copies as there are assignments for $y_1, ..., y_n$ satisfying ϕ at M, i.e. for $P = \{\overline{v} \in A_1 \times ... \times A_n \mid \phi(\overline{v}) \text{ is satisfied at } M\}$, we have $|P|$ copies of t in $\beta_{ll}(\Sigma_{ll})$. Moreover, the representation of local data around t needs $|A_1 \times ... \times A_n|$ places in $\beta_{ll}(\Sigma_{ll})$.

In contrast, our algorithm for β_{hl} calculates (along ϕ) a partition of P in k disjoint cartesian products of the form $(A_1^1 \times ... \times A_n^1) \uplus ... \uplus (A_1^k \times ... \times A_n^k)$, and such that k is minimal, with each $A_i^j \subset A_i$. Thus β_{hl} only needs k copies of t and at most $n \times k$ places for the representation of local data. (In the worst case, each A_i^j is a singleton and we obtain $\beta_{hl} = \beta_{ll}$). For a lot of 'real world' examples with large or infinite $|P|$ we obtain very small partitions ($k < 10$) and thus very small models.

[1] J. Esparza: Model Checking Using Net Unfoldings. *Science of Computer Programming*, Volume 23, 151–195, Elsevier (1994).

Addresses

Hans Fleischhack: Fachbereich Informatik, CvO Universität Oldenburg, D-26111 Oldenburg, Germany

Elisabeth Pelz: LACL, Dept. d'Informatique, Bât. P3, Université Paris XII–Val de Marne, F-94010 Créteil Cedex, France