Bounded Model Checking with Matching μ-Logic

1 Introduction

1.1 Transition Systems and Language Semantics

We define a signature of transition systems $\mathbb{\Sigma}^{\mathsf{TS}} = (\{\mathit{State}\}, \{\bullet \in \Sigma^{\mu}_{\mathit{State},\mathit{State}}\})$, where "$\bullet$" is a unary symbol (one-path next).

$\mathbb{\Sigma}^{\mathsf{TS}}$-models captures exactly the transition systems, where $\bullet \in \Sigma^{\mathsf{TS}}_{\mathit{State},\mathit{State}}$ is interpreted as the transition relation R. Specifically, for any transition system $\mathbb{S} = (S, R)$, we can regard $\mathbb{S}$ as a $\mathbb{\Sigma}^{\mathsf{TS}}$-model where S is the carrier set of *State* and $\bullet_{\mathbb{S}}(t) = \{s \in S \mid s\,R\,t\}$ contains all *R-predecessors* of t. $\bullet\varphi$ is matched by states which *have at least one next state* that satisfies φ. Its dual $\circ\varphi \equiv \neg\bullet\neg\varphi$ (called "all-path next") is matched by s if *for all* states t such that $s\,R\,t$, we have t matches φ. In particular, if s has no successor, then s matches $\circ\varphi$ for all φ.

A Language semantics can be seen as the definition of a transition system. In $\mathbb{K}$ framework, a language semantics is defined as a finite set of *reachability rules* over the signature $\mathbb{\Sigma}^{\mathsf{RS}} = \mathbb{\Sigma}^{\mathsf{cfg}} \cup \{\bullet \in \Sigma_{\mathit{Cfg},\mathit{Cfg}}\}$. $\mathbb{\Sigma}^{\mathsf{cfg}}$ is the signature (of *static program configurations*). It may have various sorts and symbols, among which there is a distinguished sort *Cfg*. Fix a $\mathbb{\Sigma}^{\mathsf{cfg}}$-model M^{cfg} called the *configuration model*, where $M^{\mathsf{cfg}}_{\mathit{Cfg}}$ is the set of all configurations. *Reachability rules*, or simply *rules* have the form $\varphi_1 \Rightarrow \varphi_2$ where φ_1, φ_2 are ML (without μ) $\mathbb{\Sigma}^{\mathsf{cfg}}$-patterns. A reachability system yields a transition system $\mathbb{S} = (M^{\mathsf{cfg}}_{\mathit{Cfg}}, R)$ where $s\ R\ t$ if there exist a rule $\varphi_1 \Rightarrow \varphi_2 \in S$ and an M^{cfg}-valuation ρ such that $s \in \bar{\rho}(\varphi_1)$ and $t \in \bar{\rho}(\varphi_2)$.

In matching logic, we can capture the same transition system by:

1. Desugar each reachablitiy rule $\varphi_{l_i} \Rightarrow \varphi_{r_i}$ to $\forall x_i . \varphi_{l_i} \to \bullet\varphi_{r_i}$, $x_i \in FV(\varphi_{l_i})$

2. Introduce *STEP* axiom schema:

$$\varphi \to \circ \bigvee_{i=1} \exists x_i . \lceil \varphi_{l_i} \wedge \varphi \rceil \wedge \varphi_{r_i}$$

   φ is an arbitrary pattern of sort *Cfg*

The set of reachability rule describes what should be included in the transition relation. The STEP axiom ensures that no junk is added to the transition relation. It is equivalent to say that the transition relation is the *least* set that satisfies the reachability rules.

1.2 Modal μ-logic in matching μ-logic

We embed modal μ-logic in matching μ-logic as follows:

$$\varphi ::= \varphi_{\mathsf{cfg}} \mid \varphi \wedge \varphi \mid \neg\varphi \mid \circ\varphi \mid \mu X.\, \varphi \text{ if } \varphi \text{ is positive in } X$$

φ_{cfg} is any matching logic patterns (without μ) of sort *Cfg*.

The following temporal modalities can be added as derived constructs:

$$\begin{aligned}
\text{“always” } \Box\varphi &\equiv \nu X.\, \varphi \wedge \circ X \\
\text{“eventually” } \Diamond\varphi &\equiv \mu X.\, \varphi \vee \bullet X \\
\text{“until” } \varphi_1\ U\ \varphi_2 &\equiv \mu X.\, \varphi_2 \vee (\varphi_1 \wedge \bullet X) \\
\text{“well-founded” } \mathsf{WF} &\equiv \mu X. \circ X
\end{aligned}$$

2 Bounded Model Checking

2.1 Example

2.1.1 Always

2.1.2 Eventually

2.2 Simplify pattern $\varphi \wedge \neg\psi$

Let us assume that term t_1 and t_2 are unifiable and $t_1 \wedge t_2 = t_1 \wedge mgu(t_1, t_2)$. Now we try to simplify the pattern $P = (t_1 \wedge c_1) \bigwedge \neg(t_2 \wedge c_2)$, where c_1 and c_2 are predicates. Suppose all the quantifiers are already pushed to the left. The key observation is that $t_1 \wedge \neg t_2 = t_1 \wedge \neg(t_1 \wedge t_2)$.

$$\begin{aligned}
P &= (t_1 \wedge c_1) \wedge (\neg t_2 \vee \neg c_2) \\
&= (t_1 \wedge c_1 \wedge \neg t_2) \vee (t_1 \wedge c_1 \wedge \neg c_2) \\
&= (t_1 \wedge \neg(t_1 \wedge t_2) \wedge c_1) \vee (t_1 \wedge c_1 \wedge \neg c_2) \\
&= (t_1 \wedge (\neg t_1 \vee \neg mgu(t_1, t_2) \wedge c_1) \vee (t_1 \wedge c_1 \wedge \neg c_2) \\
&= (t_1 \wedge c_1 \wedge \neg mgu(t_1, t_2)) \vee (t_1 \wedge c_1 \wedge \neg c_2) \\
&= t_1 \wedge c_1 \wedge \neg(mgu(t_1, t_2) \wedge c_2)
\end{aligned}$$

2.3 Soundness and Completeness