Refinement for Symbolic Trajectory Evaluation

Author and Author

Chalmers

Abstract. Model refinement such that it preserves symbolic trajectory evalutions.

Keywords: STE · Refinement · ?

1 Symbolic Trajectory Evaluation

Symbolic trajectory evaluation [3] (STE) is a high-performance model checking technique based on *symbolic simulation* extended with a temporal *next-time* operator to describe circuit behaviour over time. In its simplest form, STE tests the validity of an *assertion* of the form $A \Rightarrow C$, where both the *antecedent* A and *consequent* C are formulas in the following logic:

$$f ::= p \mid f \wedge f \mid P \rightarrow f \mid \mathbf{N}\, f$$

Here, p is a simple predicate over "values" in a circuit and P is a Boolean propositional formula, and the operators $\wedge$, $\rightarrow$ and $\mathbf{N}$ are conjunction, domain restriction and the next-time operator, respectively.

If the circuit contains Boolean signals, p is typically drawn from the following two predicates: $n\,\mathbf{is}\,1$ and $n\,\mathbf{is}\,0$, where n ranges over the signals (or nodes) in a circuit. For example, suppose we have a unit-delayed, two-input AND-gate, then it is reasonable to assume that the assertion $(in_1\ \mathbf{is}\,1 \wedge in_2\ \mathbf{is}\,1) \Rightarrow \mathbf{N}(out\ \mathbf{is}\,1)$ is true. Indeed, STE efficiently validates such statements for us.

While the truth semantics of an assertion in STE is defined as the satisfaction of its "defining" trajectory (bounded sequence of states) relative to a model structure of the circuit, what the STE algorithm computes is exactly the solution of a data-flow equation [1] in the classic format [2]. ...

2 Set-theoretic STE

Consider an arbitrary, but fixed, digital circuit M operating in discrete time. A *configuration* of M, denoted by $\mathbb{C}$, is non-empty and finite set that represents a snapshot of M at a discrete point in time. If the circuit M has m boolean signals, then its set of configurations is typically represented as a sequence $\mathbb{B}^m$, where $\mathbb{B} = \{0, 1\}$ is the set of boolean values.

Circuit Model A simple conceptual model of M is a *transition relation*, $M_R \subseteq \mathbb{C} \times \mathbb{C}$, where $(c, c') \in M_R$ means that M can move from c to c' in one step[1]. The power set of $\mathbb{C}$, denoted by $\wp(\mathbb{C})$, can be viewed as a the set of *predicates* on configurations, where $\cap$, $\cup$, and $\subseteq$ correspond to conjunction, disjunction and implication, respectively. We denote by $\cap S$ and $\cup S$ the intersection and union of all members of any $S \subseteq \wp(\mathbb{C})$.

M_R induces a *predicate transformer* $M_F \in \wp(\mathbb{C}) \to \wp(\mathbb{C})$ using the relational image operation:

$$M_F(C) = \{c' \in \mathbb{C} \mid \exists c \in C : (c, c') \in M_R\}$$

It is intuitively obvious that if M is in one of the configurations in $C \in \wp(\mathbb{C})$, then in one time step it must be in one of the configurations in $M_F(p)$. We also see that M_F distributes over arbitrary unions:

$$M_F(\cup S) = \cup\{M_F(C) \mid C \in S\}$$

for all $S \subseteq \wp(\mathbb{C})$. In general, any M_F that satisfies this distributive property also defines a M_R through the equivalence $(c, c') \in M_R \Leftrightarrow c' \in M_F(\{c\})$, that is to say, there is no loss of information going from M_R to M_F or vice versa. We adopt this functional model of M and drop its subscript.

Exactly what $\mathbb{C}$ and its signals are, is not important in this section. In practice, however, signals are typically divided into external, such as "inputs" and "outputs", and internal parts. While an input signal is generally controlled by the external environment, and thus unconstrained by M itself, non-input signals are determined by the circuit topology and functionality. For example, suppose M is the earlier example of a unit-delayed, two-input AND gate. We could then define its model $M \in \wp(\mathbb{B}^3) \to \wp(\mathbb{B}^3)$ as follows:

$$M(C) = \{\langle b_1, b_2, i_1 \wedge i_2\rangle \in \mathbb{B}^3 \mid \langle i_1, i_2, o\rangle \in C\}$$

Here i_1 and i_2 refer to the two inputs of the AND gate, o the ignored output, and b_1 and b_2 are unconstrained inputs for the new configurations.

Assertions and Satisfaction A *trajectory assertion* for M is quintuple $A = (S, s_0, R, \pi_\mathrm{a}, \pi_\mathrm{c})$, where S is a finite set of *states*, $s_0 \in S$ is an *initial state*, $R \subseteq S \times S$ is a *transition relation*, $\pi_\mathrm{a} \in S \to \wp(\mathbb{C})$ and $\pi_\mathrm{c} \in S \to \wp(\mathbb{C})$ label each state s with an *antecedent* $\pi_\mathrm{a}(s)$ and a *consequent* $\pi_\mathrm{c}(s)$. We assume that $(s, s_0) \notin S$ for all $s \in S$ without any loss of generality.

The circuit model M intuitively *satisfies* a trajectory assertion A if, for every path τ through M and every path ρ through A, τ satisfying the antecedents of ρ entails that τ also satisfies the consequents of ρ. To be specific, a path τ in M is referred to as a *trajectory* and is defined as a non-empty sequences of configurations, $\tau \in \mathbb{C}^+$, such that $\tau_n \in M(\{\tau_{n-1}\})$ for all $n \in \mathbb{N} : 0 < n < |\tau|$. And a path, or *run*, ρ of A is similarly a non-empty sequence of states, $\rho \in S^+$,

[1] Mention how this affects circuits with zero-delays?

such that $\rho_0 = s_0$ and $(\rho_{n-1}, \rho_n) \in R$ for all $n \in \mathbb{N} : 0 < n < |\rho|$. A finite trajectory τ satisfies the antecedents of a finite run ρ, denoted by $\tau \models_{\mathrm{a}} \rho$, iff $\tau_n \in \pi_{\mathrm{a}}(\rho_n)$ for all $n \in \mathbb{N} : n < |\tau| = |\rho|$; satisfaction of consequents is defined similarly with π_{c} and denoted by $\tau \models_{\mathrm{c}} \rho$.

That M satisfies A, denoted by $M \models A$, can then be formalized as follows:

$$\forall \tau \in \mathit{Traj}(M)^n : \forall \rho \in \mathit{Runs}(A)^n : |\tau| = |\rho| \Rightarrow (\tau \models_{\mathrm{a}} \rho \Rightarrow \tau \models_{\mathrm{c}} \rho)$$

where $\mathit{Traj}(M)^n$ and $\mathit{Runs}(A)^n$ denote the sets of all finite trajectories of M and runs of A, respectively. This satisfaction can be formulated equivalently as a problem for deterministic finite automaton.

2.1 Refinement

Consider another fixed, but arbitrary, circuit model $N \in \wp(\mathbb{D}) \to \wp(\mathbb{D})$, where $\mathbb{D}$ is a non-empty and finite set of configurations. Exactly what configurations such as $\mathbb{C}$ and $\mathbb{D}$ are, were not important previously. But to reason about refinement, which relates the visible behaviour of circuits, we make a distinction between their elements. Let $\sim \subseteq \mathbb{C} \times \mathbb{C}$ be an equivalence relation on $\mathbb{C}$. The equivalence class of a $c \in \mathbb{C}$ under $\sim$, denoted by $[c]$, is defined as $[c] = \{c' \in \mathbb{C} \mid c' \sim c\}$. With a slight abuse of notation, we overload both $\sim$ and $[\cdot]$ to accept configurations in $\mathbb{D}$. Furthermore, we extend $[\cdot]$ to sets $C \subseteq \mathbb{C}$ by taking the union of its image, i.e. $[C] = \cup\{[c] \in \wp(\mathbb{C}) \mid c \in C\}$.

Let there be a Galois connection between the predicates $\wp(\mathbb{C})$ and $\wp(\mathbb{D})$ ordered by set inclusion, given in terms of two monotone functions: an *abstraction* $\alpha \in \wp(\mathbb{C}) \to \wp(\mathbb{D})$ and a *concretisation* $\gamma \in \wp(\mathbb{D}) \to \wp(\mathbb{C})$ function. An essential property of a Galois connection is that, for all $C \in \wp(\mathbb{C})$ and $D \in \wp(\mathbb{D})$, the pair α and γ satisfy the equality: $\alpha(C) \subseteq D \Leftrightarrow C \subseteq \gamma(D)$. Intuitively, this relation is an extension of the partial orderings inside $\wp(\mathbb{C})$ and $\wp(\mathbb{D})$ to an ordering between them. $\forall c \in \mathbb{C} : \alpha([\{c\}]) \neq \emptyset.\ c_1 \sim c_2 \Rightarrow (\alpha(c_1) \subseteq [d] \Rightarrow \alpha(c_2) \subseteq [d])$.

We can now formalize that M *refines* N, denoted by $M \leq N$, as follows:

$$\forall \tau \in \mathit{Traj}(M)^\infty : \exists \upsilon \in \mathit{Traj}(N)^\infty : \alpha([\tau]) \subseteq [\upsilon]$$

where $\mathit{Traj}(M)^\infty$ and $\mathit{Traj}(N)^\infty$ denotes the sets of all *infinite* trajectories in M and N, respectively, and $\alpha([\tau]) \subseteq [\upsilon]$ implies that $\alpha([\tau_n]) \subseteq [\upsilon_n]$ for all $n \in \mathbb{N}$. We note that trajectories are prefix-closed and, hence, every prefix of an infinite trajectory is a finite trajectory. The reverse, that every finite trajectory can be extended into an infinite trajectory, follows from how there are no "stuck" configurations in M. $\forall c \in \mathbb{C} : M(\{c\}) \neq \emptyset$.

Recall that a trajectory assertion for N is a quintuple $A = (S, s_0, R, \pi_{\mathrm{a}}, \pi_{\mathrm{c}})$, where $\pi_{\mathrm{a}} \in S \to \wp(\mathbb{D})$ and $\pi_{\mathrm{c}} \in S \to \wp(\mathbb{D})$ label each $s \in S$ with its antecedents and consequents, respectively. If π_{a} and π_{c} are invariant under the equivalence class $\sim$ of $\mathbb{D}$, i.e. $\pi_{\mathrm{a}}(s) = [\pi_{\mathrm{a}}(s)]$ and $\pi_{\mathrm{c}}(s) = [\pi_{\mathrm{c}}(s)]$ for all $s \in S$, then we refer to A as an *trajectory class assertion* and suffix it as A_c. Furthermore, we define $\gamma(A) = (S, s_0, R, \gamma(\pi_{\mathrm{a}}), \gamma(\pi_{\mathrm{c}}))$, where $\gamma(\pi_{\mathrm{a}}) = \lambda s \in S : \gamma(\pi_{\mathrm{a}}(s))$ and similarly $\gamma(\pi_{\mathrm{c}}) = \lambda s \in S : \gamma(\pi_{\mathrm{c}}(s))$.

Theorem 1. $M \leq N \Rightarrow (N \models A_c \Rightarrow M \models \gamma(A_c))$

Let $\ll \subseteq \wp(\mathbb{C}) \times \wp(\mathbb{D})$ be a *simulation relation* between predicates $\mathbb{C}$ and $\mathbb{D}$ ordered by set inclusion, such that, if $C \ll D$ for all $C \in \mathbb{C}$ and $D \in \mathbb{D}$, then the following conditions hold: $\alpha([C]) \subseteq [D]$, and $M(C) \ll N(D)$. ...

We say that M refines N by *set-theoretic simulation*, denoted by $M \leq_{\text{set}} N$, if there exists a simulation relation $\ll$ such that $\mathbb{C} \ll \mathbb{D}$, $c_1 \sim c_2 \Rightarrow (\alpha(c_1) \subseteq [d] \Rightarrow \alpha(c_2) \subseteq [d])$, ...

Theorem 2. $M \leq N \Leftrightarrow M \leq_{set} N$

Theorem 3. $M \leq_{set} N \Rightarrow (N \models A_c \Rightarrow M \models \gamma(A_c))$

A Appendices

Lemma 1. $M \leq N \Rightarrow (c_1 \sim c_2 \Rightarrow (\alpha(c_1) \subseteq [d] \Rightarrow \alpha(c_2) \subseteq [d]))$

Suppose $c_1 \sim c_2$ and $\alpha(c_1) \subseteq [d]$ but $\alpha(c_2) \nsubseteq [d]$. Because $\sim$ is a partitioning of $\mathbb{D}$, there exists no $d_2 \sim d$ such that both $\alpha(c_1) \subseteq [d_2]$ and $\alpha(c_2) \subseteq [d_2]$. But refinement states that such a d_2 must exist, hence a contradiction.

A.1 Proof of Theorem 1

We freely expand the definition of $[\cdot]$ for $\emptyset$, singletons $\{c \in \mathbb{C}\}$, and $\mathbb{C}$.

Lemma 2. $C = [C] \wedge [d] \cap C \neq \emptyset \Rightarrow [d] \subseteq C$

$$\begin{aligned} [d] \cap C \neq \emptyset &\Rightarrow \exists x : x \in [d] \wedge x \in C && \text{(definition of } \cap) \\ &\Leftrightarrow \exists x : x \in [d] \wedge [x] \subseteq C && \text{(invariance of } C) \\ &\Rightarrow [d] \subseteq C && ([x] = [d]) \end{aligned}$$

Lemma 3. $d \in \pi_c(s) \wedge \alpha([c]) \subseteq [d] \Rightarrow c \in \gamma(\pi_c(s))$

$$\begin{aligned} d \in \pi_c(s) &\Leftrightarrow [d] \subseteq \pi_c(s) && \text{(invariance of } \pi_c) \\ &\Rightarrow \alpha([c]) \subseteq \pi_c(s) && (\alpha([c]) \subseteq [d] \text{ assumption}) \\ &\Rightarrow \alpha(\{c\}) \subseteq \pi_c(s) && (\alpha \text{ distributes over } \cup) \\ &\Leftrightarrow c \in \gamma(\pi_c(s)) && \text{(Galois connection)} \end{aligned}$$

Lemma 4. $c \in \gamma(\pi_a(s)) \wedge \alpha([c]) \subseteq [d] \Rightarrow d \in \pi_a(s)$

$$\begin{aligned} c \in \gamma(\pi_a(s)) &\Leftrightarrow \alpha(\{c\}) \subseteq \pi_a(s) && \text{(Galois connection)} \quad (1) \\ \alpha([c]) \subseteq [d] &\Rightarrow \alpha(\{c\}) \subseteq [d] && (\alpha \text{ distributes over } \cup) \\ &\Rightarrow [d] \subseteq \pi_a(s) && \text{(lemma 2 and equation 1)} \end{aligned}$$

Proof. We are given $\tau \in \mathit{Traj}(M)$ and $\rho \in \mathit{Runs}(\gamma(A))$, such that $|\tau| = |\rho|$ and $\tau_n \in \gamma(\pi_a(\rho_n))$ for all $n \in \mathbb{N} : n < |\tau|$. We must then show that $\tau_n \in \gamma(\pi_c(\rho_n))$. By the refinement assumption, there must exist $\tau' \in \mathit{Traj}(M)^\infty$ and $\upsilon' \in \mathit{Traj}(N)^\infty$ such that $\tau \prec \tau'$ and $\alpha([\tau']) \subseteq [\upsilon']$. Let $\upsilon \prec \upsilon'$ such that $|\upsilon| = |\tau|$, which implies that $\alpha([\tau]) \subseteq [\upsilon]$. Lemma 4 then shows that $\upsilon_n \in \pi_a(\rho_n)$ and, by the assumption, we have $\upsilon_n \in \pi_c(\rho_n)$. Lemma 3 then shows that $\tau_n \in \gamma(\pi_c(\rho_n))$.

A.2 Proof of Theorem 2

Proof. We show each direction of the theorem separately.

A.3 Proof of Theorem 3

Lemma 5. $[D] = D \Rightarrow ([C \cap D] \Leftrightarrow [C] \cap D)$

$$\begin{aligned} x \in [C \cap D] &\Leftrightarrow \exists y : x \sim y \wedge y \in C \wedge y \in D && \text{(definition of } [\cdot] \text{ and } \cap) \\ &\Leftrightarrow \exists y : x \sim y \wedge y \in C \wedge x \in D && (x \sim y \text{ and } D = [D]) \\ &\Leftrightarrow x \in [C] \cap D && \text{(definition of } [\cdot] \text{ and } \cap) \end{aligned}$$

Lemma 6. $c \ll d \Rightarrow M(\gamma(\pi_a(\rho)) \cap c) \ll N(\pi_a(\rho) \cap d)$

Thought this could be useful for the next lemma.

Lemma 7. $\forall s \in S : \Phi_*(s) \ll \Psi_*(s)$

Since $\Phi_*(s) = \lim \Phi_n(s)$ and $\Psi_*(s) = \lim \Psi_n(s)$, it suffices to prove that $\Phi_n(s) \ll \Psi_n(s)$ for all $s \in S$ and $n \in \mathbb{N}$. We do so by induction on n. The base case, where $\Phi_0(s) = \bot$ and $\Psi_0(s) = \bot$, follows from ???. For the inductive step, assume that $\Phi_n(s) \ll \Psi_n(s)$ for all $s \in S$. For $s = s_0$, we have that $\Phi_{n+1}(s_0) = \top$ and $\Psi_{n+1}(s_0) = \top$, which follows from ???. For any $s \neq s_0$, we have:

$$\begin{aligned} &\Psi_n(s) \ll \Phi_n(s) \\ &\quad \Rightarrow \Psi_{n+1}(s) \ll \Phi_{n+1}(s) \\ &\quad = \cup\{G(s')(\Psi_n(s')) \mid (s', s) \in R\} \ll \cup\{F(s')(\Phi_n(s')) \mid (s', s) \in R\} \\ &\quad = \cup\{M(\gamma(\pi_{\mathrm{a}}(s')) \cap \Psi_n(s')) \mid (s', s) \in R\} \ll \cup\{N(\pi_{\mathrm{a}}(s') \cap \Phi_n(s')) \mid (s', s) \in R\} \end{aligned}$$

Proof. Consider an arbitrary $s \in S$ and let $p = \Phi_*(s) \cap \gamma(\pi_{\mathrm{a}}(s))$, thus $p \subseteq \Phi_*(s)$ and $p \subseteq \gamma(\pi_{\mathrm{a}}(s))$ or $\alpha(p) \subseteq \pi_{\mathrm{a}}(s)$. Lemma 7 implies that $\alpha([\Phi_*(s)]) \subseteq [\Psi_*(s)]$ and, by the monotonicity of α and $[\cdot]$, $\alpha([p]) \subseteq \alpha([\Phi_*(s)]) \subseteq [\Psi_*(s)]$. Using the invariance of π_{c} and lemma 5, the satisfaction of $\pi_{\mathrm{c}}(s)$ by N can be restated as $\pi_{\mathrm{c}}(s) \supseteq [\Psi_*(s) \cap \pi_{\mathrm{a}}(s)] = [\Psi_*(s)] \cap \pi_{\mathrm{a}}(s)$. Since $\alpha(p)$ is $\subseteq [\Psi_*(s)]$ and $\subseteq \pi_{\mathrm{a}}(s)$, we see that $\alpha(p) \subseteq \pi_{\mathrm{c}}(s)$, or $p \subseteq \gamma(\pi_{\mathrm{c}}(s))$ as desired.

References

1. Chou, C.T.: The mathematical foundation of symbolic trajectory evaluation. In: International Conference on Computer Aided Verification. pp. 196–207. Springer (1999)
2. Muchnick, S., et al.: Advanced compiler design implementation. Morgan kaufmann (1997)
3. Seger, C.J.H., Bryant, R.E.: Formal verification by symbolic evaluation of partially-ordered trajectories. Formal Methods in System Design **6**(2), 147–189 (1995)