Refinement for Symbolic Trajectory Evaluation

Authors

Chalmers

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

Keywords: STE · Refinement · ?

1 Introduction to STE

1.1 Original STE (WIP)

Symbolic trajectory evaluation [5] (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 three predicates: $\top$, $n\,\mathbf{is}\,1$ 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 $(\mathit{in_1}\,\mathbf{is}\,1 \wedge \mathit{in_2}\,\mathbf{is}\,1) \Rightarrow \mathbf{N}(\mathit{out}\,\mathbf{is}\,1)$ 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 [4]. . . .

1.2 Lattice-theoretic STE (OK)

Circuits Consider an arbitrary, but fixed, digital circuit M operating in discrete time. A *configuration* of M, denoted by 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.

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

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

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

It is intuitively obvious that if M is in one of the configurations in $p \in \mathcal{P}(C)$, then in one time step it must be in one of the configurations in $M_F(p)$, or in other words, $(c, c') \in M_R \Leftrightarrow c' \in M_F(\{c\})$. We adopt this functional model of M and drop its subscript.

Manipulating subsets of $\mathbb{B}^m$ is however impractical for even moderately large m, which leads us to one of the key insights of symbolic STE. Namely, instead of manipulating subsets of $\mathbb{B}^m$ directly, one can use sequences of ternary values $\mathbb{T} = \mathbb{B} \cup \{X\}$ to approximate them, whose sizes are only linear in m. Here the 1 and 0 from $\mathbb{B}$ denotes specific, defined values whereas X denotes an "unknown" value that could be either 1 or 0. This intuition induces a partial order $\sqsubseteq$ on $\mathbb{T}$, where $0 \sqsubseteq X$ and $1 \sqsubseteq X$[1]. For any $m \in \mathbb{N}$, this ordering on $\mathbb{T}$ is lifted component-wise to $\mathbb{T}^m$.

Lattices Let $(\hat{P}, \sqsubseteq)$ be a finite, complete lattice of *abstract predicates* in which the meet $\sqcap$ and join $\sqcup$ of any subset $Q \subseteq \hat{P}$ exists. Similar to the previous set operations for power sets, $\sqcap$, $\sqcup$ and $\sqsubseteq$ correspond to conjunction, disjunction and implication for abstract predicates, respectively. Furthermore, for any $Q \subseteq \hat{P}$, we denote by $\sqcap Q$ and $\sqcup Q$ the meet and join of all members of Q.

Note that $\mathbb{T}^m$ does not quite form a complete lattice because it lacks a bottom: both $0 \sqsubseteq X$ and $1 \sqsubseteq X$ but 0 and 1 are equally defined. A special bottom element $\perp$ is therefore introduced, such that $\perp \sqsubseteq t$ and $\perp \neq t$ for all $t \in \mathbb{T}^m$. The extended $\mathbb{T}^m_\perp = \mathbb{T}^m \cup \{\perp\}$ then becomes a complete lattice. We denote its top element $\langle X, \ldots, X\rangle$ by $\top$.

Let there also be Galois connection $\ll \in \subseteq \mathcal{P}(C) \times \hat{P}$, relating concrete and abstract predicates such that for all $Q \subseteq \mathcal{P}(C)$ and $\hat{Q} \subseteq \hat{P}$:

$$Q \ll \hat{Q} \Leftrightarrow \cup Q \ll \sqcap \hat{Q}$$

where $Q \ll \hat{Q} \Leftrightarrow \forall p \in Q : \forall \hat{p} \in \hat{Q} : p \ll \hat{p}$. Intuitively, $p \ll \hat{p}$ states that p can be "approximated" as $\hat{p}$, and we note that $\ll$ is an extension of the partial orders of $\mathcal{P}(C)$ and $\hat{P}$ to an ordering between $\mathcal{P}(C)$ and $\hat{P}$.

It is sometimes convenient to define a Galois connection in terms of an *abstraction* $\alpha \in \mathcal{P}(C) \to \hat{P}$ and a *concretisation* $\gamma \in \hat{P} \to \mathcal{P}(C)$ function, from which one can derive $\ll$ as follows: $p \ll \hat{p} \Leftrightarrow p \subseteq \gamma(\hat{p})$ or $p \ll \hat{p} \Leftrightarrow \alpha(p) \sqsubseteq \hat{p}$. For

[1] Reverse of original ordering in STE.

example, a Galois connection from $\mathcal{P}(\mathbb{B}^m)$ to $\mathbb{T}_\perp^m$ for any $m \in \mathbb{N}$ can be defined in a natural way by specifying its concretisation function $\Gamma \in \mathbb{T}_\perp^m \to \mathcal{P}(\mathbb{B}^m)$:

$$\begin{aligned} \Gamma(\langle t_0, \ldots, t_{m-1}\rangle) &= \{\langle b_0, \ldots, b_{m-1}\rangle \in \mathbb{B}^m \mid \forall i < m : t_i \neq X \Rightarrow b_i = t_i\} \\ \Gamma(\perp) &= \emptyset \end{aligned}$$

Listing each boolean sequence approximated by the given ternary sequence.

Trajectories An *abstract predicate transformer* $\hat{M} \in \hat{P} \to \hat{P}$ is an *abstract interpretation* [2] of $M \in \mathcal{P}(C) \to \mathcal{P}(C)$ iff it preserves $\perp$, $\hat{M}(\perp) = \perp$; is monotonic, $\hat{p} \sqsubseteq \hat{q} \Rightarrow \hat{M}(\hat{p}) \sqsubseteq \hat{M}(\hat{q})$ for all $\hat{p}, \hat{q} \in \hat{P}$; and $\ll$ is a *simulation relation* from $\mathcal{P}(C)$ to $\hat{P}$, $p \ll \hat{p} \Rightarrow M(p) \ll \hat{M}(\hat{p})$ for all $p \in \mathcal{P}(C)$ and $\hat{p} \in \hat{P}$.

A *trajectory assertion* for $\hat{M}$ is a quintuple $\hat{A} = (S, s_0, R, \pi_a, \pi_c)$, 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*, $\pi_a \in S \to \hat{P}$ and $\pi_c \in S \to \hat{P}$ label each state s with an *antecedent* $\pi_a(s)$ and a *consequent* $\pi_c(s)$. Furthermore, we assume that $(s, s_0) \notin S$ for all $s \in S$ without any loss of generality.

Define $F \in S \to (\hat{P} \to \hat{P})$ and $\mathcal{F} \in (S \to \hat{P}) \to (S \to \hat{P})$ as follows:

$$F(s)(\hat{p}) = \hat{M}(\pi_a(s) \sqcap \hat{p}) \tag{1}$$

$$\mathcal{F}(\Phi)(s) = \textbf{if } (s = s_0) \textbf{ then } \top \textbf{ else } \sqcup \{F(s')(\Phi(s')) \mid (s', s) \in R\} \tag{2}$$

for all $\Phi \in S \to \mathcal{P}(C)$ and $s \in S$. F preserves $\perp$ and both F and $\mathcal{F}$ are monotonic, where two $\Phi, \Phi' \in S \to \hat{P}$ are ordered as $\Phi \sqsubseteq \Phi' \Leftrightarrow \forall s \in S : \Phi(s) \sqsubseteq \Phi'(s)$. Let $\Phi_* \in S \to \hat{P}$ be the least fixpoint of the equation $\Phi = \mathcal{F}(\Phi)$ [3]. Since both S and $\hat{P}$ are finite, Φ_* is given by $\lim \Phi_n(s)$ where Φ_n is defined as follows:

$$\Phi_n = \textbf{if } (n = 0) \textbf{ then } (\lambda s \in S : \perp) \textbf{ else } \mathcal{F}(\Phi_{n-1}) \tag{3}$$

We say that the abstract circuit $\hat{M}$ *satisfies* a lattice-based, abstract trajectory assertion $\hat{A}$, denoted by $\hat{M} \models \hat{A}$, iff:

$$\forall s \in S : \Phi_*(s) \sqcap \pi_\alpha(s) \sqsubseteq \pi_c(s) \tag{4}$$

$\hat{M} \models \hat{A}$ implies that a concretisation of $\hat{A}$ can also be satisfied by the original, set-based model M [1].

2 System refinement (WIP)

Consider another fixed, but arbitrary, circuit N such that configurations of M and N have the same externally visible elements but can differ internally. Let $\hat{N} \in \hat{Q} \to \hat{Q}$ be an abstract predicate transformer of N, we then we say that $\hat{M}$ *refines* $\hat{N}$ if every *externally visible behaviour* allowed by $\hat{M}$ is also allowed by $\hat{N}$, regardless of any initial configurations.

M always sets it $\hat{I}$ to $\top$ and doesn't really work for zero-delay circuits.

The value domain for inputs and outputs must be the same so they are comparable. Not just that they have the "same" inputs and output elements.

Example?

Drivers Let the *externally visible* parts of an abstract predicate $\hat{P}$ be the subsets given by two projections, i and o, identifying the "inputs" and "outputs" of $\hat{P}$, respectively. Further, let $[\![\cdot]\!] \in \hat{P} \rightarrow \hat{O}$ be a mapping that takes each $\hat{p} \in \hat{P}$ to its visible outputs $[\![\hat{p}]\!] \in \hat{O}$; $[\![\cdot]\!]$ is extended to sequences component-wise. With a slight abuse of notation, we overload both projections and the mapping to also accept predicates from $\hat{Q}$ and note that $i(\hat{P}) = i(\hat{Q}) = \hat{I}$ and $o(\hat{P}) = o(\hat{Q}) = \hat{O}$ since M and N share inputs and outputs.

A *driver* of $\hat{M}$ and $\hat{N}$ is a nonempty sequence of inputs, $\delta \in \hat{I}^+$, and induces a trajectory τ in $\hat{M}$ (resp. $\hat{N}$) where $\tau[0] = \top$ and $\forall i \in \mathbb{N} : 0 < i < |\delta + 1| \Rightarrow \tau[j] = \hat{M}(\delta[j-1] \sqcap \tau[j-1])$; the trajectory induced by a driver δ in $\hat{M}$ is denoted by $\mathit{Traj}(\hat{M})(\delta)$. Intuitively, if $\hat{M}$ produces the same, or at least more specified, outputs than $\hat{N}$ for all possible drivers, then every visible behaviour of $\hat{M}$ is covered by $\hat{N}$. We thus say that $\hat{M}$ refines $\hat{N}$, denoted by $\hat{M} \leq \hat{N}$, iff:

$$\forall \delta \in \hat{I}^+ : [\![\mathit{Traj}(\hat{M})(\delta)]\!] \sqsubseteq [\![\mathit{Traj}(\hat{N})(\delta)]\!]$$

Example!

Simulation Let $\preceq \in \hat{P} \times \hat{Q}$ be a simulation relation such that $\hat{p} \preceq \hat{q}$ implies (1) $[\![\hat{p}]\!] \sqsubseteq [\![\hat{q}]\!]$ and (2) $\hat{M}(\hat{\imath} \sqcap \hat{p}) \preceq \hat{N}(\hat{\imath} \sqcap \hat{q})$ for all inputs $\hat{\imath} \in \hat{I}$. We extend this relation to $\hat{M}$ and $\hat{N}$ such that $\hat{M} \preceq \hat{N}$ iff their top elements are related, $\top \preceq \top$. We then simplify refinement thus: $\hat{M} \leq \hat{N} \Leftrightarrow \hat{M} \preceq \hat{N}$.

Example!

Trajectory A trajectory assertion $\hat{A} = (S, s_0, R, \pi_a, \pi_c)$ for $\hat{N}$ where antecedents only mention inputs, $\pi_a \in S \rightarrow \hat{I}$, and consequents only mention outputs, $\pi_c \in S \rightarrow \hat{O}$, is referred to as an *external trajectory assertion*. Intuitively, a satisfied external trajectory assertion is property of N that must hold regardless of its internal state.

Let $G \in S \rightarrow (\hat{Q} \rightarrow \hat{Q})$ and $\mathcal{G} \in (S \rightarrow \hat{Q}) \rightarrow (S \rightarrow \hat{Q})$ be the duals of F and $\mathcal{F}$ in $\hat{N}$, respectively. Further, let Ψ_* be the least fix point of $\Psi = \mathcal{G}(\Psi)$ and the dual of Φ_* in $\hat{N}$. Fix for red below.

For all below, assume $\hat{M} \preceq \hat{N}$.

(1): $\hat{p} \preceq \hat{q} \Rightarrow \forall s \in S : F(s)(\hat{p}) \preceq G(s)(\hat{q})$. $F(s)(\hat{p}) = \hat{M}(\pi_a(s) \sqcap \hat{p})$ and $G(s)(\hat{q}) = \hat{N}(\pi_a(s) \sqcap \hat{q})$ where $\pi_a(s) \in \hat{I}$. By prop. 2 of $\hat{p} \preceq \hat{q}$, we thus know that $F(s)(\hat{p}) \preceq G(s)(\hat{q})$.

(2): $\Phi_*(s) \preceq \Psi_*(s)$ for all $s \in S$. By induction on n, like below. Base case and case when $s = s_0$ are both trivial. For the inductive step, assume that $\Phi_n(s) \preceq \Psi_n(s)$ for all $s \in S$. For any $s \neq s_0$, we have that $\Phi_{n+1} = \sqcup\{F(s')(\Phi_n(s')) \mid (s', s) \in R\}$. By (1) we know that ...

If $\hat{M} \preceq \hat{N}$, we claim that $[\![\Phi_*(s)]\!] \sqsubseteq [\![\Psi_*(s)]\!]$ for all $s \in S$. Since $\Phi_*(s) = \lim \Phi_n(s)$ and $\Psi_*(s) = \lim \Psi_n(s)$, it suffices to prove that $[\![\Phi_n(s)]\!] \sqsubseteq [\![\Psi_n(s)]\!]$ for all $s \in S$ and $n \in \mathbb{N}$. We do so by induction on n. The base case is trivially true ($[\![\bot]\!] \sqsubseteq [\![\bot]\!]$). For the inductive step, assume $[\![\Phi_n(s)]\!] \sqsubseteq [\![\Psi_n(s)]\!]$ for all $s \in S$. That $[\![\Phi_{n+1}(s)]\!] \sqsubseteq [\![\Psi_{n+1}(s)]\!]$ for s_0 is also trivially correct ($[\![\top]\!] \sqsubseteq [\![\top]\!]$). For any $s \neq s_0 \in S$, we have $[\![\Phi_{n+1}(s)]\!] = [\![\sqcup\{F(s')(\Phi_n(s')) \mid (s', s) \in R\}]\!] \sqsubseteq [\![\sqcup\{F(s')(\Psi_n(s')) \mid (s', s) \in R\}]\!]$. I would like to claim that this is $\sqsubseteq [\![\sqcup\{G(s')(\Psi_n(s')) \mid (s', s) \in R\}]\!] = [\![\Psi_{n+1}]\!]$. But I'm missing a relation between F and G, which should come from $\preceq$.

Theorem 1. *Given two abstract predicate transformers $\hat{M}$ and $\hat{N}$ such that $\hat{M} \preceq \hat{N}$, and a external trajectory assertion $\hat{A}$ for $\hat{N}$, then $\hat{N} \models \hat{A} \Rightarrow \hat{M} \models \hat{A}$.*

A Proofs (WIP)

We show that $\hat{M} \preceq \hat{N} \Leftrightarrow \hat{M} \leq \hat{N}$.

$\boldsymbol{\hat{M} \preceq \hat{N} \Rightarrow \hat{M} \leq \hat{N}}$**:** Consider an arbitrary $\delta \in \hat{I}^+$ and the two trajectories it induces, $\tau \in \mathit{Traj}(\hat{M})(\delta)$ and $v \in \mathit{Traj}(\hat{N})(\delta)$. For index 0, prop. 2 of $\hat{M} \preceq \hat{N}$ and $\delta[0] \in \hat{I}$ implies that $\tau[0] = \hat{M}(\delta[0]) = \hat{M}(\delta[0] \sqcap \top) \preceq \hat{N}(\delta[0] \sqcap \top)\hat{N}(\delta[0]) = v[0]$, from which prop. 1 states that $[\![\tau[0]]\!] \sqsubseteq [\![v[0]]\!]$. For index 1, since $\delta[1] \in \hat{I}$, prop. 2 of $\hat{M}(\tau[0]) \preceq \hat{N}(v[0])$ implies that $\tau[1] = \hat{M}(\delta[1] \sqcap \tau[0]) \sqsubseteq \hat{N}(\delta[1] \sqcap v[0]) = v[1]$, from which prop. 1 states that $[\![\tau[1]]\!] \sqsubseteq [\![v[1]]\!]$. Etc. for higher indices.

Counterexample?

$\boldsymbol{\hat{M} \preceq \hat{N} \Leftarrow \hat{M} \leq \hat{N}}$**:** Prop. 1 of $\hat{M} \preceq \hat{N}$, that $\top \preceq \top$, is obviously true. For prop. 2, we note that every $\hat{\imath}_0 \in \hat{I}$ also forms a valid 1-length driver $\langle \hat{\imath}_0 \rangle \in \hat{I}^+$. Combined with $\hat{M} \leq \hat{N}$, we then have that $[\![\hat{M}(\hat{\imath}_0 \sqcap \top)]\!] = [\![\hat{M}(\hat{\imath}_0)]\!] \sqsubseteq [\![\hat{N}(\hat{\imath}_0)]\!] = [\![\hat{N}(\hat{\imath}_0 \sqcap \top)]\!]$ for all $\hat{\imath}_0 \in \hat{I}$. The relation thus holds at least for the first step. But for any choice of $\hat{\imath}_0$, every following choice of $\hat{\imath}_1 \in \hat{I}$ also forms a valid 2-length driver $\langle \hat{\imath}_0, \hat{\imath}_1 \rangle \in \hat{I}^+$. We therefore have that $[\![\hat{M}(\hat{\imath}_1 \sqcap \hat{M}(\hat{\imath}_0))]\!] \sqsubseteq [\![\hat{N}(\hat{\imath}_1 \sqcap \hat{N}(\hat{\imath}_0))]\!]$ for all $\hat{\imath}_0, \hat{\imath}_1 \in \hat{I}$ and the relation thus hold for a second step as well. Etc. for further steps. Build this relation and show that it has the desired properties.

Construct the simulation relation.

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. Cousot, P.: Abstract interpretation. ACM Computing Surveys (CSUR) **28**(2), 324–328 (1996)
3. Davey, B.A., Priestley, H.A.: Introduction to lattices and order. Cambridge university press (2002)

4. Muchnick, S., et al.: Advanced compiler design implementation. Morgan kaufmann (1997)
5. 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)