

# Refinement for Symbolic Trajectory Evaluation

Author and Author

Chalmers

**Abstract.** Model refinement such that it preserves symbolic trajectory evaluations.

**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:  $n \text{ is } 1$  and  $n \text{ is } 0$ , 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  $(in_1 \text{ is } 1 \wedge in_2 \text{ is } 1) \Rightarrow \mathbf{N}(out \text{ is } 1)$  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<sup>1</sup>. 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*  $M_F \in \wp(\mathbb{C}) \rightarrow \wp(\mathbb{C})$  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, supposed  $M$  is the earlier example of a unit-delayed two-input AND gate, we could then define its model  $M \in \wp(\mathbb{B}^3) \rightarrow \wp(\mathbb{B}^3)$  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  $A = (S, s_0, R, \pi_a, \pi_c)$ , 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*,  $\pi_a \in S \rightarrow \wp(\mathbb{C})$  and  $\pi_c \in S \rightarrow \wp(\mathbb{C})$  label each state  $s$  with an *antecedent*  $\pi_a(s)$  and a *consequent*  $\pi_c(s)$ . We assume that  $(s, s_0) \notin R$  for all  $s \in S$  without any loss of generality.

The circuit model  $M$  intuitively *satisfies* an assertion  $A$  if, for every *trajectory*  $\tau$  through  $M$  and every *run*  $\rho$  through  $A$ ,  $\tau$  satisfying the antecedents of  $\rho$  entails that  $\tau$  also satisfies the consequents of  $\rho$ . To be specific, a *trajectory* of  $M$  is 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 *run* of  $A$  is a non-empty sequence of states,  $\rho \in S^+$ , such that  $\rho_0 = s_0$  and  $(\rho_{n-1}, \rho_n) \in R$  for all  $n \in \mathbb{N} : 0 < n < |\rho|$ . A

<sup>1</sup> Mention how this affects circuits with zero-delays?

finite trajectory  $\tau$  satisfies the antecedents of a finite run  $\rho$ , denoted by  $\tau \models_a \rho$ , iff  $\tau_n \in \pi_a(\rho_n)$  for all  $n \in \mathbb{N} : n < |\tau| = |\rho|$ ; satisfaction of consequents is defined similarly with  $\pi_c$  and denoted by  $\tau \models_c \rho$ .

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

$$\forall \tau \in \text{Traj}(M)^n : \forall \rho \in \text{Runs}(A)^n : |\tau| = |\rho| \Rightarrow (\tau \models_a \rho \Rightarrow \tau \models_c \rho)$$

where  $\text{Traj}(M)^n$  and  $\text{Runs}(A)^n$  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  $N \in \wp(\mathbb{D}) \rightarrow \wp(\mathbb{D})$ , 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 external 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  $\mathbb{D}$ . We also extend  $[\cdot]$  to sets  $C \subseteq \mathbb{C}$  as  $[C] = \cup\{[c] \in \wp(\mathbb{C}) \mid c \in C\}$ .

Let there be a Galois connection between predicates  $\wp(\mathbb{C})$  and  $\wp(\mathbb{D})$  ordered by set inclusion. The usual definition of a Galois connection is in terms of an *abstraction*  $\alpha \in \wp(\mathbb{C}) \rightarrow \wp(\mathbb{D})$  and a *concretisation*  $\gamma \in \wp(\mathbb{D}) \rightarrow \wp(\mathbb{C})$  function, such that  $\alpha(C) \subseteq D \Leftrightarrow C \subseteq \gamma(D)$  for all  $C \in \wp(\mathbb{C})$  and  $D \in \wp(\mathbb{D})$ .

**We require that  $\alpha(\{[c]\}) \neq \emptyset$  for all  $c \in \mathbb{C}$ .**

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

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

That is, for every infinite sequence of configurations  $\tau$  permitted by  $M$ , there must exist a sequence  $v$  for  $N$  which approximates the behaviour of  $\tau$  according to their equivalence relations.

**We require that  $M(\{[c]\}) \neq \emptyset$  for all  $c \in \mathbb{C}$ .**

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

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

Let  $\ll \subseteq \wp(\mathbb{C}) \times \wp(\mathbb{D})$  be the greatest *simulation relation* between predicates of  $C \subseteq \mathbb{C}$  and  $D \subseteq \mathbb{D}$ , such that  $C$  is approximated by  $D$  according to their equivalence classes, i.e.  $C \ll D$  iff  $M(C) \ll N(D)$  and  $\alpha([C]) \subseteq [D]$ . We say that  $M$  refines  $N$  by set-theoretic simulation, denoted by  $M \leq_{\text{set}} N$ , iff ...

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

## A Appendices

### A.1 Proof of Theorem 1

We freely expand the relation  $\ll$  into its standard Galois form and use the facts that  $[\emptyset] = \emptyset$ ,  $[\{c\}] = [c]$ , and  $[\mathbb{C}] = \mathbb{C}$  (similar facts hold for  $\mathbb{D}$ ).

**Lemma 1.**  $C = [C] \wedge [d] \cap C \neq \emptyset \Rightarrow [d] \subseteq C$

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

**Lemma 2.**  $d \in \pi_c(s) \wedge \{c\} \ll \{d\} \Rightarrow c \in \gamma(\pi_c(s))$

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

**Lemma 3.**  $c \in \gamma(\pi_a(s)) \wedge \{c\} \ll \{d\} \Rightarrow d \in \pi_a(s)$

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

*Proof.* We are given  $\tau \in \text{Traj}(M)$  and  $\rho \in \text{Runs}(\gamma(A))$ , 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  $\tau' \in \text{Traj}(M)^\infty$  and  $v' \in \text{Traj}(N)^\infty$  such that  $\tau \prec \tau'$  and  $\tau' \ll v'$ . Let  $v \prec v'$  such that  $|v| = |\tau|$ , which implies that  $\tau \ll v$ . Lemma 3 then shows that  $v_n \in \pi_a(\rho_n)$  and, by the assumption, we have  $v_n \in \pi_c(\rho_n)$ . Lemma 2 then shows that  $\tau_n \in \gamma(\pi_c(\rho_n))$ .

### A.2 Proof of Theorem 2

**Lemma 4.**  $C \ll D \wedge C' \ll D' \Rightarrow (C \cup C') \ll (D \cup D')$

*Proof.* Text.

*Proof.* We show each direction of the theorem separately.

$\Rightarrow$ : We define a relation  $R^n \subseteq \wp(\mathbb{C}) \times \wp(\mathbb{D})$  as follows:

$$R^n = \cup\{(\{\tau_n\}, \{v_n\}) \mid \tau \in \text{Traj}(M)^\infty : v \in \text{Traj}(N)^\infty : \alpha([\tau]) \subseteq [v]\}$$

where the union of two pairs  $R^n(C, D)$  and  $R^n(C', D')$  is given by the union of its two parts  $R^n(C \cup C', D \cup D')$ . We then define  $\ll = R^0 \cup R^1 \cup \dots$  and show that it satisfies the desired conditions.

$\Leftarrow$ : ...

### A.3 Proof of Theorem ??

First a lemma that shows  $\llbracket \hat{p} \sqcap \hat{\pi}_a(s) \rrbracket \Leftrightarrow \llbracket \hat{p} \rrbracket \sqcap \hat{\pi}_a(s)$  is a reasonable assumption.

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

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

Secondly we show a few helpful lemmas that regard the fix-points and functions used to determine satisfaction with  $\hat{M}$  and  $\hat{N}$ . Before that, we duplicate the earlier definitions of  $F$ ,  $\mathcal{F}$  and  $\Phi$  to differentiate between those used with  $\hat{M}$  and those with  $\hat{N}$ . Specifically, let  $G$ ,  $\mathcal{G}$  and  $\Psi$  be equivalent operations for  $\hat{N}$ , as  $F$ ,  $\mathcal{F}$  and  $\Phi$  are for  $\hat{M}$ :

$$\begin{aligned} G(s)(\hat{q}) &= \hat{N}(\pi_a(s) \sqcap \hat{q}) \\ \mathcal{G}(\Psi)(s) &= \text{if } (s = s_0) \text{ then } \top \text{ else } \sqcup \{G(s')(\Psi(s')) \mid (s', s) \in R\} \\ \Psi_n &= \text{if } (n = 0) \text{ then } (\lambda s \in S : \perp) \text{ else } \mathcal{G}(\Psi_{n-1}) \end{aligned}$$

Furthermore, let  $\Psi_*$  be the least fixpoint of  $\Psi = \mathcal{G}(\Psi)$  and given by  $\lim \Psi_n(s)$ .

**Lemma 6.**  $\perp \lll \perp \wedge \top \lll \top$

A Galois connection always relates the two tops and bottoms, i.e.  $\hat{\alpha}(\top) \sqsubseteq \top$  and  $\hat{\alpha}(\perp) \sqsubseteq \perp$ . Because  $\llbracket \cdot \rrbracket$  preserves both tops and bottoms, it then follows that both  $\hat{\alpha}(\llbracket \top \rrbracket) \sqsubseteq \llbracket \top \rrbracket$ , or  $\top \lll \top$ , and  $\hat{\alpha}(\llbracket \perp \rrbracket) \sqsubseteq \llbracket \perp \rrbracket$ , or  $\perp \lll \perp$ , holds as well.

**Lemma 7.**  $\hat{p} \lll \hat{q} \Rightarrow \forall s \in S : F(s)(\hat{p}) \lll G(s)(\hat{q})$

$$\begin{aligned} \hat{\gamma}(\llbracket G(s)(\hat{q}) \rrbracket) &= \hat{\gamma}(\llbracket \hat{N}(\hat{\pi}_a(s) \sqcap \hat{q}) \rrbracket) && \text{(definition of } G) \\ &\sqsupseteq \llbracket \hat{M}(\hat{\gamma}(\hat{\pi}_a(s) \sqcap \hat{q})) \rrbracket && \text{(simulation relation?)} \\ &= \llbracket \hat{M}(\hat{\gamma}(\hat{\pi}_a(s)) \sqcap \hat{\gamma}(\hat{q})) \rrbracket && (\hat{\gamma} \text{ distributes over } \sqcap) \\ &\sqsupseteq \llbracket \hat{M}(\hat{\gamma}(\hat{\pi}_a(s))) \sqcap \hat{p} \rrbracket && \text{(Galois connection?)} \\ &= \llbracket F(s)(\hat{p}) \rrbracket && \text{(definition of } F) \end{aligned}$$

**Lemma 8.**  $\forall s \in S : \Phi_*(s) \lll \Psi_*(s)$

Since  $\Phi_*(s) = \lim \Phi_n(s)$  and  $\Psi_*(s) = \lim \Psi_n(s)$ , it suffices to prove that  $\Phi_n(s) \lll \Psi_n(s)$  for all  $s \in S$  and  $n \in \mathbb{N}$ . We do so by induction on  $n$ . The base case, where  $\Phi_0(s) = \perp$  and  $\Psi_0(s) = \perp$ , follows from lemma 6. For the inductive step, assume that  $\Phi_n(s) \lll \Psi_n(s)$  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 also follows from lemma 6. For any  $s \neq s_0$ , we have:

$$\begin{aligned}
\hat{\gamma}(\llbracket \Psi_{n+1}(s) \rrbracket) &= \hat{\gamma}(\llbracket \sqcup \{G(s')(\Psi_n(s')) \mid (s', s) \in R\} \rrbracket) && \text{(definition of } \Psi_{n+1} \text{ and } \mathcal{G}) \\
&= \hat{\gamma}(\sqcup \{ \llbracket G(s')(\Psi_n(s')) \rrbracket \mid (s', s) \in R \}) && (\llbracket \cdot \rrbracket \text{ distributes over } \sqcup) \\
&\supseteq \sqcup \{ \hat{\gamma}(\llbracket G(s')(\Psi_n(s')) \rrbracket) \mid (s', s) \in R \} && (\hat{\gamma} \text{ is monotone}) \\
&\supseteq \sqcup \{ \llbracket F(s')(\Phi_n(s')) \rrbracket \mid (s', s) \in R \} && \text{(I.H and lemma 7)} \\
&= \llbracket \sqcup \{ F(s')(\Phi_n(s')) \mid (s', s) \in R \} \rrbracket && (\llbracket \cdot \rrbracket \text{ distributes over } \sqcup) \\
&= \llbracket \Phi_{n+1}(s) \rrbracket && \text{(definition of } \Phi_{n+1} \text{ and } \mathcal{F})
\end{aligned}$$

Consider an arbitrary  $s \in S$  and let  $\hat{p} = \Phi_*(s) \sqcap \hat{\gamma}(\hat{\pi}_a(s))$ , where  $\hat{p} \sqsubseteq \Phi_*(s)$  and  $\hat{p} \sqsubseteq \hat{\gamma}(\hat{\pi}_a(s))$ , or  $\hat{\alpha}(\hat{p}) \sqsubseteq \hat{\pi}_a(s)$ . Lemma 8 tells us  $\Phi_*(s) \lll \Psi_*(s)$ , and thus  $\hat{\alpha}(\hat{p}) \sqsubseteq \hat{\alpha}(\llbracket \hat{p} \rrbracket) \sqsubseteq \hat{\alpha}(\llbracket \Phi_*(s) \rrbracket) \sqsubseteq \llbracket \Psi_*(s) \rrbracket$  by the monotonicity of  $\llbracket \cdot \rrbracket$  and  $\hat{\alpha}$ . Using the invariance of  $\hat{\pi}_a(s)$  and property **name** of  $\llbracket \cdot \rrbracket$ , we note that the assumption can be restated as  $\hat{\pi}_c(s) \supseteq \Psi_*(s) \sqcap \hat{\pi}_a(s) = \llbracket \Psi_*(s) \rrbracket \sqcap \hat{\pi}_a(s) = \llbracket \Psi_*(s) \rrbracket \sqcap \hat{\pi}_a(s)$ . It then follows that  $\alpha(\hat{p}) \sqsubseteq \hat{\pi}_c(s)$ , or  $\hat{p} \sqsubseteq \hat{\gamma}(\hat{\pi}_c(s))$  as desired.

#### A.4 Proof of Theorem ??

Text.

## 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)