Foundations of Software Technology and Theoretical Computer Science:
22nd Conference, Proceedings, M. Agrawal, A. Seth (eds.)
Springer Lecture Notes in Computer Science **2556** (2002) 289–300.

Hereditary history preserving bisimulation is decidable for trace-labelled systems*

Madhavan Mukund

Chennai Mathematical Institute, 92 GN Chetty Road, Chennai 600017, India
Email: madhavan@cmi.ac.in

Abstract. Hereditary history preserving bisimulation is a natural extension of bisimulation to the setting of so-called "true" concurrency. Somewhat surprisingly, this extension turns out to be undecidable, in general, for finite-state concurrent systems. In this paper, we show that for a substantial and useful class of finite-state concurrent systems—those whose semantics can be described in terms of Mazurkiewicz traces—hereditary history preserving is decidable.

1 Introduction

While branching time semantics is relatively well understood in the interleaved approach to the semantics of concurrent systems, the situation is not so clear when labelled partial orders are used to record the behaviour of such systems. If no restriction is placed on the structure of a concurrent system, the interplay between nondeterminism and concurrency results in many seemingly simple problems becoming computationally intractable.

An example of this is the problem of checking whether two finite-state systems are equivalent with respect to the concurrency-preserving branching-time behavioural equivalence known as hereditary history preserving bisimulation. Hereditary history preserving bisimulation is a natural extension of bisimulation, as defined by Park and Milner [11, 13], from a setting where concurrency is equated with nondeterministic interleaving to a richer setting where nondeterminism and concurrency are represented explicitly and independently. Hereditary history preserving bisimulation was first defined by Bednarczyk [1], but became more well known when it reappeared as a natural construction in a general, categorical approach to nondeterminism and concurrency arising out of the work of Winskel and Nielsen[7, 17].

Hereditary history preserving bisimulation requires two concurrent systems to retain the same nondeterministic choices as they evolve, as in conventional bisimulation. In addition, at every state, each of the systems also has to faithfully simulate all steps—sets of pairwise independent actions—performed by other system. Although this seems to be a fairly innocuous extension, the repercussions are quite severe. It turns out that history preserving bisimulation is, in general, undecidable for finite-state concurrent systems [8].

* Partially supported by IFCPAR Project 2102-1 (ACSMV).

A few positive results have been obtained regarding hereditary history preserving bisimulation. In [12], a game-theoretic formulation is presented, along with a characterization in terms of a Hennessy-Milner style modal logic with past modalities. The decidability question was investigated in [5] where some very restricted positive results were obtained.

Earlier, a weaker notion of history preserving bisimulation had been proposed in [3, 15, 16]. Here, too, the two systems have to progressively simulate each others' concurrent steps. However, the bisimulation is built up one action at a time, so each interleaving of a concurrent step in one system may be simulated by a different, incompatible, step in the other system. Thus, from the standpoint of faithfully preserving concurrency and nondeterminism, this notion is slightly unsatisfactory. The decidability of this variety of bisimulation was established in [6].

In this paper, we examine the decidability question afresh for a restricted class of concurrent systems—those whose behaviours can be described by Mazurkiewicz traces. Our main result is that hereditary history preserving bisimulation is decidable for this class of systems.

Mazurkiewicz traces [10] are labelled partial orders generated by independence alphabets of the form (Σ, I), where I is a static *independence* relation over Σ. If $(a, b) \in I$, a and b are deemed to be independent actions that may occur concurrently in any context where they are jointly enabled. Traces are a natural formalism for describing the behaviour of various static networks of communicating finite-state agents as modelled by Petri nets [14] or communicating finite-state automata [18]. Hence, our positive result is applicable to a substantial and useful subclass of finite-state concurrent systems.

The paper is organized as follows. We begin with some basic definitions about labelled Petri nets, the system model that we work with in this paper. In Section 3 we define hereditary history preserving bisimulation. In the next section, we identify the subclass of systems that we focus on—those whose semantics can be defined using Mazurkiewicz traces. In Section 5, we show that hereditary history preserving bisimulation is equivalent to a notion of step bisimulation on step transition systems. This characterization is used to derive the main decidability result in Section 6. We conclude with a brief discussion.

2 Preliminaries

We use labelled 1-safe Petri nets as our basic model of finite-state concurrent systems. This choice of model is not important: we could have, instead, worked with any other model that has an explicit notion of independence or concurrency built in, such as labelled asynchronous transition systems [2], asynchronous automata [18] or transition systems with independence [17].

Nets A *net* is a quadruple (S, T, F, M_{in}) where:

- S is a finite, non-empty set of *places*.
- T is a finite, non-empty set of *transitions*.

– $F \subseteq (S \times T) \cup (T \times S)$ is the *flow relation*.
– $M_{in} : S \to \mathbb{N}_0$ is the *initial marking*, where $\mathbb{N}_0 = \{0, 1, 2, \ldots\}$.

As usual, for x in $S \cup T$, we use ${}^\bullet x$ to denote the set $\{y \in S \cup T \mid (y, x) \in F\}$ and $x^\bullet$ to denote the set $\{y \in S \cup T \mid (x, y) \in F\}$.

Reachable markings A *marking* of (S, T, F, M_{in}) is a function $M : S \to \mathbb{N}_0$ and corresponds to a global state of the system. A transition t is enabled at marking M, denoted $M \xrightarrow{t}$, if $M(s) > 0$ for all $s \in {}^\bullet t$. When t occurs, M evolves to a new marking M', written $M \xrightarrow{t} M'$, where

$$\forall s \in S.\ M'(s) = \begin{cases} M(s) - 1 \text{ if } s \in ({}^\bullet t - t^\bullet), \\ M(s) + 1 \text{ if } s \in (t^\bullet - {}^\bullet t), \\ M(s) \qquad \text{otherwise.} \end{cases}$$

Notice that M' is uniquely fixed by M and t. Let $w = t_1 t_2 \ldots t_k$. We write $M \xrightarrow{w} M'$ to indicate that $M \xrightarrow{t_1} M_1 \xrightarrow{t_2} M_2 \cdots M_{k-1} \xrightarrow{t_k} M'$. Similarly, $M \xrightarrow{w}$ denotes that there exists M' such that $M \xrightarrow{w} M'$.

Let M_0 be a marking of (S, T, F, M_{in}). The set of markings reachable from M_0, denoted $\mathcal{R}(M_0)$ is defined inductively as follows:

– $M_0 \in \mathcal{R}(M_0)$.
– If $M \in \mathcal{R}(M_0)$ and $M \xrightarrow{t} M'$, then $M' \in \mathcal{R}(M_0)$.

1-safe nets The net (S, T, F, M_{in}) is said to be *1-safe* if for all $M \in \mathcal{R}(M_{in})$, for all $s \in S$, $M(s) \leq 1$. Clearly, if N is 1-safe, then its global state space is finite since the number of distinct markings in $\mathcal{R}(M_{in})$ is bounded by $2^{|T|}$. In this paper, we assume that every net we consider is 1-safe.

Labelled nets Let Σ be a set of actions. A Σ-labelled net is a structure $N = (S, T, F, M_{in}, \lambda)$ where (S, T, F, M_{in}) is a (1-safe) net and $\lambda : T \to \Sigma$ is a labelling function.

Independence of transitions We say that transitions t_1 and t_2 are *independent* if they have disjoint neighbourhoods—that is, $({}^\bullet t_1 \cup t_1{}^\bullet) \cap ({}^\bullet t_2 \cup t_2{}^\bullet) = \emptyset$. We write $t_i I_N t_j$ to denote that t_i and t_j are independent in N. If $\neg(t_i I_N t_j)$ we write $t_i D_N t_j$, denoting that t_i and t_j are *dependent*. Independent transitions satisfy forward and sideways diamond properties. Let t_1 and t_2 be independent. If $M \xrightarrow{t_1} M_1$ and $M \xrightarrow{t_2} M_2$ then there exists M' such that $M_1 \xrightarrow{t_2} M'$ and $M_2 \xrightarrow{t_1} M'$. Further, if $M \xrightarrow{t_1} M_1 \xrightarrow{t_2} M'$ then there exists M_2 such that $M \xrightarrow{t_2} M_2 \xrightarrow{t_1} M'$.

Runs Let $N = (S, T, F, M_{in}, \lambda)$ be a labelled net. Each transition sequence $w = t_1 t_2 \ldots t_k$ such that $M_0 \xrightarrow{w} M_w$ gives rise to a labelled partial order that we call a *run*. The run $po(w)$ associated with w is a triple $(E, \leq, \tau)$ where E is a set of events partially ordered by $\leq$ and $\tau : E \to T$ is a labelling function. Each event in E corresponds to a transition from w and is labelled by the underlying transition. The partial order $\leq$ is the reflexive, transitive closure of the relation $e_i \prec e_j \Leftrightarrow i < j \wedge t_i D_N t_j$, where $\tau(e_i) = t_i$ and $\tau(e_j) = t_j$.

More formally, let $w = t_1 t_2 \ldots t_k$. Then, $po(w) = (E, \leq, \tau)$ such that:

(i) $E = \{1, 2, \ldots, k\}$.
(ii) For each $i \in E$, $\tau(i) = t_i$.
(iii) For $i, j \in E$, if i and j are unordered in $po(w)$ then $t_i I_N t_j$.
(iv) For $i, j \in E$, if $i \lessdot j$ then $t_i D_N t_j$, where $\lessdot \; = \; < \setminus <^2$ is the immediate successor relation in $po(w)$.

Let $Runs(N) = \{po(w) \mid w \in T^*, M_{in} \xrightarrow{w}\}$ denote the set of runs of N. With each run $r \in Runs(N)$, we can associate a unique marking M_r, the global state of N after the computation r.

If $r = po(w)$ and $r' = po(wt)$—that is, r' extends r by adding an event corresponding to the transition t—we denote r' as $r + t$. If $u = \{t_1, t_2, \ldots, t_k\}$ is a set of pairwise independent transitions enabled at M_r, then we denote the run $r + t_1 + t_2 + \cdots + t_k$ by $r + u$.

For a run $r = (E, \leq, \tau)$, $\max(r)$ denotes the set of maximal events in E. Let $r = po(w)$, where $w = t_1 t_2 \ldots t_k$, and let $j \in \max(r)$. Then $r - j$ denotes the run obtained by deleting the event j from r—that is, $r - j = po(w')$, where $w' = t_1 t_2 \ldots t_{j-1} t_{j+1} \ldots t_k$.

Two runs are said to be isomorphic if they are isomorphic as labelled partial orders. We write $r \simeq r'$ to indicate that r and r' are isomorphic runs.

For each net, the empty sequence of transitions ε gives rise to the empty run $(\emptyset, \emptyset, \emptyset)$. We use $r_\emptyset$ uniformly to denote the empty run for all nets.

3 History Preserving Bisimulations

For the rest of this section, fix a pair of labelled nets $N_i = (S_i, T_i, F_i, M^i_{in}, \lambda_i)$, $i \in \{1, 2\}$, whose transitions are labelled by a common set of actions Σ. A history preserving bisimulation is a relation between the runs of N_1 and the runs of N_2 which asserts that the two systems have equivalent observable capabilities, even when we take concurrency into account.

We define history preserving bisimulations in terms of matched runs.

Matched runs Let $\rho = (E, \leq, \tau_1, \tau_2)$ be a partial order equipped with two labelling functions such that $\tau_i : E \to T_i$, $i \in \{1, 2\}$, labels each event in E with a transition from T_i. The structure $(E, \leq, \tau_1, \tau_2)$ is a *matched run* of N_1 and N_2 if it satisfies the following conditions:

- $\rho_1 = (E, \leq, \tau_1)$ is a run of N_1 and $\rho_2 = (E, \leq, \tau_2)$ is a run of N_2.
- For all $e \in E$, $\lambda_1(\tau_1(e)) = \lambda_2(\tau_2(e))$.

Let ρ and ρ' be matched runs such that ρ' extends ρ by k events with $\rho'_1 = \rho + u_1$ and $\rho'_2 = \rho_2 + u_2$, where u_1 and u_2 are pairwise disjoint subsets of size k of T_1 and T_2, respectively. We denote ρ' by $\rho + \langle u_1, u_2 \rangle$. If $u_1 = \{t_1\}$ and $u_2 = \{t_2\}$ are singletons, we write $\rho + \langle t_1, t_2 \rangle$ rather than $\rho + \langle \{t_1\}, \{t_2\} \rangle$.

Let e be a maximal event in the matched run ρ. Then $\rho' = \rho - e$ is the matched run obtained by deleting e from ρ. If $u = \{e_1, e_2, \ldots, e_k\}$ is a subset of maximal events in ρ, we write $\rho - u$ to denote the run $(((\rho - e_1) - e_2) - \cdots - e_{k-1}) - e_k$

obtained by deleting all the events from u (clearly, the events in u may be removed in any order).

Hereditary history preserving bisimulation A *hereditary history preserving bisimulation* between $N_1 = (S_1, T_1, F_1, M_{in}^1, \lambda_1)$ and $N_2 = (S_2, T_2, F_2, M_{in}^2, \lambda_2)$ is a set H of matched runs of the two nets such that:

(i) The empty matched run $(\emptyset, \emptyset, \emptyset, \emptyset)$ belongs to H.
(ii) Let $\rho = (E, \leq, \tau_1, \tau_2) \in H$. For each transition $t_1 \in T_1$ enabled at M_{ρ_1}, there is a transition t_2 in T_2 such that $\rho + \langle t_1, t_2 \rangle \in H$.
(iii) Let $\rho = (E, \leq, \tau_1, \tau_2) \in H$. For each transition $t_2 \in T_2$ enabled at M_{ρ_2}, there is a transition t_1 in T_1 such that $\rho + \langle t_1, t_2 \rangle \in H$.
(iv) For each maximal event e in ρ, the matched run $\rho - \{e\}$ is in H.

The first three clauses correspond to the weaker definition of history preserving bisimulations [3, 15, 16]. The last clause strengthens the definition by ensuring that the bisimulation extends concurrent steps in a uniform way, regardless of the order in which the concurrent step is executed. The weaker definition permits different interleavings of the same concurrent step to be simulated in different ways. (The original definition of hereditary history preserving bisimulation required all prefixes of a matched run to belong to the relation H. However, it was shown in [12] that is suffices to check the condition for the substructures obtained by deleting each of the maximal events in the given matched run.) Figure 1, taken from [5], illustrates the difference between the two definitions.

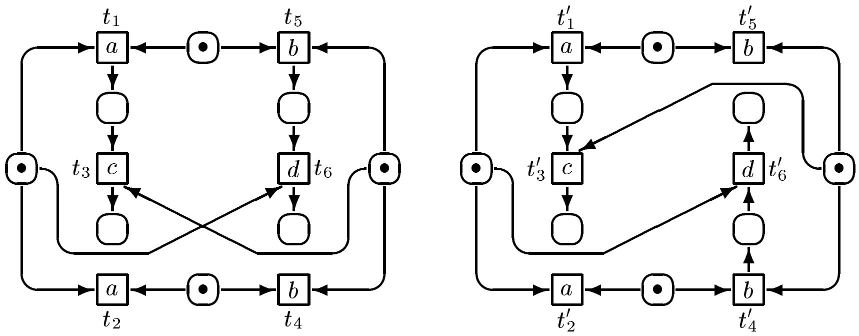

Fig. 1.

In this example, consider the step $\{t_1, t_4\}$ in the net on the left. If we execute t_1 before t_4, we can simulate this by t_1' followed by t_4', preserving forward choices at each point. Similarly, if we execute t_4 before t_1, we can simulate it by t_5' followed by t_2'. However, neither simulation is faithful to the original step and this is caught by the last clause in the definition of hereditary history preserving bisimulations—for instance, if we simulate $t_1 t_4$ by $t_1' t_4'$, we can "undo" the maximal events t_1 and t_1' to reach markings in which a d-labelled transition is enabled on the right but not on the left.

The difficulty with establishing whether two labelled nets are hereditary history-preserving bisimilar is that the bisimulation is defined at the level of matched runs, which correspond to the infinite "unfolded" behaviours of the two nets. For normal sequential bisimulation, given a pair of finite transition systems, a bisimulation relation exists between their unfoldings if and only if a (finite) bisimulation relation exists between the states of the original system.

Every Petri net can be regarded as a labelled transition system whose states correspond to the (reachable) markings of the net. In general, a hereditary history preserving bisimulation at the level of matched runs cannot be "folded" down to a relation the level of markings, as demonstrated in Figure 2. These two nets are hereditary history preserving bisimilar. The markings $\{s_3, s_4\}$ and $\{s'_4, s'_5\}$ are not equivalent after executing $\{t_2, t_3\}$ and $\{t'_1, t'_3\}$. However, if we reach the same markings after executing $\{t_2, t_3, t_4, t_5\}$ and $\{t'_2, t'_3, t'_4, t'_5\}$, the markings do become equivalent.

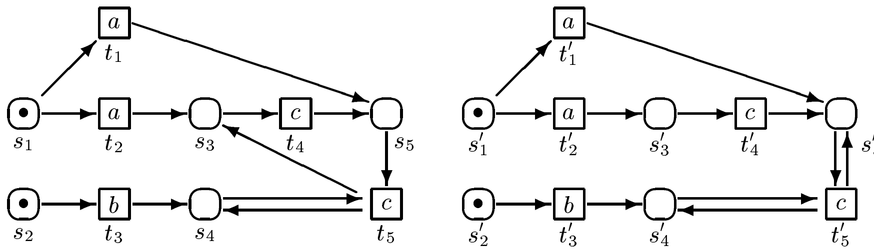

Fig. 2.

4 Trace-labelled systems

Trace alphabets A *trace alphabet* is a pair (Σ, I), where $I \subseteq \Sigma \times \Sigma$ is an irreflexive and symmetric *independence relation*. The complement of I is denoted D and called the *dependence relation*.

The independence relation I induces a natural equivalence relation $\sim$ on words over Σ. Intuitively, two words are related by $\sim$ if we can go from one to another by repeatedly swapping adjacent independent letters. More formally, let $\sim_0$ be the relation $\{(wabw', wbaw') \mid w, w' \in \Sigma^*, aIb\}$. Then $\sim$ is the reflexive, transitive closure of $\sim_0$. The equivalence classes generated by $\sim$ are called *(Mazurkiewicz) traces*. We denote the trace containing the word w by $[w]$.

Trace-labelled nets Let (Σ, I) be a trace alphabet. A *labelled net* $N = (S, T, F, M_{in}, \lambda)$ with $\lambda : T \to \Sigma$ is said to be *trace-labelled* if $t_i I_N t_j \Leftrightarrow \lambda(t_i) I \lambda(t_j)$ for all pairs of transitions $t_i, t_j \in T$.

For trace-labelled nets, the partial order structure of a run is determined solely by the labelling on the underlying transition sequence. Formally, we have the following result.

Proposition 1. *Let $N = (S, T, F, M_{in}, \lambda)$ be a trace-labelled net and let $\rho(w)$ and $\rho(w')$ be two runs of N such that $w = t_1 t_2 \dots t_k$, $w' = t'_1 t'_2 \dots t'_k$ and $\lambda(t_i) = \lambda(t'_i)$ for each $i \in \{1, 2, \dots, k\}$. Then, $\rho(w) \simeq \rho(w')$.*

We omit the proof this result, which is a reformulation of a standard result of trace theory that any linearization of a trace fixes its underlying partial order representation [4].

Observe that the nets in Figure 2 are not trace-labelled. For instance, in the net on the left, the transition sequences $t_1 t_3 t_5$ and $t_2 t_3 t_4$ both generate the same labelled sequence, *abc*, but give rise to non-isomorphic runs.

In Figure 2, one reason that we could not fold down the hereditary history preserving bisimulation relation to a relation on markings was because the independence between actions b and c is context-dependent. This problem is eliminated if we work with trace-labelled nets. Nevertheless, even for trace-labelled nets, we cannot always translate a hereditary history preserving bisimulation relation into a relation on markings. Figure 3 shows the reachable markings of a pair of trace-labelled nets that are hereditary history preserving bisimilar, where aIb. Here, if we initially execute $M_{in} \xrightarrow{b} M_1$ and $M'_{in} \xrightarrow{b} M'_1$, the markings are not equivalent, but after the sequences $M_{in} \xrightarrow{a} M_2 \xrightarrow{b} M_7 \xrightarrow{e} M_1$ and $M'_{in} \xrightarrow{a} M'_3 \xrightarrow{b} M'_6 \xrightarrow{e} M'_1$, the markings are in fact equivalent.

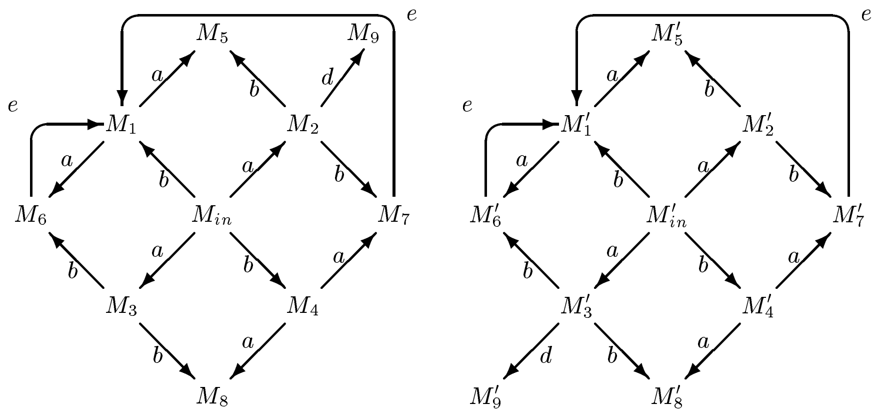

Fig. 3.

5 Synchronized step transition systems

Step transition systems Let (A, I) be a trace alphabet. A *step transition system* over (A, I) is a structure $TS = (Q, \delta, q_{in})$ where Q is a set of states with an initial state $q_{in} \in Q$ and $\delta : Q \times 2^A \to Q$ is a step transition function satisfying the following conditions:

(i) If $\delta(q,u) = q'$, then for each distinct pair $a_i, a_j \in u$, $a_i I a_j$.
(ii) If $\delta(q,u) = q'$, then for all $v \subseteq u$, there exists q_v such that $\delta(q,v) = q_v$ and $\delta(q_v, u \setminus v) = q'$.

Thus, transitions in a step transition system are labelled by sets of pairwise independent actions in such a way that every step can be broken up into all possible substeps. This definition of step transition systems is equivalent to *deterministic* distributed transition systems, as defined in [9].

Notice that for independent actions a and b, it is possible to have transitions $\delta(q,a) = q_a$, $\delta(q,b) = q_b$ and $\delta(q_a,b) = \delta(q_b,a) = q'$, but *not* have the step transition $\delta(q,\{a,b\}) = q'$. A step transition system that does not exhibit such anomalies is said to be coherent.

Coherent step transition systems A step transition system $TS = (Q, \delta, q_{in})$ over (A, I) is said to be coherent if the following holds: whenever $\delta(q,a) = q_a$ and $\delta(q_a, b) = q'$ for aIb, it is the case that $\delta(q,\{a,b\}) = q'$.

Proposition 2. *Let $TS = (Q, \delta, q_{in})$ be a coherent step transition system over (A, I) and let $q \in Q$ be a state that is reachable from the initial state q_{in}. Let $w \in A^*$ be a word such that $\delta(q_{in}, w) = q$. For every $w' \sim w$, $\delta(q_{in}, w') = q$ (recall that $\sim$ is the trace equivalence relation induced by I).*

The proof, which we omit, is by induction on the number of times we have to apply the immediate trace equivalence $\sim_0$ to go from w to w'.

Synchronous step transition systems Let $N_i = (S_i, T_i, F_i, M^i_{in}, \lambda_i)$, $i \in \{1,2\}$ be a pair of labelled Petri nets. We can extend the independence relations I_{N_1} on T_1 and I_{N_2} on T_2 to an independence relation $I_{N_{12}}$ on $T_1 \times T_2$ as follows: $(t_1,t_2) I_{N_{12}} (t'_1, t'_2)$ if and only if $t_1 I_{N_1} t'_1$ and $t_2 I_{N_2} t'_2$.

A *joint step transition system* for N_1 and N_2 is a step transition system $TS = (Q, \delta, q_{in})$ over $(T_1 \times T_2, I_{N_{12}})$ where:

- $Q = \{(r_1, r_2) \mid r_1 \in \mathit{Runs}(N_1), r_2 \in \mathit{Runs}(N_2), r_1 \simeq r_2\}$.
- $q_{in} = (r_\emptyset, r_\emptyset)$ (recall that for any net, $r_\emptyset$ denotes the empty run).
- If $\delta((r_1,r_2),(u_1,u_2)) = (r'_1, r'_2)$ then $r_1 + u_1 = r'_1$ and $r_2 + u_2 = r'_2$.

A *synchronous step transition system* for N_1 and N_2 is a joint step transition system $TS = (Q, \delta, q_{in})$ for N_1 and N_2 that satisfies the following conditions:

- TS is coherent.
- If $(r_1, r_2) \in Q$ and there is a run $r'_1 \in \mathit{Runs}(N_1)$ and a subset $u_1 \subseteq T_1$ such that $r'_1 = r_1 + u_1$, then there exists $r'_2 \in \mathit{Runs}(N_2)$ and $u_2 \subseteq T_2$ such that $\delta((r_1,r_2),(u_1,u_2)) = (r'_1, r'_2)$.
- If $(r_1, r_2) \in Q$ and there is a run $r'_2 \in \mathit{Runs}(N_2)$ and a subset $u_2 \subseteq T_2$ such that $r'_2 = r_2 + u_2$, then there exists $r'_1 \in \mathit{Runs}(N_1)$ and $u_1 \subseteq T_1$ such that $\delta((r_1,r_2),(u_1,u_2)) = (r'_1, r'_2)$.

Intuitively, a synchronous step transition system for N_1 and N_2 corresponds to a bisimulation between the unfoldings of the two nets in which each step in one net is simulated uniformly by a step from the other net.

Lemma 3. *Let $N_i = (S_i, T_i, F_i, M^i_{in}, \lambda_i)$, $i \in \{1,2\}$ be a pair of trace-labelled Petri nets. There is hereditary history preserving bisimulation between N_1 and N_2 if and only if there is a synchronous step transition system for N_1 and N_2.*

Proof. ($\Rightarrow$) Let H be a hereditary history preserving bisimulation between N_1 and N_2. We construct a synchronous step transition system $TS = (Q, \delta, q_{in})$ for N_1 and N_2 as follows:

- $Q = \{(\rho_1, \rho_2) \mid \rho \in H\}$.
- $q_{in} = (r_\emptyset, r_\emptyset)$.
- $\delta((\rho_1, \rho_2), (u_1, u_2)) = (\rho'_1, \rho'_2)$ if $\rho, \rho' \in H$ with $\rho + \langle u_1, u_2 \rangle = \rho'$.

We first verify that TS is coherent. Suppose that $\delta((\rho_1, \rho_2), (t'_1, t'_2)) = (\rho'_1, \rho'_2)$ and $\delta((\rho'_1, \rho'_2), (t''_1, t''_2)) = (\rho''_1, \rho''_2)$ such that $t'_1 I_1 t''_1$ and $t'_2 I_2 t''_2$. Then, $\rho'' = \rho + \langle \{t'_1, t''_1\}, \{t'_2, t''_2\} \rangle$ so, by the definition of δ, $\delta((\rho_1, \rho_2), (\{t'_1, t''_1\}, \{t'_2, t''_2\})) = (\rho''_1, \rho''_2)$, as required by coherence.

Next, we check forward extensibility. Suppose that (ρ_1, ρ_2) in TS and $\rho_1 + u_1 = \rho'_1 \in Runs(TS)$. Let $u_1 = \{t_1, t_2, \ldots, t_k\}$. Since $\rho \in H$, there must exist transitions $t'_1, t'_2, \ldots, t'_k \in T_2$ such that $\rho + \langle t_1, t'_1 \rangle + \langle t_2, t'_2 \rangle + \cdots + \langle t_k, t'_k \rangle = \rho'' \in H$. Setting $u_2 = \{t'_1, t'_2, \ldots, t'_k\}$, it follows that $\delta((\rho_1, \rho_2), (u_1, u_2)) = (\rho''_1, \rho''_2)$. Clearly, $\rho''_1 = \rho'_1$, so the required transition is present in δ.

($\Leftarrow$) Conversely, let $TS = (Q, \delta, q_{in})$ be a synchronous step transition for N_1 and N_2. Without loss of generality, we assume that every state $(r_1, r_2) \in Q$ is reachable from the initial state $(r_\emptyset, r_\emptyset)$—if there is an unreachable state, we can remove it from TS without affecting either the coherence or the bisimulation characteristics of TS.

Let $H = \{\rho \mid (\rho_1, \rho_2) \in Q\}$. We claim that H is a hereditary history preserving bisimulation. It is easy to verify that H satisfies the two forward extensibility conditions for hereditary history preserving bisimulations.

What we need to establish is the backward closure condition—if $\rho \in H$ and e is a maximal event in ρ, we must argue that $\rho - e$ also belongs to H. The state (ρ_1, ρ_2) in TS corresponding to ρ is reachable from $(r_\emptyset, r_\emptyset)$. Let $w = (t^1_1, t^2_1), (t^2_1, t^2_2), \ldots, (t^k_1, t^k_2)$ be a sequence of transitions such that $\delta((r_\emptyset, r_\emptyset), w) = (\rho_1, \rho_2)$. The maximal event e in ρ corresponds to some pair of transitions (t^j_1, t^j_2) in this sequence. Since TS is coherent, by Proposition 2, for every reordering w' of w that is consistent with $I_{N_{12}}$, $\delta((r_\emptyset, r_\emptyset), w') = (\rho_1, \rho_2)$. Since e is a maximal event in ρ, there is a reordering of w of the form $w' = u \cdot (t^j_1, t^j_2)$. Let $(r^u_1, r^u_2) = \delta((r_\emptyset, r_\emptyset), u)$. Then, $\delta((r^u_1, r^u_2), (t^j_1, t^j_2)) = (\rho_1, \rho_2)$. It follows, therefore, that the matched run $\rho^u \in H$ such that $\rho^u_1 = r^u_1$ and $\rho^u_2 = r^u_2$ is the matched run obtained by removing e from ρ. $\square$

6 Decidability of strong step bisimulation

Recall that our goal is to show that hereditary history preserving bisimulation is decidable for 1-safe trace-labelled nets. From the previous section, it suffices to check whether a pair of trace-labelled nets admits a synchronous step transition system. It turns out that this amounts to checking for the existence of a state-based step bisimulation between a pair of finite step transition systems.

Trace-relabelling Let (A, I_A) and (B, I_B) be a pair of trace alphabets. A trace-relabelling of (A, I_A) by (B, I_B) is a function $\lambda : A \to B$ such that aI_Aa' if and only if $\lambda(a)I_B\lambda(a')$.

Step bisimulations For $i \in \{1, 2\}$, let $TS_i = (Q_i, \delta_i, q^i_{in})$ be a step transition system over trace alphabet (A_i, I_i), and let $\lambda_i : A_i \to B$ be a trace-relabelling of (A_i, I_i) by a third trace alphabet (B, I_B). A *step bisimulation* between TS_1 and TS_2 is a relation $R \subseteq Q_1 \times Q_2$ such that:

(i) $(q^1_{in}, q^2_{in}) \in R$.
(ii) $(q_1, q_2) \in R$ and $\delta_1(q_1, u_1) = q'_1$ implies there exists u_2 such that $\lambda_1(u_1) = \lambda_2(u_2)$ and $(q'_1, \delta_2(q_2, u_2)) \in R$.
(iii) $(q_1, q_2) \in R$ and $\delta_2(q_2, u_2) = q'_2$ implies there exists u_1 such that $\lambda_1(u_1) = \lambda_2(u_2)$ and $(\delta_1(q_1, u_1), q'_2) \in R$.
(iv) Let $(q_1, q_2) \in R$ and $(\delta_1(q_1, u_1), \delta_2(q_2, u_2)) \in R$ for u_1, u_2 such that $\lambda_1(u_1) = \lambda_2(u_2) = u$. For each $v \subseteq u$, $(\delta(q_1, \lambda_1^{-1}(v)), \delta(q_2, \lambda_2^{-1}(v))) \in R$.

Thus, a step bisimulation is just a bisimulation with respect to an additional level of labelling that respects substeps. Conditions (ii) and (iii) are the normal forward extensibility criteria for bisimulation, extended to steps. The fourth condition ensures that R is closed under substeps.

Run foldings Let (Σ, I) be a trace alphabet and let $N = (S, T, F, M_{in}, \lambda)$ be a trace-labelled net over Σ. For each run $r \in Runs(N)$, recall that M_r denotes the marking associated with r. Let $top_r \subseteq \Sigma$ denote the labels associated with the maximal transitions in r. Clearly, top_r is a subset of Σ in which all actions are pairwise independent. The *run folding* of N is the step transition system $TS = (Q, \delta, q_{in})$ over (T, I_N) where:

- $Q = \{(M_r, top_r) \mid r \in Runs(N)\}$.
- $q_{in} = (M_{in}, \emptyset)$.
- For all $u \subseteq T$ such that $M_r \xrightarrow{u}$, $\delta((M_r, top_r), u) = (M_{r+u}, top_{r+u})$.

Observe that the run folding of N is a finite step transition system. The transition function δ is well-defined because N is trace-labelled—it is not difficult to see that for any pair of runs r, r' such that $M_r = M_{r'}$, $top_r = top_{r'}$ and $M_r \xrightarrow{u}$, $top_{r+u} = top_{r'+u}$.

Lemma 4. *Let (Σ, I) be a trace alphabet and $N_i = (S_i, T_i, F_i, M^i_{in}, \lambda_i)$, $i = \{1, 2\}$ be a pair of trace-labelled nets over Σ. Then, N_1 and N_2 admit a synchronized step transition system if and only if there exists a step bisimulation between the run foldings of N_1 and N_2 with trace-relabellings λ_1 and λ_2 from T_1 and T_2 into Σ, respectively.*

Proof. $(\Rightarrow)$ Let $TS = (Q, \delta, q_{in})$ be a synchronized step transition system for N_1 and N_2 and let $TS_1 = (Q_1, \delta_1, q^1_{in})$ and $TS_2 = (Q_2, \delta_2, q^2_{in})$ be the run foldings of N_1 and N_2, respectively. Recall that each state in TS is of the form (r_1, r_2), where $r_1 \in Runs(N_1)$ and $r_2 \in Runs(N_2)$ and each state in TS_i, $i \in \{1,2\}$, is of the form (M_r, top_r) for some run $r \in Runs(N_i)$.

We define a relation $R \subseteq Q_1 \times Q_2$ as follows:

$R = \{((M_1, A_1), (M_2, A_2)) \mid (r_1, r_2) \in Q, M_i = M_{r_i}, A_i = top_{r_i}, i \in \{1,2\}\}$

The fact that R is a step bisimulation between TS_1 and TS_2 follows immediately from the definition of a synchronized step transition system for N_1 and N_2. The forward extensibility conditions for R follow from the extensibility criteria for TS. The substep closure of R follows from the coherence of TS.

($\Leftarrow$) Conversely, suppose that R is a step bisimulation between TS_1 and TS_2. We have to construct a synchronized step transition system TS for N_1 and N_2.

We construct TS inductively, maintaining the invariant that for every state (r_1, r_2) that we add to TS, the corresponding pair of states $((M_{r_1}, top_{r_1}), (M_{r_2}, top_{r_2}))$ belongs to R.

We begin with the initial state $(r_\emptyset, r_\emptyset)$ that corresponds to the pair $((M^1_{in}, \emptyset), (M^2_{in}, \emptyset))$ which is guaranteed to belong to R.

Let (r_1, r_2) be a state in TS. We know that $((M_{r_1}, top_{r_1}), (M_{r_2}, top_{r_2})) \in R$. For each pair (u_1, u_2) such that $(\delta_1((M_{r_1}, top_{r_1}), u_1), \delta_2((M_{r_2}, top_{r_2}), u_2)) \in R$, add the state $(r_1 + u_1, r_2 + u_2)$ to TS.

It is easy to see that TS is a joint step transition system for N_1 and N_2. To check that it is, in fact, a synchronized step transition system for N_1 and N_2, we must establish coherence and the forward extensibility properties.

We first check coherence. Suppose that $\delta((r_1, r_2), (t_1, t_2)) = (r'_1, r'_2)$ and $\delta((r'_1, r'_2), (t'_1, t'_2)) = (r''_1, r''_2)$ where $(t_1, t_2) I_{N_{12}} (t'_1, t'_2)$. We need to show that $\delta((r_1, r_2), \{(t_1, t_2), (t'_1, t'_2)\}) = (r''_1, r''_2)$ as well. From the construction of TS, we know that both $((M_{r_1}, top_{r_1}), (M_{r_2}, top_{r_2}))$ and $((M_{r''_1}, top_{r''_1}), (M_{r''_2}, top_{r''_2}))$ belong to R. From the definition of run foldings, we have $\delta_i((M_{r_i}, top_{r_i}), \{t_i, t'_i\}) = (M_{r''_i}, top_{r''_i})$ for $i \in \{1,2\}$. Hence, $\delta((r_1, r_2), \{(t_1, t_2), (t'_1, t'_2)\}) = (r''_1, r''_2)$.

The forward extensibility properties for TS follow naturally from the properties of step bisimulations, so we omit the details. $\square$

Theorem 5. *Hereditary history preserving bisimulation is decidable for trace-labelled systems.*

Proof. From Lemmas 3 and 4, it follows that a pair of nets N_1 and N_2 admit a hereditary history preserving bisimulation if and only if there is a step bisimulation between their run foldings. It is clear that the existence of a step bisimulation between two finite step transition systems can be checked (exhaustively, if nothing else). Since the run foldings of N_1 and N_2 are guaranteed to be finite step transition systems, the result follows. $\square$

7 Discussion

As we mentioned at the outset, while branching time semantics is relatively well understood in the interleaved approach to the semantics of concurrent systems, the situation is not so clear when labelled partial orders are used to record the behaviour of such systems. In many contexts, the application of Mazurkiewicz trace theory provides a context in which apparently intractable problems become solvable. Our result here is one example of this phenomenon. This seems to suggest that it might make sense to restrict our attention to trace-labelled systems, at least initially, when attempting to build up our understanding of the interplay between nondeterminism and concurrency.

References

1. M.A. Bednarczyk: Hereditary history preserving bisimulation or what is the power of the future perfect in program logics. Technical report, Polish Academy of Sciences, Gdansk (1991).
2. M.A. Bednarczyk: Categories of asynchronous systems, PhD Thesis, Report 1/88, Computer Science, University of Sussex (1988).
3. P. Degano, R. de Nicola and U. Montanari: Partial ordering descriptions and observations of nondeterministic concurrent processes, in *Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency*, Springer LNCS **354** (1989) 438–466.
4. V. Diekert and G. Rozenberg (eds), *The Book of Traces*, World Scientific, Singapore (1995).
5. S.B. Fröschle and T.T. Hildebrandt: On plain and hereditary history-preserving bisimulation, *Proc MFCS '99*, Springer LNCS **1672** (1999) 354–365.
6. L. Jategaonkar and A. Meyer: Deciding true concurrency equivalence on safe, finite nets, *Theoretical Computer Science*, **154**(1) (1996) 107–143.
7. A. Joyal, M. Nielsen and G. Winskel: Bisimulation from open maps, *Information and Computation*, **127**(2) (1996) 164–185.
8. M. Jurdzinski and M. Nielsen: Hereditary history preserving bisimilarity is undecidable, in *Proc. STACS 2000* Springer LNCS **1770** (2000) 358–369.
9. K. Lodaya, R. Parikh, R. Ramanujam and P.S. Thiagarajan: A Logical Study of Distributed Transition Systems, *Information and Computation* **119**(1) (1995) 91–118.
10. A. Mazurkiewicz: Basic notions of trace theory, in: J.W. de Bakker, W.-P. de Roever, G. Rozenberg (eds.), *Linear time, branching time and partial order in logics and models for concurrency*, *LNCS*, **354** (1989) 285–363.
11. R. Milner: *Communication and Concurrency*, Prentice-Hall, London (1989).
12. M. Nielsen and C. Clausen: Games and Logics for a Noninterleaving Bisimulation, *Nordic Journal of Computing* **2**(2) (1995) 221–249.
13. D. Park: Concurrency and automata on infinite sequences, in *Proc. Theoretical Computer Science: 5th GI Conference*, Springer LNCS **104** (1981).
14. W. Reisig and G. Rozenberg (eds.): *Lectures on Petri Nets, Vols I and II Springer LNCS* **1492,1493** (1998).
15. A. Rabinovitch and B.A. Trakhtenbrot: Behavior strucures and nets, *Fundamenta Informaticae*, **11**(4) (1988) 357–403.
16. R. van Glabbeek and U. Goltz: Refinement of actions and equivalence notions for concurrent systems, *Proc MFCS '89*, Springer LNCS **379** (1989) 237–248.
17. G. Winskel and M. Nielsen: Models for concurrency, in S. Abramsky, D. Gabbay and T.S.E. Maibaum, eds, *Handbook of Logic in Computer Science*, *Vol 4*, Oxford (1995) 1–148.
18. W. Zielonka: Notes on finite asynchronous automata, *R.A.I.R.O.—Inform. Théor. Appl.*, **21** (1987) 99–135.