The decidability border of hereditary history preserving bisimilarity

S. B. Fröschle [1]

Institute of Informatics, University of Warsaw, Poland

Key words: concurrency, true-concurrency, bisimulation games, infinite-space verification

---

In 2000 Jurdziński and Nielsen proved that *hereditary history preserving bisimilarity* (*hhp-b*) is undecidable for finite systems [7,8], and thereby resolved a long-standing open problem. The negative outcome contrasts the weaker *history preserving bisimilarity* (*hp-b*), for which decidability is well-established [15,6]. A definition of (h)hp-b on labelled 1-safe Petri nets (*net systems*) follows.

A *labelled net* N is a tuple (P_N, T_N, F_N, l_N), where P_N is the set of places (depicted as circles), T_N is the set of transitions (depicted as boxes), $F_N : (P_N \times T_N) \cup (T_N \times P_N) \to \{0,1\}$ is the flow relation, and $l_N : T_N \to Act$ is the labelling function, where Act is a set of actions. (In our examples, l_N is determined by the transition identifiers: e.g. a, a_1, and a'_1 are labelled by a.) The pre-set of $x \in P_N \cup T_N$, ${}^\bullet x$, is $\{y \mid F_N(y,x) = 1\}$, the post-set of x, $x^\bullet$, similarly is $\{y \mid F_N(x,y) = 1\}$. A marking M of N is a map $P_N \to \mathbb{N}_0$ (depicted as tokens in places). M enables $t \in T_N$ iff $\forall p \in P_N.\ M(p) \geq F(p,t)$. If M enables t then t can occur, denoted by $M \xrightarrow{t} M'$, where M' is the resulting marking: $\forall p \in P_N.\ M'(p) = M(p) - F(p,t) + F(t,p)$. We extend this notation to sequences $w \in T_N^*$ as usual. A *labelled Petri net* $\mathcal{N}$ is a pair (N, M_0), where N is a labelled net and M_0 is the initial marking. A marking M is reachable in $\mathcal{N}$, $M \in Reach(\mathcal{N})$, iff $M_0 \xrightarrow{w} M$ for some $w \in T_N^*$. $\mathcal{N}$ is *1-safe* iff $\forall M \in Reach(\mathcal{N}).\ \forall p \in P_N.\ M(p) \leq 1$.

Let $\mathcal{N}$ be a net system. We associate an *independence relation*, $I_N \subseteq T_N \times T_N$, with N: $t\ I_N\ t'$ iff $({}^\bullet t \ \cup\ t^\bullet) \cap ({}^\bullet t' \ \cup\ t'^\bullet) = \emptyset$. We allow ourselves to remove pairs (t, t') from I_N if t and t' can never occur concurrently, i.e. if

---

Email address: `sib@mimuw.edu.pl` (S. B. Fröschle).

URL: `http://homepages.inf.ed.ac.uk/sib` (S. B. Fröschle).

[1] This work was supported by EPSRC Grant GR/R16891 and the European Community Research Training Network 'GAMES'.

$\not\exists M \in \mathit{Reach}(\mathcal{N})$ such that M enables both t and t'. $r = t_1 t_2 \ldots t_n \in T_N^*$ is a *run* of $\mathcal{N}$, $r \in \mathit{Runs}(\mathcal{N})$, iff $M_0 \xrightarrow{r} M$ for some M. We write $|r|$ for the length of r, that is $|r| = n$. We associate a strict order with r. Let $<_r$ be the transitive closure of the following 'proximate cause' relation: let i, $j \in |n|$; the ith transition proximately causes the jth in r iff $i < j$ and $\neg(t_i \; I_N \; t_j)$. The ith transition is a *maximal cause* of the jth in r, $i \in \mathit{mcauses}(r, j)$, iff $i <_r j$ and there is no $k \in |n|$ such that $i <_r k <_r j$. Let $k \in |n|$; the kth transition is *backtrack enabled* in r, $k \in \mathit{BEn}(r)$, iff $t_k \; I_N \; t_l$ for all $l \in [k+1, n]$. We define $\delta(r, k)$ to be the result of deleting the kth transition in r, that is $\delta(r, k) = t_1 \ldots t_{k-1} t_{k+1} \ldots t_n$.

The *(h)hp-b game* between Spoiler and Duplicator on two net systems $\mathcal{N}_1$, $\mathcal{N}_2$ is played as follows. Positions are pairs $(r_1, r_2) \in \mathit{Runs}(\mathcal{N}_1) \times \mathit{Runs}(\mathcal{N}_2)$. The initial position is $(\varepsilon, \varepsilon)$. A play proceeds by the following rules: (1) Spoiler chooses one of $\mathcal{N}_1$ or $\mathcal{N}_2$, say $\mathcal{N}_1$, and picks a transition t_1 that is enabled at r_1. Duplicator has to respond by executing a transition t_2 that is enabled in the opposite system such that $l_1(t_1) = l_2(t_2)$ and $\mathit{mcauses}(r_1 t_1, |r_1 t_1|) = \mathit{mcauses}(r_2 t_2, |r_2 t_2|)$. Play continues at $(r_1 t_1, r_2 t_2)$. (2) In the hhp-b game, Spoiler may alternatively perform a backtrack move: Spoiler chooses one of $\mathcal{N}_1$ or $\mathcal{N}_2$, say $\mathcal{N}_1$; he picks $k \in \mathit{BEn}(r_1)$, and backtracks the kth transition in r_1. Duplicator has to respond by backtracking the kth transition in r_2. Play resumes at $(\delta(r_1, k), \delta(r_2, k))$. (3) The play continues like this forever, in which case Duplicator wins, or until either Spoiler or Duplicator is unable to move, in which case the other participant wins. $\mathcal{N}_1$ and $\mathcal{N}_2$ are *(h)hp bisimilar* iff Duplicator has a winning strategy in the (h)hp-b game.

While hp-b is consistent with the classical branching-time view, where a system is modelled as a tree of possible behaviour, (Rule (1) only allows Spoiler to check for concurrency along a fixed run) hhp-b goes beyond this view: by allowing Spoiler to backtrack, the game is taken to the truly-concurrent unfolding level (think event structures), where the relationship of transitions concerning concurrency and conflict is globally captured. This is the root cause of the higher power of hhp-b. Results of [12,13,10] already suggest that problems which exploit the truly-concurrent unfolding level are hard to compute: such problems seem strong enough to encode tiling problems, and hence the computations of Turing machines, in a relatively straightforward sense. The undecidability of hhp-b strengthens this trend. One key insight of the proof is: *(A) A domino system D, to be played on the $\omega \times \omega$ grid, is universally encoded by a finite system $\mathcal{N}(D)$ such that the building of a domino snake can be faithfully mimicked by a special pattern of forwards and backtrack moves in the unfolding structure of $\mathcal{N}(D)$.* $\mathcal{N}(D)$ is a variant of a gadget exhibited in [10]. Motivated by (A) a new tiling connectability problem, *domino bisimilarity,* is introduced, designed to reduce to hhp-b. A second key insight of the proof is: *(B) Domino bisimilarity is undecidable by a reduction from the halting problem of 2-counter machines.*

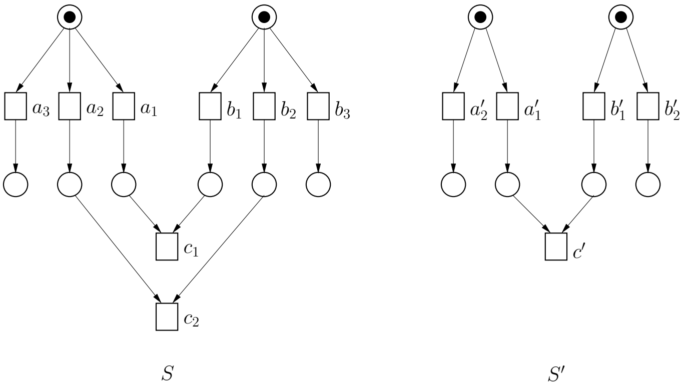

Fig. 1. Counter-example 1

To achieve (A) it seems crucial that the systems under study are free to engage in a complex but variable interplay between conflict and concurrency. This is confirmed as follows. First, hhp-b loses its power for system classes with a tree-like unfolding structure: hhp-b is decidable for BPP [2]; hhp-b coincides with hp-b for ‘concurrency-degree finite’ communication-free net systems and simple BPP [3] (and by [9] this means hhp-b is polynomial-time decidable here). Second, hhp-b is decidable for finite systems with transitive independence relation [5]. And third, hhp-b is decidable for finite bounded-asynchronous systems, which are tightly synchronized in that no thread can be left behind indefinitely [5]. Simplified, $\mathcal{N}(D)$ encodes the $\omega \times \omega$ grid by two loops of independent transitions: $x^n || y^m$ encodes (n, m). This brings about unbounded asynchrony. At each grid point (n, m) and for each choice of domino d there is a transition $d_{n,m}$ sticking out of the grid. This introduces synchronization. To enable the mimicking of domino snakes, the domino transitions are interwoven by concurrency and conflict, among themselves: compatibility between dominoes translates into independence between domino transitions of neighbouring grid points; as well as with the grid: domino transitions can wander off a little from their grid point to let us check for compatibility with neighbouring domino transitions. This brings about non-transitivity of independence. Further, it brings about the scenario *confusion*: two concurrently enabled transitions t_1, t_2 such that the occurrence of t_2 changes the set of transitions that are in conflict with t_1. Indeed, it has long been believed that hhp-b coincides with hp-b for *free-choice systems*, which exclude confusion [1]. $\mathcal{N}$ is free-choice iff for each arc $(p, t) \in F_N \cap (P_N \times T_N)$ $p^\bullet = \{t\}$ or ${}^\bullet t = \{p\}$. However, Counter-example 1 (Figure 1) demonstrates:

Theorem 1 *Hp-b and hhp-b do not coincide, in general, for free-choice systems.*

Spoiler wins the hhp-b game by the move $a_1 b_2$, backtrack a_1, $a_2 c_2$. In the hp-b game Spoiler’s first move resolves whether a c-transition can occur, and, if so, which a and b will act as its causes; Duplicator can orientate her match accordingly. Hhp-b may well be decidable here, but there is also a contrasting

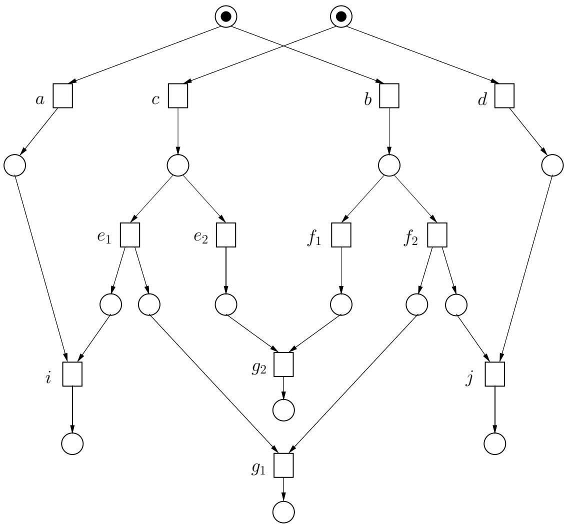

Fig. 2. Counter-example 2, System S

intuition: since the free-choice condition acts only locally there is a sense in which free-choice systems can simulate any mixture of concurrency and conflict. This is no longer true if we additionally require that every transition can always be made to occur again. Indeed, it has been shown: *coherent* hhp-b (chhp-b) is decidable for a subclass of *live* free-choice systems; chhp-b, hhp-b, and hp-b all coincide for a slightly more restricted class [3]. Given these results, one would conjecture hhp-b is decidable for live free-choice systems.

A second aspect to explore is the role of the labelling. In [8] it is shown that domino bisimilarity and hhp-b do not lose their power when we impose that every domino, respectively transition, is labelled by the same action: domino bisimilarity reduces to unlabelled domino bisimilarity, which reduces to unlabelled hhp-b. This is in analogy with results of the interleaving world [14]. On the other hand, there are certain aspects, intrinsically provided by the unlabelled versions, which seem crucial for (B). The reduction from the halting problem to domino bisimilarity employs two tiling systems each of which is a combination of several variants of a primary tiling system (which encodes a 2-counter machine). This brings about *nondeterminism*: there is choice between two transitions with the same label at some state; as well as a certain interplay between labelling and independence: the corresponding systems are not *trace-labelled*: there is no $I \subseteq Act \times Act$ such that $\forall t, t' \in T_N.\ t\, I_N\, t'$ iff $l_N(t)\, I\, l_N(t')$.

Hhp-b does lose its power when we exclude nondeterminism: Duplicator has at most one winning strategy in the hp-b game, which, if it exists, is also a winning strategy in the hhp-b game [3]. This also implies domino bisimilarity is decidable for domino systems with injective labelling: $\mathcal{N}(D)$ is deterministic for such D. In contrast, Counter-example 1 further demonstrates:

Theorem 2 *Hp-b and hhp-b do not coincide, in general, for trace-labelled*

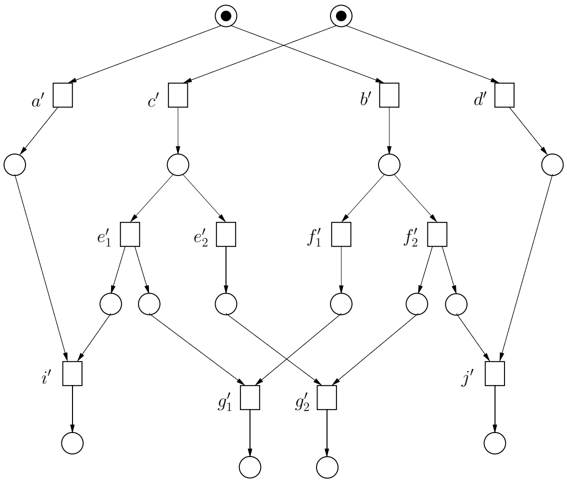

Fig. 3. Counter-example 2, System S'

free-choice systems.

Although it is claimed in [11] that hhp-b is decidable for finite trace-labelled systems the following will show that this is an open problem after all. [11] suggests that on this class hhp-b is captured by allowing Spoiler to backtrack within *concurrent steps*, that is sequences of pairwise independent transitions, only. The corresponding concept *step bisimilarity*[2] does not adequately capture this idea, and consequently the proof of ‘hhp-b implies step bisimilarity’ is incorrect (and it seems unlikely that this direction can be achieved otherwise; c.f. [4] for details). The corrected version *hp-b plus backtracking within matched steps* (*btstep-b*) is played as follows: Spoiler chooses one of $\mathcal{N}_1$ or $\mathcal{N}_2$, say $\mathcal{N}_1$, and performs a concurrent step v_1 that is enabled at r_1. Duplicator has to respond by executing a concurrent step v_2 that is enabled at r_2 such that v_2 matches v_1 with respect to labelling and maximal causes. At this point, and only at this point, Spoiler is allowed to backtrack any set of transitions $u_1 \subseteq v_1$ in $r_1 v_1$; Duplicator has to backtrack the corresponding transitions in $r_2 v_2$, and play continues at the new position. It is routine to check that hhp-b does imply btstep-b. However, Counter-example 2 (Figure 2 and 3) demonstrates that [11] cannot easily be repaired:

Theorem 3 *Btstep-b and hhp-b do not coincide, in general, for trace-labelled free-choice systems.*

To win the hhp-b game Spoiler performs $ce_1 bf_2$: after ce_1 there is the potential of i, which forces Duplicator to match e_1 by e'_1; after $ce_1 bf_2$ transition g_1 is enabled, and Duplicator has to match f_2 by f'_1. But then Spoiler can backtrack e_1 and c, perform d, and expose the j. To detect the difference between S and S', Spoiler needs to backtrack over two causally dependent transitions such as bf_2. But this is not possible in the btstep-b game.

---

[2] This equivalence is finer than what is usually referred to by ‘step bisimilarity’.

A slight variation of Counter-example 2 refines our understanding of the role of nondeterminism: replace the one preplace of e_1 and e_2 by two: one for e_1, one for e_2; proceed similarly for the f's and in S'. The two systems are hp-b but not hhp-b. An instance of nondeterministic choice, say between t_1 and t_2, can arise because t_1 and t_2 are in conflict, or because t_1 and t_2 are concurrent. The undecidability proof and all counter-examples exhibited previously leave open whether the second type of nondeterminism, that is *auto-concurrency*, is at all relevant for the power of hhp-b. The modified counter-example proves that it is indeed: we can employ auto-concurrency rather than nondeterministic conflict to distinguish between hp-b and hhp-b. The simple result demonstrates that true-concurrency may realize its power in many different ways. This makes borderline investigations of truly-concurrent problems more difficult, but also highlights how intriguiging true-concurrency can be and that it deserves further investigation.

Acknowledgements. I thank Madhavan Mukund for checking my argumentation concerning hhp-b on trace-labelled systems. I also thank the editor and the anonymous referees for their constructive recommendations.

References

[1] A. Cheng. *Reasoning About Concurrent Computational Systems.* PhD thesis, BRICS, University of Aarhus, 1996.

[2] S. Fröschle. Decidability of plain and hereditary history-preserving bisimulation for BPP. In *EXPRESS'99*, volume 27 of *ENTCS*, 1999.

[3] S. Fröschle. *Decidability and Coincidence of Equivalences for Concurrency.* PhD thesis, University of Edinburgh, 2004.

[4] S. Fröschle. The decidability of hereditary history preserving bisimilarity on trace-labelled systems is unresolved. Technical Report EDI-INF-RR-0231, University of Edinburgh, 2004.

[5] S. Fröschle and T. Hildebrandt. On plain and hereditary history-preserving bisimulation. In *MFCS'99*, volume 1672 of *LNCS*, pages 354–365. 1999.

[6] L. Jategaonkar and A. R. Meyer. Deciding true concurrency equivalences on safe, finite nets. *TCS*, 154(1):107–143, 1996.

[7] M. Jurdziński and M. Nielsen. Hereditary history preserving bisimilarity is undecidable. In *STACS'00*, volume 1770 of *LNCS*, pages 358–369. 2000.

[8] M. Jurdziński, M. Nielsen, and J. Srba. Undecidability of domino games and hhp-bisimilarity. *Inform. and Comput.*, 184:343–368, 2003.

[9] S. Lasota. A polynomial-time algorithm for deciding true concurrency equivalences of basic parallel processes. In *MFCS'03*, volume 2747 of *LNCS*, pages 521–530. 2003.

[10] P. Madhusudan and P. S. Thiagarajan. Controllers for discrete event systems via morphisms. In *CONCUR'98*, volume 1466 of *LNCS*, pages 18–33. 1998.

[11] M. Mukund. Hereditary history preserving bisimulation is decidable for trace-labelled systems. In *FST TCS'02*, volume 2556 of *LNCS*, pages 289–300. 2002.

[12] W. Penczek. On undecidability of propositional temporal logics on trace systems. *Inform. Process. Lett.*, 43(3):147–153, 1992.

[13] W. Penczek and R. Kuiper. Traces and logic. In V. Diekert and G. Rozenberg, editors, *The Book of Traces*, pages 307–381. World Scientific, 1995.

[14] J. Srba. On the power of labels in transition systems. In *CONCUR'01*, volume 2154 of *LNCS*, pages 277–291. 2001.

[15] W. Vogler. Deciding history preserving bisimilarity. In *ICALP'91*, volume 510 of *LNCS*, pages 495–505. 1991.