High undecidability of weak bisimilarity for Petri nets

Petr Jančar *
Dept. of Computer Science, University of Ostrava
Dvořákova 7, 701 03 Ostrava, Czech Republic
e-mail: jancar@oudec.osu.cz

Abstract

It is shown that the problem whether two labelled place/transition Petri nets (with initial markings) are weakly bisimilar is highly undecidable – it resides at least at level ω of the hyperarithmetical hierarchy; on the other hand it belongs to Σ^1_1 (the first level of the analytical hierarchy). It contrasts with Π^0_1-completeness of the same problem for trace (language) equivalence. Relations to similar problems for the process algebra BPP (Basic Parallel Processes) are also discussed.

1 Introduction

In the theory of parallelism and concurrency, much effort is devoted to the decidability questions for various classes of (models of) processes and various notions of behavioural equivalences. For natural reasons, the question of algorithmic decidability is among the first ones to be asked when developing automatic verification techniques.

Although the main result here concerns Petri nets and high undecidability, an important source of motivation can be found in a decidability question for the process algebra BPP (Basic Parallel Processes, cf. e.g. [2]).

Recently, interesting results have been obtained for both BPP and Petri nets; the results concerned two of the central behavioural equivalences, namely the bisimulation equivalence (or bisimilarity) and the trace (or language) equivalence. In fact, BPP can be viewed as a subclass of Petri nets and the processes of obtaining the mentioned results for both models have been closely interconnected. A short survey is contained in the next paragraphs.

*Supported by the Grant Agency of Czech republic, Grant No. 201/93/2123

First we briefly recall a definition of BPP (the precise definition of Petri nets being given in Section 2).
The abstract syntax of a BPP-expression is

$$E ::= \mathbf{0} \mid X \mid a.E \mid E + E \mid E \| E$$

The operational semantics is standard – $\mathbf{0}$ is a nil-process, X is a process variable, $a.E$ performs the action a and then behaves like E, the operator $+$ means nondeterministic choice and $\|$ parallel merging (no communication between processes is allowed). A BPP-process is given by a family of equations $X_1 = E_1, \ldots, X_n = E_n$ where X_1 is the leading variable (the initial state).

It is easy to verify that, with respect to the "natural" corresponding transition systems, BPP can be viewed as a subclass of (labelled) Petri nets – every transition has exactly one input place in this case.

In [2] it is shown in an elegant way that (strong) bisimilarity is decidable for BPP.

Remark. In fact, the decidability was first established for a subclass – the normed processes – and later extended for all BPP. Recently, a polynomial algorithm for the normed BPP has been shown (the complexity for all BPP remaining open); cf. [9].
Although using different techniques, the results for BPP resemble very much those for BPA (Basic Process Algebra; cf. [3], [8]).

The decidability result for BPP still left unsolved the decidability question for the whole class of Petri nets which had been open for some time. Later the question was answered negatively in [12] – bisimilarity is undecidable for Petri nets. As for the trace equivalence, its undecidability for Petri nets has been known from [6]; in fact, [12] also provides a more direct proof of this undecidability. It is interesting that the trace equivalence is undecidable even for BPP; it is shown in [7] by a clever modification of the technique of [12].

Remark. In [11], another modification was added – to show undecidability for other equivalences in the linear time – branching time spectrum of [5] on BPP. On the other hand, decidability results for some finer versions of bisimilarity on BPP have been obtained in [14].

Allowing nonobservable, internal, actions (commonly denoted by τ), weak versions of the considered equivalences can be defined. For practice, these are more important, unfortunately the decidability is often affected; instead of finitely branching transition systems we can get infinitely branching ones and it surely makes things less feasible.

Of course, undecidability for a (strong) equivalence immediately implies undecidability for the relevant weak version. Therefore when being interested strictly in decidability/undecidability questions, the most important question emerging from the above discussion is that of decidability of weak bisimilarity for BPP.

It nicely demonstrates a difference between finite branching and infinite branching.
In finitely branching systems, the problem of non-bisimilarity (non-equivalence) of two states s, s' is usually easier than that of bisimilarity (equivalence). There is a natural systematic method exploring all possibilities of constructing a minimal bisimulation relation containing the pair (s, s'). The method resembles constructing a (finitely branching) tree in the breadth-first manner (any node corresponds to a finite set of pairs). If s,s' are non-bisimilar, it will be revealed eventually – any leaf of the so far constructed tree will correspond to a (finite) relation containing an (obviously) non-bisimilar pair. Thus we get, in fact, a finite witness of the non-bisimilarity. Therefore semi-decidability of the negative case is clear.
In the positive case, a similar finite witness of equivalence often does not exist. If it does exist, it is then much more demanding to show it; cf. [2] for the case of bisimilarity on BPP or [3] for BPA.
For the weak bisimilarity on BPP, [4] shows that a finite witness of the *positive* case exists and it is the *negative* case for which semi-decidability is not clear (leaving the decidability question open). In other words, the task to show the existence of a finite witness of weak non-bisimilarity for BPP-processes has not been accomplished yet.

Remark. One thing is clear: if the problem should be undecidable (i.e. the negative case being non-semidecidable), it can not be shown by reducing the halting problem to the negative case, which was, in principle, the method of showing the undecidability results mentioned above.
We should also mention that some decidability results for weak versions of some finer equivalences on a *subclass* of BPP are shown in [14].

It brings us back to a closer look at the whole class of Petri nets. In fact, we have indicated that the (strong) bisimulation equivalence problem is Π_1^0-complete – the non-equivalence (the negative case) being recursively equivalent to the halting problem (hence Σ_1^0-complete). Therefore it still could be the case that the weak non-bisimilarity is semi-decidable for the whole class of Petri nets, which would imply decidability for BPP. It is not unimaginable at first sight that non-bisimilarity could be demonstrated in some reasonable sense (by a finite witness). Unfortunately, we show here that it is not the case. It turns out that the weak non-bisimilarity on Petri nets is not only non-semidecidable but is beyond the arithmetical hierarchy (!). It might explain a bit why the semi-decidability of weak non-bisimilarity for BPP (which hopefully is the case) is not so easy to prove.

Although the result does not imply the desired decidability for BPP and has therefore no direct "practical" consequence, it has some value (as the author hopes) at least from the theoretical viewpoint. Having estimated the degree of undecidability for the weak version of bisimilarity for Petri nets, it is natural to ask a similar question for the (weak) trace equivalence. Using some well-known results concerning Petri nets, we show here that this problem remains

Π_1^0-complete. In this sense, the result highlights an aspect of the difference between the bisimilarity and the trace equivalence. On the other hand, in case of BPP the trace equivalence (in both versions) is Π_1^0-complete but it seems very probable that the weak bisimilarity is decidable (at least it is in Σ_1^0), which shows an interesting difference with the situation at Petri nets. It might illustrate the power of internal communication in some sense.

The main result is also completed by an upper bound, which is natural to be looked for when having a lower bound. The problem of weak bisimilarity for Petri nets is easily shown to be in Σ_1^1 (the first level of the analytical hierarchy); hence the negative formulation is in Π_1^1. Whether the problem belongs to the intersection Δ_1^1 (i.e. whether it is hyperarithmetical) is left open.

The paper is organized as follows. Section 2 gives basic definitions and claims the results. The relevant proofs are given in Section 3.

A preliminary version of the paper appeared as the report [13]. The report is self-contained since it includes an appendix describing a needed construction of [12]. Here we only refer to [12] omitting the appendix.

2 Definitions and results

$\mathcal{N}$ denotes the set of nonnegative integers, A^* the set of finite sequences of elements of A.

A labelled net is a tuple $\Sigma = (P, T, F, L)$ where P and T are finite disjoint sets of *places* and *transitions* respectively, $F : (P \times T) \cup (T \times P) \longrightarrow \{0, 1\}$ is a *flow function* (for $F(x, y) = 1$, there is an *arc* from x to y; we do not use multiple arcs in the paper) and $L : T \longrightarrow A$ is a *labelling* (attaches an action name – from a set A – to each transition). We can suppose a fixed (countable) A containing a special, *unobservable*, action τ, the other actions being *observable*.

By *a Petri net* we mean a tuple $N = (\Sigma, M_0)$, where Σ is a labelled net and M_0 the *initial marking*; a *marking* M being a function $M : P \longrightarrow \mathcal{N}$. (A marking gives the number of *tokens* for each place). A transition t is *enabled* at a marking M, $M \xrightarrow{t}_\Sigma$, if $M(p) \geq F(p, t)$ for every $p \in P$. An enabled transition t may *fire* at a marking M yielding marking M', $M \xrightarrow{t}_\Sigma M'$, where $M'(p) = M(p) - F(p, t) + F(t, p)$ for all $p \in P$.
In the natural way, the definitions can be extended for finite sequences of transitions $\sigma \in T^*$.

For any $a \in A$ (including τ), by $M \xrightarrow{a}_\Sigma M'$ we mean that $M \xrightarrow{t}_\Sigma M'$ for some t with $L(t) = a$; by $M \xRightarrow{a}_\Sigma M'$ we mean that $M \xrightarrow{\sigma}_\Sigma M'$ for some sequence σ of transitions, one of them being labelled by a, the others by τ (in case $a = \tau$, the sequence can be empty; i.e. $M \xRightarrow{\tau}_\Sigma M$ for all markings M). In the natural way, we define the relations $M \xRightarrow{w}_\Sigma$ and $M \xRightarrow{w}_\Sigma M'$ for finite sequences $w \in A^*$.

The *set of (observable) traces* (or the *language*) of a Petri net $N = (\Sigma, M_0)$ is defined as $\mathcal{T}(N) = \{w \in (A \setminus \{\tau\})^* \mid M_0 \stackrel{w}{\Longrightarrow}_\Sigma\}$. Two Petri nets N_1, N_2 are *(weakly) trace equivalent* if $\mathcal{T}(N_1) = \mathcal{T}(N_2)$.

Given two labelled nets $\Sigma_1 = (P_1, T_1, F_1, L_1)$, $\Sigma_2 = (P_2, T_2, F_2, L_2)$, a binary relation $R \subseteq \mathcal{N}^{P_1} \times \mathcal{N}^{P_2}$ is a *weak bisimulation* (cf. [16]) if the following conditions hold for all $(M_1, M_2) \in R$:
– for every a and M_1' s.t. $M_1 \stackrel{a}{\Longrightarrow}_{\Sigma_1} M_1'$, there is M_2' s.t. $M_2 \stackrel{a}{\Longrightarrow}_{\Sigma_2} M_2'$, where $(M_1', M_2') \in R$
and conversely
– for every a and M_2' s.t. $M_2 \stackrel{a}{\Longrightarrow}_{\Sigma_2} M_2'$, there is M_1' s.t. $M_1 \stackrel{a}{\Longrightarrow}_{\Sigma_1} M_1'$, where $(M_1', M_2') \in R$.

Two Petri nets N_1, N_2 are *weakly bisimilar* if there is a weak bisimulation relating their initial markings.

(If the relation $\stackrel{a}{\Longrightarrow}$ is replaced by $\stackrel{a}{\longrightarrow}$, we get the notion of strong bisimilarity.)

Let TE denote the set of (Gödel numbers, codes or notations of) all pairs of Petri nets (N_1, N_2) which are (weakly) trace equivalent. NTE will denote the set of all (weakly) trace non-equivalent pairs.
Similarly, let WB denote the set of all pairs of Petri nets (N_1, N_2) which are weakly bisimilar and NWB the set of all weakly non-bisimilar pairs.

The next theorems show the position of the defined sets in the (hyper)arithmetical and analytical hierarchies, the details of which can be found e.g. in [1].

Theorem 1.

NTE is Σ^0_1-complete (i.e. complete among the recursively enumerable sets); hence TE is Π^0_1-complete.

Theorem 2.

WB is in Σ^1_1 (the 1st level of the analytical hierarchy); hence NWB is in Π^1_1.

Theorem 3.

WB and NWB reside at least at the level ω in the hyperarithmetical hierarchy.

Theorem 1 is established using some well-known results regarding (un)decidability in Petri nets.
The upper bound of Theorem 2 is established by using directly the definition of weak bisimilarity.
The proof of the lower bound in Theorem 3 is more involved. It shows an algorithm which, given a 1st order formula ϕ of the standard arithmetic structure $(\mathcal{N}, 0, S, +, \cdot)$, constructs two Petri nets which are weakly bisimilar iff ϕ is true. This yields the result since the set of true 1st order sentences of arithmetics lies at level ω in the hyperarithmetical hierarchy (cf. e.g. [1], p. 560).

3 Proofs

We start with the proof concerning the trace equivalence.

Proof of Theorem 1.

It is well-known that the halting problem is Σ_1^0-complete. Since it is recursively reducible to NTE (via Hilbert's 10th problem as in [6] or directly as in [12]), it suffices to show that it is semi-decidable, given two Petri nets N_1, N_2, whether $\mathcal{T}(N_1) \neq \mathcal{T}(N_2)$.

The semi-decidability follows immediately from the decidability of the problem $\mathcal{P}$:

Instance: a Petri net N and an (observable) sequence w.

Question: Is $w \in \mathcal{T}(N)$?

We can then generate all finite (observable) traces and stop when reaching a sequence belonging to one of the sets $\mathcal{T}(N_1)$, $\mathcal{T}(N_2)$ and not belonging to the other.

The decidability of the problem $\mathcal{P}$ can be shown by various ways using some well-known results from Petri net theory. We outline one of these ways.

To the net N we can add a finite control unit whose states determine which transitions are allowed to fire next; it simplifies our considerations but, of course, it could be modelled by a modification directly in the net N (cf. e.g. [10]). In our case, for $w = a_1 a_2 \ldots a_n$, the control unit will have states $q_1, q_2, \ldots, q_{n+1}$, where q_1 is the initial one. Each q_i $(i = 1, 2, \ldots, n)$ only allows transitions labelled by τ or a_i; any τ-transition leaves the control unit in the (current) state q_i whereas any a_i-transition moves it into the (next) state q_{i+1}. Then it suffices to ask whether q_{n+1} is reachable, which is decidable (cf. [15]).

(In fact, we do not need the decidability of reachability. A technique using the Karp, Miller reachability tree, related to the coverability problem, would suffice; cf. e.g. [18] for definitions.)

$\square$

We continue with a short proof of the upper bound for WB.

Proof of Theorem 2.

By definition,

(the code of) a pair (Σ_1, M_1), (Σ_2, M_2) is in WB

iff

there is a set $X \subseteq \mathcal{N}^{r_1} \times \mathcal{N}^{r_2}$ (where r_i is the number of places in Σ_i) such that

$$(M_1, M_2) \in X \quad \bigwedge$$

$$\bigwedge \quad \forall (x,y) \in X \; [\; \forall a \forall x' \; (\; (x \stackrel{a}{\Longrightarrow}_{\Sigma_1} x') \longrightarrow \exists y' (y \stackrel{a}{\Longrightarrow}_{\Sigma_2} y' \wedge (x',y') \in X)\;)\;] \quad \bigwedge$$

$$\bigwedge \quad \forall (x,y) \in X \; [\; \forall a \forall y' \; (\; (y \stackrel{a}{\Longrightarrow}_{\Sigma_2} y') \longrightarrow \exists x' (x \stackrel{a}{\Longrightarrow}_{\Sigma_1} x' \wedge (x',y') \in X)\;)\;]$$

where $\forall a$ means for all a which occur in Σ_1 and/or Σ_2.

Because the relation $\stackrel{a}{\Longrightarrow}$ surely is arithmetical (it is even recursive due to the decidability of the reachability problem), it can be easily verified that there is a 2nd order formula $\phi(x, X)$ with one free individuum variable x, one free set variable X and with no set quantifiers such that for all (codes) $c \in \mathcal{N}$:

$$c \in WB \Longleftrightarrow \exists X \phi(c, X)$$

Thus the desired upper bound is established. $\square$

Now we prove Theorem 3.

The main idea lies in showing a certain (algorithmic) construction $\mathcal{CONS}$ which can be used for any Minsky counter machine C ([17]) with an even (for convenience) number of (nonnegative) counters $c_1, c_2, \ldots, c_{2n}$; the construction yields a pair of Petri nets N_1, N_2 such that

$$\begin{array}{c} N_1,\ N_2 \text{ are weakly bisimilar} \\ \text{iff} \\ \text{it is true that} \quad \exists x_1 \forall x_2 \ldots \exists x_{2n-1} \forall x_{2n} \neg ! C(x_1, x_2, \ldots, x_{2n}) \end{array}$$

By the notation $\neg ! C(x_1, x_2, \ldots, x_{2n})$ we mean that C does not halt for the input counter values $x_1, x_2, \ldots, x_{2n}$.

When we show the construction $\mathcal{CONS}$, the claim is obvious:

Proof of Theorem 3. (supposing the construction $\mathcal{CONS}$)

Given any 1st order formula ϕ of the standard arithmetic structure $(\mathcal{N}, 0, S, +, \cdot)$, we can transform it into an equivalent formula in the form $\exists x_1 \forall x_2 \ldots \exists x_{2n-1} \forall x_{2n} \phi'(x_1, x_2, \ldots, x_{2n})$ for some n, ϕ' being quantifier-free. (It can be accomplished by a standard transformation into the prenex normal form and by using dummy variables.)

Formula ϕ' is, in fact, a recursive predicate and it is a technical routine to construct a counter machine C with $2n$ counters such that for all $x_1, x_2, \ldots, x_{2n}$

$$\neg ! C(x_1, x_2, \ldots, x_{2n}) \quad \text{iff} \quad \phi'(x_1, x_2, \ldots, x_{2n}) \text{ is true}$$

(in fact, C could need some additional counter(s); in this case we could use more dummy variables and then use the respective counters as the additional ones).

For C, we can construct N_1, N_2 by our (promised) construction $\mathcal{CONS}$ and we have

$$\begin{array}{c} \phi \text{ is true in } (\mathcal{N}, 0, S, +, \cdot) \\ \text{iff} \\ \exists x_1 \forall x_2 \ldots \exists x_{2n-1} \forall x_{2n} \phi'(x_1, x_2, \ldots, x_{2n}) \text{ is true in } (\mathcal{N}, 0, S, +, \cdot) \\ \text{iff} \\ \exists x_1 \forall x_2 \ldots \exists x_{2n-1} \forall x_{2n} \neg ! C(x_1, x_2, \ldots, x_{2n}) \text{ is true} \\ \text{iff} \\ N_1,\ N_2 \text{ are weakly bisimilar} \end{array}$$

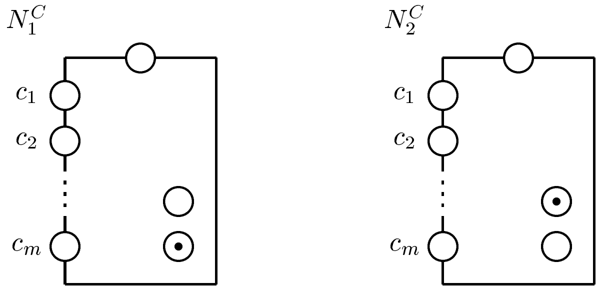

Figure 1:

by which the proof is finished. $\square$

Now we have to describe the construction $\mathcal{CONS}$.

As a basis, we use a result from [12]. It shows a construction which, given a counter machine C with m counters, yields two Petri nets N_1^C and N_2^C (cf. Fig.1). The nets have the same static structure; places $c_1, c_2, \ldots, c_m$ for counters, a "start" place (the upper one in Fig.1), two special places (drawn inside the box) and some other places (and transitions, of course). They only differ in the (predefined) initial marking which is zero except for the two special places (as shown in Fig.1).

Remark. By labelling the transitions in N_1^C (and hence also in N_2^C) no τ-labels are used; therefore weak and strong bisimilarity coincide in this case.

We do not describe details of the construction, which can be found in [12] or in the appendix of [13]. They are only necessary for checking the proof of the following (basic) proposition.

Proposition BP

There is an algorithm which, given any counter machine C, constructs two Petri nets N_1^C, N_2^C, each containing a designated start place and places $c_1, c_2, \ldots, c_m$ where m is the number of counters of C, such that:
for any $x_1, \ldots, x_m$ and $x'_1, \ldots, x'_m$, if we put a token in the start place of both N_1^C and N_2^C, $x_1, \ldots, x_m$ tokens in places $c_1, \ldots, c_m$ of N_1^C and $x'_1, \ldots, x'_m$ tokens in $c_1, \ldots, c_m$ of N_2^C respectively then the resulting nets are (weakly) bisimilar iff $x_i = x'_i$ for all i, $1 \le i \le m$, and $\neg !C(x_1, \ldots, x_m)$.

Proof.

If we considered the case with the same input counter values only ($x_i = x'_i$ for all i), the construction from [12] would be the desired algorithm, in fact.

When a counter machine C has different computations for any different inputs $x_1, \ldots, x_m$ and $x'_1, \ldots, x'_m$ then an easy inspection of the construction of [12] shows that N_1^C and N_2^C are nonbisimilar when starting with different counter values (Player 1 in the below described bisimulation game forces the situation, when the counter to be tested is zero in one net and nonzero in the other; i.e. in one net a transition, with its unique label, is enabled while in the other net not). Therefore for such a machine C, the construction of [12] is sufficient again.

To prove the proposition completely, it suffices to realize that for any C we can construct the following C' with an additional auxiliary counter: C' starts with "moving" the value of each counter, successively, into the auxiliary counter and back and then computes like C. Therefore C' really has different computations for any different inputs (and is equivalent to C w.r.t. halting). Hence the nets $N_1^{C'}$ and $N_2^{C'}$ in the "original" sense of [12], with the place for the auxiliary counter viewed as "internal", can be taken as N_1^C, N_2^C in our proposition. □

For describing and verifying $\mathcal{CONS}$, some game terminology will be useful. First we recall a natural definition of a bisimulation game (cf. e.g. [19]).

(Weak) Bisimulation Game

1. *Prerequisities*
   There are two players, Player 1 and Player 2, and a pair of Petri nets N_1, N_2 (as the "playboard").

2. *Rules*
   Player 1 chooses one of the nets and changes the marking (by firing a sequence of transitions) according to the relation $\stackrel{a}{\Longrightarrow}$ for some a.
   Then Player 2 replies by changing the marking in *the other* net according to $\stackrel{a}{\Longrightarrow}$ for the same a (if it is possible).
   Again, Player 1 chooses one of the nets ..., Player 2 replies ... etc.

3. *Result*
   The player who has no possible move (being his turn) loses; the other player wins. The case of an infinite run of the game is considered as successful for Player 2 (he defends successfully).

It is not difficult to see that Player 1 has a winning strategy if and only if N_1, N_2 are not weakly bisimilar.

In the rest of this section, we suppose a fixed counter machine C with $2n$ counters $c_1, c_2, \ldots, c_{2n}$. Let us now consider another game.

Game G (for Player 1 and Player 2)

a/ *Board:* Petri nets N_1^C and N_2^C (cf. Fig.1)

b/ *Rules:*

1. First, there are $2n$ special moves:
   – Player 2 begins and puts an arbitrary number of tokens x_1 in the

places c_1 in both nets (the same value in both nets),
– Player 1 continues and puts an arbitrary number of tokens x_2 in the places c_2,
– Player 2 continues setting the value of c_3,
...
– Player 1 sets the value of c_{2n}.

2. After these $2n$ moves, a token is put in the start place of N_1^C as well as in the start place of N_2^C and the bisimulation game starts (Player 1 chooses ...).

Proposition BP ensures that Player 1 has no winning strategy in the game G (in other words: Player 2 has a defending strategy) iff it is true that $\exists x_1 \forall x_2 \ldots \exists x_{2n-1} \forall x_{2n} \neg ! C(x_1, x_2, \ldots, x_{2n})$.

In fact, $\mathcal{CONS}$ will yield Petri nets N_1, N_2 such that the game G on N_1^C, N_2^C can be simulated by playing the bisimulation game on N_1, N_2. These Petri nets will be based on (more copies of) N_1^C, N_2^C. Any concrete run of the game G on N_1^C, N_2^C will correspond to a run of the bisimulation game on N_1, N_2. There will be runs on N_1, N_2 which do not correspond to runs on N_1^C, N_2^C but it will be shown that neither of the Players can get any advantage by causing such runs. In this sense, the "external" rules of G for the first $2n$ moves on N_1^C, N_2^C are inherently present in the bisimulation game on N_1, N_2. Let us remark that setting a counter value will be accomplished by τ-transitions, which is their "only" use.

Now we show an inductive construction, for $j = 2n, 2n-1, \ldots, 1, 0$, of Petri nets N_j^L, N_j^R (L for "left", R for "right"). N_0^L, N_0^R will be the desired N_1, N_2 (the output of $\mathcal{CONS}$). Intuitively, (starting) the pair N_j^L, N_j^R corresponds to the situation in the game G where j counters have been set.

We illustrate the construction in figures where we use broken arrows leading from a place to a (sub)net; the meaning is explained in Fig.2. Such an arrow leading from a place r to a (sub)net Σ and being labelled by a subset of places of Σ denotes the fact that there are special τ-labelled transitions; r is a "run-place" for these transitions and firing a sequence of them allows to add (or to subtract) an arbitrary – but the same for all – number to (or from) all places in the subset. Fig.2 illustrates it for a 2-element subset.

N_j^L and N_j^R will contain more "copies" of (places corresponding to) counters. By labelling a broken arrow by c_{2k+1}, for example, we understand that the label c_{2k+1} stands for the set of all copies of the counter c_{2k+1} in the target subnet.

The (inductive) construction of N_j^L and N_j^R

1. Let N_{2n}^L be N_1^C and let N_{2n}^R be N_2^C.
2. For an odd $j = 2k+1$, N_{2k+1}^L, N_{2k+1}^R are shown in Fig.3. (N_{2k+1}^L contains N_{2k+1}^R as a subnet).

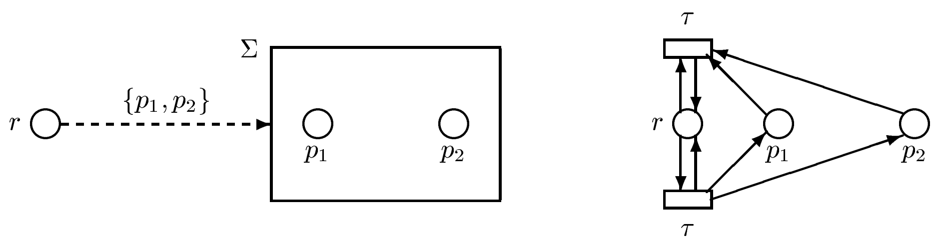

Figure 2: Meaning of broken arrows

3. For an even $j = 2k$, N^L_{2k}, N^R_{2k} are shown in Fig.4. (N^R_{2k} contains N^L_{2k} as a subnet).

The action a in the figures is observable (not τ). By the "start place" of N^L_j (or N^R_j) we mean the upper place in the figure.

Consider now the bisimulation game on the nets N^L_0 and N^R_0 with a token in the start place of both – and zero elsewhere with the exception of the special (internal) places of copies of N^C_1, N^C_2 (recall Fig.1), which are subnets of the nets being considered.

Notice that, within any move, none of the Players has any reason for firing a τ-sequence only or for adding some τ-sequence after firing an a – the results with and without such a τ-sequence are obviously weakly bisimilar (the effect of such a sequence can be achieved or "undone" freely by a τ-sequence starting the next move). Therefore we can suppose the beginning of the game as follows: Player 1 fires some τ-sequence and an a-transition in one net, Player 2 replies with firing a τ-sequence and an a-transition in the other net etc. – after $2n$ such moves of Player 1 (and $2n$ answers of Player 2), two from the copies of N^C_1, N^C_2 are "started" (with some input values determined by previous τ-sequences).

Now it suffices to prove the following propositions $\mathcal{P}_j$, $j = 2n, 2n-1, \ldots, 1, 0$. $\mathcal{P}_0$ shows that N^L_0, N^R_0 are really our "desired" N_1, N_2 (the output of $\mathcal{CONS}$ relevant to our fixed counter machine C). We again deal with the cases "j even" and "j odd" separately.

Proposition $\mathcal{P}_{2k}$

For any $x_1, x_2, \ldots, x_{2k}$ and $x'_1, x'_2, \ldots, x'_{2k}$, when we put a token in the start place of both N^L_{2k} and N^R_{2k}, $x_1, x_2, \ldots, x_{2k}$ tokens in all copies of $c_1, c_2, \ldots, c_{2k}$ in N^L_{2k} and $x'_1, x'_2, \ldots, x'_{2k}$ tokens in all copies of $c_1, c_2, \ldots, c_{2k}$ in N^R_{2k} respectively then the resulting nets are weakly bisimilar iff $x_i = x'_i$ for all $i = 1, 2, \ldots, 2k$ and $\exists x_{2k+1} \forall x_{2k+2} \ldots \exists x_{2n-1} \forall x_{2n} \neg ! C(x_1, x_2, \ldots, x_{2n})$.

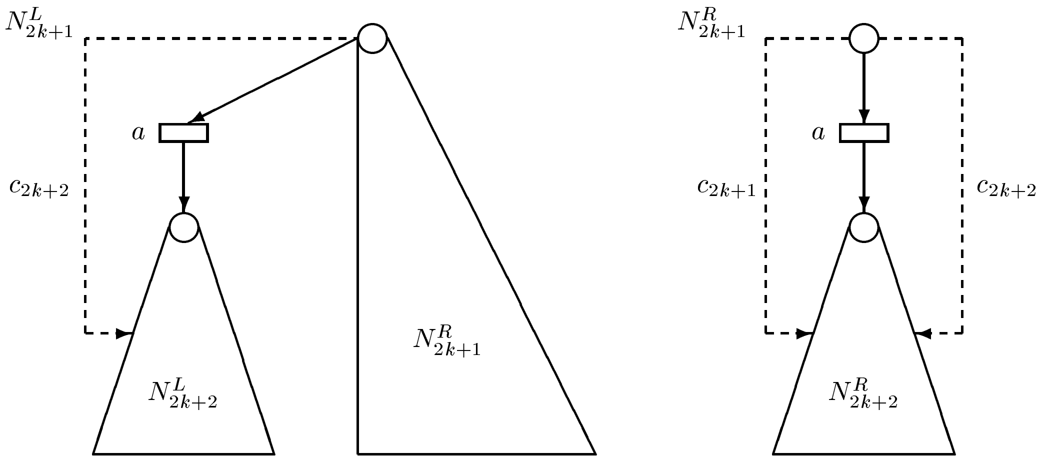

Figure 3: N^L_j and N^R_j for ODD $j = 2k + 1$

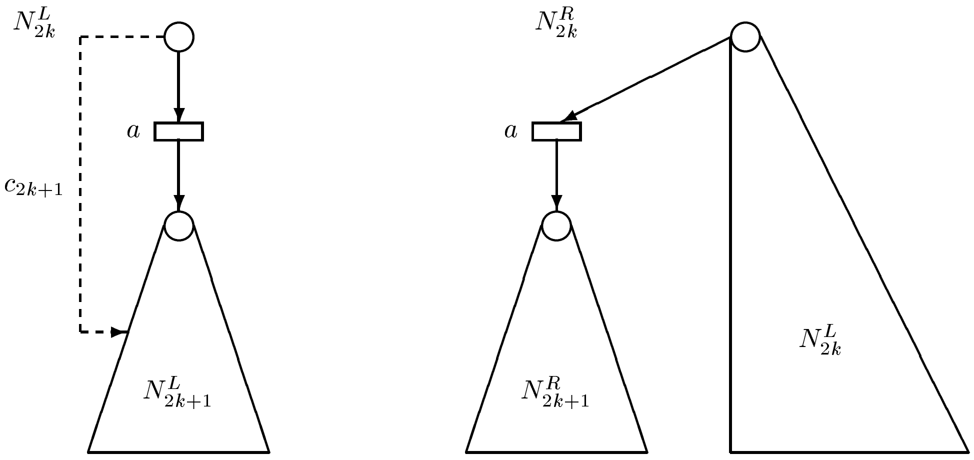

Figure 4: N^L_j and N^R_j for EVEN $j = 2k$

Proposition $\mathcal{P}_{2k+1}$

For any $x_1, x_2, \ldots, x_{2k}$*, any* $x'_1, x'_2, \ldots, x'_{2k}$ *and any* x_{2k+1}*, when we put a token in the start place of both* N^L_{2k+1} *and* N^R_{2k+1}*,* $x_1, x_2, \ldots, x_{2k}$ *tokens in all copies of* $c_1, c_2, \ldots, c_{2k}$ *in* N^L_{2k+1}*,* $x'_1, x'_2, \ldots, x'_{2k}$ *tokens in all copies of* $c_1, c_2, \ldots, c_{2k}$ *in* N^R_{2k+1} *respectively and* x_{2k+1} *tokens in all copies of* c_{2k+1} *in* N^L_{2k+1} *then the resulting nets are weakly bisimilar iff* $x_i = x'_i$ *for all* $i = 1, 2, \ldots, 2k$ *and* $\forall x_{2k+2} \exists x_{2k+3} \forall x_{2k+4} \ldots \exists x_{2n-1} \forall x_{2n} \neg !C(x_1, x_2, \ldots, x_{2n})$.

First note that the base case, Proposition $\mathcal{P}_{2n}$, follows immediately from Proposition BP. (Recall that N^L_{2n} is N^C_1 and N^R_{2n} is N^C_2.) Therefore to prove $\mathcal{P}_{2k}$ ($\mathcal{P}_{2k+1}$), we can suppose that $\mathcal{P}_{2k+1}$ ($\mathcal{P}_{2k+2}$) holds. We start with Proposition $\mathcal{P}_{2k}$ since it is a bit easier.

Proof of Proposition $\mathcal{P}_{2k}$

Consider N^L_{2k} and N^R_{2k}, $k < n$, (cf. Fig.4) with a token in the start place of both and $x_1, x_2, \ldots, x_{2k}$ tokens in all copies of $c_1, c_2, \ldots, c_{2k}$ in N^L_{2k} and $x'_1, x'_2, \ldots, x'_{2k}$ tokens in all copies of $c_1, c_2, \ldots, c_{2k}$ in N^R_{2k} respectively.

If $x_i \neq x'_i$ for some $i, 1 \leq i \leq 2k$, then Player 1 can "start" N^R_{2k+1} firing the "left" a-transition in N^R_{2k} after which Player 2 is forced to start N^L_{2k+1} – but not having the possibility to change the inequality $x_i \neq x'_i$, he reaches a "situation" which is non-bisimilar due to Proposition $\mathcal{P}_{2k+1}$.

Now suppose $x_i = x'_i$ for all $i, 1 \leq i \leq 2k$. If Player 1 wants to win, he has to choose the left a in N^R_{2k} – by choosing another a, Player 2 could obviously reach the identical situation in both nets (the same "active" subnets). Recall that it has no sense for Player 1 to add a τ-sequence and note that values of $c_1, c_2, \ldots, c_{2k}$ remain unchanged in (now active) N^R_{2k+1}. Player 2 answers by firing a in N^L_{2k} preceeded by setting an arbitrary value x_{2k+1} in (all copies of) c_{2k+1} in N^L_{2k+1} (which now becomes active). If there is some x_{2k+1} such that $\forall x_{2k+2} \exists x_{2k+3} \forall x_{2k+4} \ldots \exists x_{2n-1} \forall x_{2n} \neg !C(x_1, x_2, \ldots, x_{2n})$ then Player 2 could choose such x_{2k+1} and reach a bisimilar situation; otherwise, no matter what he sets in c_{2k+1}, the resulting situation is non-bisimilar (both variants follow from Proposition $\mathcal{P}_{2k+1}$). $\square$

Proof of Proposition $\mathcal{P}_{2k+1}$

Consider N^L_{2k+1} and N^R_{2k+1}, (cf. Fig.3) with a token in the start place of both, $x_1, x_2, \ldots, x_{2k}$ tokens in all copies of $c_1, c_2, \ldots, c_{2k}$ in N^L_{2k+1}, $x'_1, x'_2, \ldots, x'_{2k}$ tokens in all copies of $c_1, c_2, \ldots, c_{2k}$ in N^R_{2k+1} respectively and x_{2k+1} tokens in all copies of c_{2k+1} in N^L_{2k+1}.

The case with $x_i \neq x'_i$ for some $i, 1 \leq i \leq 2k$ is similar to that in the proof for $\mathcal{P}_{2k}$.

Now suppose $x_i = x'_i$ for all $i, 1 \leq i \leq 2k$. It is clear that if Player 1 wants to win, he has to choose the left a – now in N^L_{2k+1}, thus starting N^L_{2k+2}. But before this starting he can set any value x_{2k+2} in (all copies of) c_{2k+2} in N^L_{2k+2}. Player

2 answers in N^R_{2k+1} by setting the same values of (all copies of) c_{2k+1} and c_{2k+2} in N^R_{2k+2} as are set in N^L_{2k+2} (in fact, Player 2 sets the same value of c_{2k+1} as he chose in the "left" net in the previous move and sets the same value of c_{2k+2} as Player 1 has just chosen); note that if he had not set the same values, he would have reached a nonbisimilar situation due to Proposition $\mathcal{P}_{2k+2}$. If now for all x_{2k+2} it holds that $\exists x_{2k+3} \forall x_{2k+4} \ldots \exists x_{2n-1} \forall x_{2n} \neg ! C(x_1, x_2, \ldots, x_{2n})$ we have surely arrived at a bisimilar situation; otherwise Player 1 could choose some x_{2k+2} which ensures arriving at a nonbisimilar situation (again due to induction hypothesis $\mathcal{P}_{2k+2}$). $\square$

Thus we have finished the description and the correctness proof for the construction $\mathcal{CONS}$ and we are done.

Acknowledgements

I would like to thank J.Bradfield and J.Esparza for discussions and comments on the first version of this paper.

References

[1] J. Barwise (ed.), *Handbook of Mathematical Logic* (North Holland, 5th printing 1989).

[2] S. Christensen, Y. Hirshfeld and F. Moller, Bisimulation equivalence is decidable for all Basic Parallel Processes, in: *Proc. CONCUR'93*, Lecture Notes in Computer Science, Vol. 715 (Springer, Berlin, 1993) 143–157.

[3] S. Christensen, H. Hüttel and C. Stirling, Bisimulation equivalence is decidable for all context-free processes, in: *Proc. CONCUR'92*, Lecture Notes in Computer Science, Vol. 630 (Springer, Berlin, 1992) 138–147.

[4] J. Esparza, On the uniform word problem for commutative context-free grammars, unpublished.

[5] R.J. van Glabbeek, The linear time – branching time spectrum, in: *Proc. CONCUR'90*, Lecture Notes in Computer Science, Vol. 458 (Springer, Berlin, 1990) 278–297.

[6] M. Hack, Decision problems for Petri nets and vector addition systems, MAC Tech. Memo 53, MIT, 1975.

[7] Y. Hirshfeld, Petri nets and the equivalence problem, in: *Proc. Computer Science Logic '93*, Lecture Notes in Computer Science, Vol. 832 (Springer-Verlag, 1994) 165–174.

[8] Y. Hirshfeld, M. Jerrum and F. Moller, A polynomial algorithm for deciding bisimilarity of normed context-free processes, Tech.rep. ECS-LFCS-94-286, Dept. of comp. sci., Univ. of Edinburgh, UK, March 1994 (to appear in *Theor. Comp. Sci.*).

[9] Y. Hirshfeld, M. Jerrum and F. Moller, A polynomial-time algorithm for deciding bisimulation equivalence of normed Basic Parallel Processes, Tech.rep. ECS-LFCS-94-288, Dept. of comp. sci., Univ. of Edinburgh, UK, April 1994.

[10] J. Hopcroft and J. Pansiot, On the reachability problem for 5-dimensional vector addition systems, *Theoret. Comput. Sci.* **8** (1979) 135–159.

[11] H. Hüttel, Undecidable equivalences for Basic Parallel processes, 13th Conf. on Foundations of Software Technology and Theoretical Computer Science, Bombay, India, 1993
(also Tech.rep. ECS-LFCS-93-276, Dept. of comp. sci., Univ. of Edinburgh, UK, August 1993).

[12] P. Jančar, Decidability questions for bisimilarity of Petri nets and some related problems, in *Proc. STACS'94*, Lecture Notes in Computer Science, Vol. 775 (Springer, Berlin, 1994) 581–592
(an extended version should appear in *Theor. Comp. Sci.*)

[13] P. Jančar, High undecidability of weak bisimilarity for Petri nets, Tech.rep. ECS-LFCS-94-298, Dept. of comp. sci., Univ. of Edinburgh, UK, August 1994.

[14] A. Kiehn and M. Hennessy, On the decidability of non-interleaving process equivalences, in *Proc. CONCUR'94*, Lecture Notes in Computer Science, Vol. 836 (Springer-Verlag, 1994).

[15] E. Mayr, An algorithm for the general Petri net reachability problem, *SIAM J. Comput.* **13** (1984) 441–460.

[16] R. Milner, *Communication and Concurrency* (Prentice Hall, 1989).

[17] M. Minsky, *Computation: Finite and Infinite Machines* (Prentice Hall, 1967).

[18] J.L. Peterson, *Petri Net Theory and the Modelling of Systems* (Prentice Hall, 1981).

[19] C. Stirling, Modal and temporal logics for processes, Lecture notes to the Summerschool in logical methods in concurrency, Aarhus University, Denmark, 1993.