Decidability Questions for Bisimilarity of Petri Nets and Some Related Problems*

Petr Jančar
Dept. of Computer Science, University of Ostrava
Dvořákova 7, 701 00 Ostrava 1, Czech Republic
e-mail: jancar@oudec.osu.cz

Abstract

The main result is undecidability of bisimilarity for labelled (place / transition) Petri nets. The same technique applies to the (prefix) language equivalence and reachability set equality, which yields stronger versions with simpler proofs of already known results. The paper also mentions decidability of bisimilarity if one of the nets is deterministic up to bisimilarity. Another decidability result concerns semilinear bisimulations and extends the result of [CHM93] for Basic Parallel Processes (BPP).

1 Introduction

The relation of bisimulation plays an important role in the theory of parallelism and concurrency (cf. e.g.[M89]). An interesting question concerns decidability of bisimilarity for various classes of (models of) processes (see e.g. [CHS92],[CHM93] for recent results). In fact, BPP of [CHM93] are a special subclass of Petri nets. For the general (place/transition labelled) Petri nets, the problem was mentioned as open e.g. in [ABS91].

Using the halting problem for Minsky counter machines, this paper shows undecidability of the problem even if restricted to labelled Petri nets with a fixed static structure and 2 unbounded places.

The proof also shows undecidability of (prefix) language equivalence for the mentioned Petri nets with 2 unbounded places. This problem for (unrestricted) Petri nets is known to be undecidable due to Hack ([H75]); Valk and Vidal-Naquet ([VV81]) showed that nets with 4 and 5 unbounded places are sufficient for the undecidability.

*Partly supported by the Grant Agency of Czech Republic, Grant No. 201/93/2123

A similar technique applies also to equality of reachability sets, which yields undecidability of the problem even if restricted to Petri nets with 5 unbounded places. The known proofs in [B73], [H76] (see also [P81]) use Hilbert's 10th problem and Petri nets weakly computing polynomials; they do not put any bound on the number of unbounded places.

In this sense, the technique of the proof for bisimilarity also yields stronger versions of some known results. In addition, it shows them in a significantly simpler way.

This paper also contains some decidability results. The decidability of bisimilarity for one-to-one labelled (or "unlabelled") Petri nets is clear due to reducibility of (prefix) language equivalence of these nets to the reachability problem (cf. [H75], [M84]).
We mention here another reduction, the details of which are given in [J93], allowing an easy generalization for the nets which are deterministic up to bisimilarity.

Another subclass of labelled Petri nets for which the decidability of bisimilarity has been known is the above mentioned BPP of [CHM93] (isomorphic to Petri nets where each transition has one input place only). The proof employs a technique (suggested by Y.Hirshfeld) which is, in fact, more general – it implies decidability for the subclass where the bisimulation equivalence is a congruence w.r.t. (nonnegative vector) addition.
Here the result is further extended: we show that the existence of a semilinear bisimulation is sufficient for the decidability. It is completed by the fact, known from [ES69], that any congruence is semilinear.

Section 2 contains basic definitions, Section 3 the undecidability results, Section 4 the decidability results. Section 5 contains additional remarks (e.g. the relation to vector addition systems) and some hints for further work.

The paper is based on the report [J93].

2 Definitions

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

A (labelled) static net is a tuple (P, T, F), (P, T, F, L) respectively, where P and T are finite disjoint sets of *places* and *transitions* respectively, $F : (P \times T) \cup (T \times P) \longrightarrow \mathcal{N}$ is a *flow function* (for $F(x, y) > 0$, there is an *arc* from x to y with *multiplicity* $F(x, y)$) and $L : T \longrightarrow A$ is a *labelling* (attaches an action name – from a set A – to each transition). By L we also denote the homomorphic extension $L : T^* \longrightarrow A^*$.

A (labelled) Petri net is a tuple $N = (S, M_0)$, where S is a (labelled) static net and M_0 is an *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, $M \xrightarrow{t}$, 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', $M \xrightarrow{t} M'$, 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 sequences of transitions $\sigma \in T^*$.

The *reachability set* of a Petri net N is defined as
$\mathcal{R}(N) = \{M \mid M_0 \xrightarrow{\sigma} M \ for\ some\ \sigma \in T^*\}$.
A *place* $p \in P$ is *unbounded* if for any $k \in \mathcal{N}$ there is $M \in \mathcal{R}(N)$ s.t. $M(p) > k$.

The *(prefix) language* of a labelled Petri net N is defined as
$\mathcal{L}(N) = \{w \in A^* \mid M_0 \xrightarrow{\sigma} \ for\ some\ \sigma\ with\ L(\sigma) = w\}$.

Given two labelled static nets (P_1, T_1, F_1, L_1), (P_2, T_2, F_2, L_2) , a binary relation $R \subseteq \mathcal{N}^{P_1} \times \mathcal{N}^{P_2}$ is a *bisimulation* if for all $(M_1, M_2) \in R$:
- for each $t_1 \in T_1$, $M_1 \xrightarrow{t_1} M_1'$, there is $t_2 \in T_2$ s.t. $L_1(t_1) = L_2(t_2)$ and $M_2 \xrightarrow{t_2} M_2'$, where $(M_1', M_2') \in R$
and conversely
- for each $t_2 \in T_2$, $M_2 \xrightarrow{t_2} M_2'$, there is $t_1 \in T_1$ s.t. $L_1(t_1) = L_2(t_2)$ and $M_1 \xrightarrow{t_1} M_1'$, where $(M_1', M_2') \in R$.

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

Notice that $\mathcal{L}(N_1) = \mathcal{L}(N_2)$ for bisimilar nets N_1, N_2.

3 Undecidability Results

A counter machine C with nonnegative counters $c_1, c_2, ..., c_m$ is a program

$$1 : COMM_1;\ 2 : COMM_2;\ ......\ ;\ n : COMM_n$$

where $COMM_n$ is a $HALT$-command and $COMM_i$ $(i = 1, 2, ..., n-1)$ are commands of the following two types

1/ $c_j := c_j + 1;\ goto\ k$

2/ $if\ c_j = 0\ then\ goto\ k_1\ else\ (c_j := c_j - 1;\ goto\ k_2)$

assuming $1 \leq k, k_1, k_2 \leq n$, $1 \leq j \leq m$.

The set BS of *branching states* is defined as $BS = \{i \mid COMM_i$ is of the type 2$\}$.

It is well-known (cf. [M67]) that there is a fixed ("universal") counter machine C with two counters c_1, c_2 such that it is undecidable for given input values x_1, x_2 of c_1, c_2 whether C halts or not.

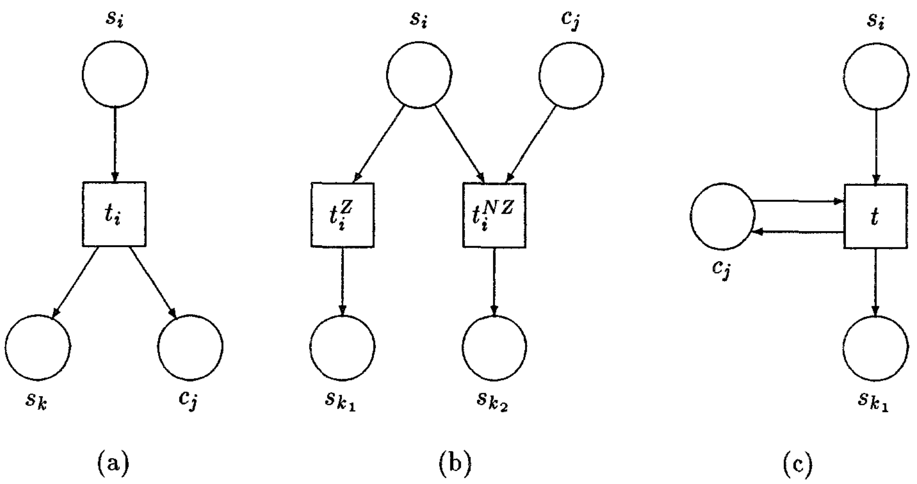

Figure 1:

Consider a counter machine C, with input values $x_1, x_2, ..., x_m$, in the above notation. We describe a construction of the *basic net* N_C which simulates C in a weak sense. By adding (x, y) we mean increasing $F(x, y)$ by 1 (mostly it means from 0 to 1 – adding one arc), unless otherwise stated. If $F(x, y)$ is not mentioned explicitly, it is equal to 0.

Construction of N_C

1. Let $c_1, c_2, ..., c_m$ (the counter part) and $s_1, s_2, ..., s_n$ (the state part) be places of N_C .

2. For $i = 1, 2, ..., n-1$ add new transitions and arcs depending on the type of $COMM_i$:

Case 1: $COMM_i$ is $c_j := c_j + 1$; *goto* k :
Add t_i with (s_i, t_i), (t_i, c_j), (t_i, s_k) (cf. Fig.1(a))

Case 2: $COMM_i$ is *if* $c_j = 0$ *then goto* k_1 *else* ($c_j := c_j - 1$; *goto* k_2) :
add t_i^Z (Z for zero) with (s_i, t_i^Z), (t_i^Z, s_{k_1}), and
t_i^{NZ} (NZ for non-zero) with (s_i, t_i^{NZ}), (c_j, t_i^{NZ}), (t_i^{NZ}, s_{k_2}) (cf. Fig.1(b)).

3. The initial marking will consist of the input values $x_1, x_2, ..., x_m$ in places $c_1, c_2, ..., c_m$, 1 token in s_1, 0 in the other places, which completes the construction.

N_C can simulate C in a natural way but (only) transitions t_i^Z can "*cheat*", i.e. fire although the relevant c_j is not 0.

Adding a *dc-transition* (*dc* for "definitely cheating") to N_C for some $i \in BS$ means adding a new transition t with (s_i, t), (c_j, t), (t, c_j), (t, s_{k_1}), j, k_1 taken from $COMM_i$ (cf Fig.1(c)).
Notice that such t has the same effect as t_i^Z but firing it always means cheating.

Now we establish the main theorems.

Theorem 3.1. *Bisimilarity as well as language equivalence are undecidable for labelled Petri nets, even if restricted to nets with a fixed static structure and 2 unbounded places.*

Proof. Let C be a (fixed) universal counter machine, with input values x_1, x_2, and N_C the basic net in the notation as above. Let us construct nets N_1, N_2 as follows.

Construction of N_1, N_2

1. To N_C, add new places p, p' and a new transition x with arcs (s_n, x), (p, x).
2. Take any any one-to-one labelling L of transitions.
3. For each $i \in BS$, add two dc–transitions t_i', t_i'' (with the relevant arcs) and additional arcs (p, t_i'), (t_i', p'), (p', t_i''), (t_i'', p) and put $L(t_i') = L(t_i'') = L(t_i^Z)$ (cf Fig.2).
4. Now take two copies of the arised net.
   In one copy put 1 token in p and 0 in p' (elsewhere the initial marking coincides with that of N_C); the resulting marking will be denoted by M_1, the whole net by N_1.
   In the other copy put 1 token in p' and 0 in p; the resulting marking will be denoted by M_2, the whole net by N_2.

Notice that only c_1, c_2 are (possibly) unbounded.

Now we show that the following conditions are equivalent

a) C does not halt (for the given inputs x_1, x_2)

b) N_1, N_2 are bisimilar

c) $\mathcal{L}(N_1) \subseteq \mathcal{L}(N_2)$

d) $\mathcal{L}(N_1) = \mathcal{L}(N_2)$

which proves the theorem. Thus we also directly show the undecidability of the language containment problem, although it follows from the undecidability of the language equivalence problem.

If C halts (for input x_1, x_2): $L(\sigma)$ where σ is the correct (non-cheating) sequence ended by x belongs to $\mathcal{L}(N_1)$ and not to $\mathcal{L}(N_2)$. (Firing σ in N_2 we

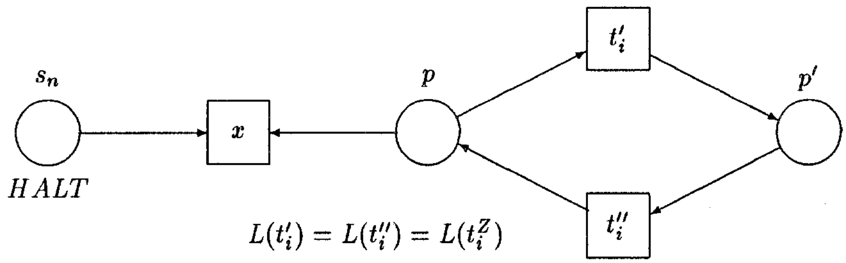

Figure 2:

have no possibility to fire a dc–transition; hence we can not move the token from p' to p and x will remain disabled.)
Hence $\mathcal{L}(N_1) \not\subseteq \mathcal{L}(N_2)$ which implies that $\mathcal{L}(N_1) \neq \mathcal{L}(N_2)$ and that N_1, N_2 are not bisimilar.

If C does not halt: Consider the set $\mathcal{M}$ of all couples (M', M'') where M', M'' are reachable without cheating from M_1, M_2 respectively and $M'(p) = 1, M'(p') = 0,\ M''(p) = 0, M''(p') = 1$.
$\mathcal{D}$ will denote the diagonal, the set of all couples (M, M).

We show that the union $\mathcal{D} \cup \mathcal{M}$ is a bisimulation containing (M_1, M_2) (notice that $(M_1, M_2) \in \mathcal{M}$).

As the static nets underlying N_1 and N_2 are the same, the condition from the definition of bisimulation is clear for any couple $(M, M) \in \mathcal{D}$.
As regards a couple $(M', M'') \in \mathcal{M}$:
– for any noncheating firing in M' (M'') there is the same noncheating firing in M'' (M') yielding again a couple of markings from $\mathcal{M}$,
– for any cheating firing of t^Z_i or t'_i in M', firing of t''_i or t^Z_i respectively is possible in M'' resulting in a couple $(M, M) \in \mathcal{D}$. Similarly for t^Z_i, t''_i in M'' and t'_i, t^Z_i in M'.

Hence N_1, N_2 are bisimilar, which implies $\mathcal{L}(N_1) = \mathcal{L}(N_2)$ and $\mathcal{L}(N_1) \subseteq \mathcal{L}(N_2)$. □

Remark. Considering only language equivalence, we could use a simpler, "nonsymmetric", construction: N_1 without p' and dc–transitions, N_2 with only one set of dc–transitions moving the token from p' to p. Recently Hirshfeld [Hi93] modified the construction showing undecidability of language equivalence even for labelled Petri nets equivalent to BPP (each transition has one input place only).

Theorem 3.2. *The containment and the equality problems for reachability sets of Petri nets are undecidable, even if restricted to nets with one of two fixed static structures and 5 unbounded places.*

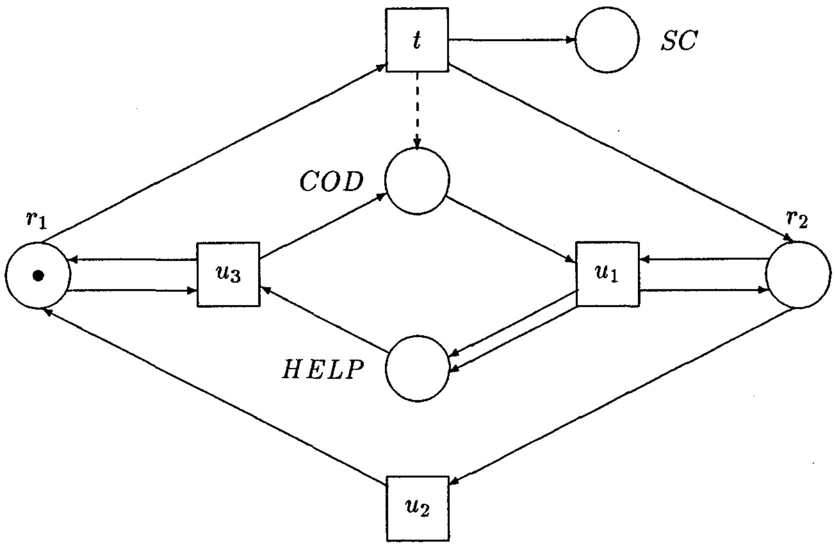

Figure 3:

Proof. Let C and N_C be as in Proof of Theorem 3.1. Let us perform the following construction of a net N.

Construction of N

1. Take N_C and add a dc–transition t'_i for each $i \in BS$.
2. Add places COD,$HELP$,SC (step counter) and r_1,r_2; put 1 token in r_1, 0 in the others.
3. Add arcs (r_1, t),(t, r_2),(t, SC) for each (so far constructed) transition t and (t_i^{NZ}, COD) for each t_i^{NZ}.
4. Add transitions u_1, u_2, u_3 and arcs (COD, u_1), (r_2, u_1), (u_1, r_2), $(u_1, HELP)$ with $F(u_1, HELP) = 2$, (r_2, u_2), (u_2, r_1), $(HELP, u_3)$, (r_1, u_3), (u_3, r_1), (u_3, COD) (cf. Fig.3).
5. The arised net is denoted by N.

Hence each "*non-u_i*" transition in N "moves" the token from r_1 to r_2 and adds a token to SC; each transition t_i^{NZ}, in addition, adds a token to COD. Before next firing of a *non-u_i* transition, a sequence from $u_1^* u_2 u_3^*$ is performed (possibly)

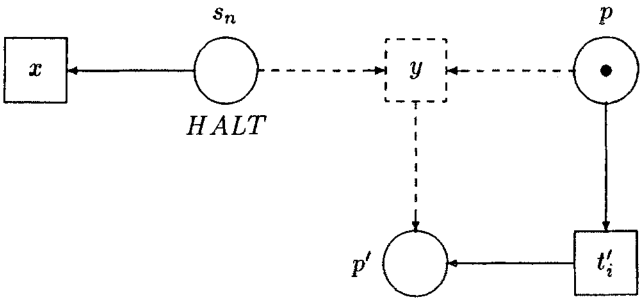

Figure 4:

changing COD. Notice that the maximal change of COD (with $HELP$ empty) can be $COD := 2.COD$ or $COD := 2(COD + 1)$ (for t_i^{NZ}).

Using N, let us construct N_1,N_2 as follows.

Construction of N_1, N_2

1. Take N and add a place p with 1 token and a place p' with 0 tokens.
2. Add a transition x with the arc (s_n, x) and the arcs (p, t'_i), (t'_i, p') for each $i \in BS$.
3. The arised net will be denoted by N_2. N_1 arises from N_2 by adding a transition y and the arcs (s_n, y), (p, y), (y, p') (cf. Fig.4).

Trivially $\mathcal{R}(N_2) \subseteq \mathcal{R}(N_1)$. Also notice that only places c_1,c_2,COD,$HELP$,SC are (possibly) unbounded.

Now we show that the following conditions are equivalent

a) C does not halt (for the given inputs x_1, x_2)

b) $\mathcal{R}(N_1) \subseteq \mathcal{R}(N_2)$

c) $\mathcal{R}(N_1) = \mathcal{R}(N_2)$

which proves the theorem.

If C halts (for input x_1, x_2): N_1 can perform the correct (non-cheating) sequence finished by y with the maximal intermediate changes of COD. If N_2 "wants" to reach the same marking, it must fire the same number of transitions counted in SC; but, not having y, it must digress from the path of N_1 (N_2 cheats, i.e. uses some t_i^Z or t'_i instead of t_i^{NZ}) and can not reach at the same time the same value of COD (it is clear from the idea of COD as a binary number). Hence $\mathcal{R}(N_1) \not\subseteq \mathcal{R}(N_2)$ implying $\mathcal{R}(N_1) \neq \mathcal{R}(N_2)$.

If C does not halt : N_1 and N_2 only differ in the transition y. If N_1 uses it in some firing sequence (it can be only once – at the end of the sequence) it means that no t_i' has been fired and a token has been put in the $HALT$ place s_n; hence at least one firing of t_i^Z was cheating. After performing the same sequence with one t_i' instead of t_i^Z, N_2 reaches the same marking by firing x (instead of y).
Hence $\mathcal{R}(N_1) \subseteq \mathcal{R}(N_2)$ and $\mathcal{R}(N_1) = \mathcal{R}(N_2)$. □

Remark. The construction can be slightly modified so that also the reachability sets restricted to unbounded places have the same properties.

4 Decidability Results

4.1 Deterministic Nets

First we consider one-to-one labelled (or "unlabelled") Petri nets. It is clear that the bisimilarity problem is the same as the language equivalence problem in that case; the latter is known to be recursively equivalent to the reachability problem (cf. [H75]) which is known to be decidable from [M84]. (The *reachability problem* is to decide for a given Petri net N and a marking M whether $M \in \mathcal{R}(N)$.)

We do not give details here but [J93] shows another reduction of the language equivalence to the reachability problem. The reduction is simpler than that in [H75] and allows a straightforward generalization.

This generalization shows decidability of bisimilarity for *deterministic nets*, which are nets where no reachable marking enables two different transitions with the same labels. We can even allow the nets to be *deterministic up to bisimilarity* – in such a net, different transitions with the same labels can be enabled simultaneously but their firings have to lead to bisimilar results. It even suffices when *one* of the nets is deterministic up to bisimilarity.

Hence we have the following theorem (the proof is shown in [J93]).

Theorem 4.1. *Bisimilarity is decidable for two labelled Petri nets, supposing one of them is deterministic up to bisimilarity (hence if one of them is deterministic, hence if one of them is one-to-one labelled).*

[J93] also contains the following technical complexity results.

Lemma.
1. The bisimilarity problem for one-to-one labelled Petri nets is at least as hard as the reachability problem (and can be reduced to it).
2. The problem, whether a given net is deterministic, can be reduced to the coverability problem and is at least as hard.
3. The problem, whether a given net is deterministic up to bisimilarity, can be reduced to the reachability problem and is at least as hard.

The *coverability problem* is to decide for a given Petri net N and a marking M whether there is $M' \in \mathcal{R}(N)$ such that $M' \geq M$ ($\geq$ taken componentwise).

4.2 Semilinear Bisimulations

For our aims, we can suppose nets where each transition t has at least one input place p ($F(p,t) \geq 1$); if not, we can always add such p with 1 token and arcs (p,t), (t,p). Also notice the obvious fact that if we add to a net N another net N' with the zero initial marking (we simply put N and N' beside each other) then the resulting net is bisimilar to N (N' has no effect).

Hence it is clear that, without loss of generality, we can only consider bisimilarity for the case of both nets having the same static structure (they differ in initial markings only).

Then a bisimulation is, in fact, a relation on $\mathcal{N}^n$ for the relevant n. An equivalence relation R on $\mathcal{N}^n$ will be called a *congruence* if $(u,v) \in R$ implies $(u+w, v+w) \in R$ for any $w \in \mathcal{N}^n$ (addition taken componentwise).

As we already mentioned, the recent result of [CHM93] shows, in fact, that bisimilarity is decidable for the class of Petri nets where the bisimulation equivalence (the greatest bisimulation) is a congruence.

We extend this result using the notion of semilinear sets (cf. e.g. [GS66]).

Definition. A *set* $B \subseteq \mathcal{N}^k$ of k-dimensional nonnegative vectors is *linear* if there are vectors b (*basis*), $c_1, c_2, ..., c_n$ (*periods*) from $\mathcal{N}^k$ such that
$B = \{b + x_1c_1 + x_2c_2 + \ldots + x_nc_n \mid x_i \in \mathcal{N}, 1 \leq i \leq n\}$.
B is a *semilinear set* if it is a finite union of linear sets.

Labelled Petri nets are a special case of finitely branching transition systems. Generally, non-bisimilarity is semi-decidable for such systems. (cf. e.g. [M89], [CHS92]). Hence semi-decidability of bisimilarity is sufficient to show decidability.

Theorem 4.2. *For the class of (couples of) labelled Petri nets where bisimilarity implies the existence of a semilinear bisimulation relating the initial markings, bisimilarity is decidable.*

Proof. Due to decidability of Presburger arithmetic (theory of addition) (see e.g. [O78]), it can be verified whether a given semilinear set is a bisimulation (w.r.t. two given nets); it is not difficult to verify that the conditions from the definition of bisimulation can be then expressed by a Presburger formula.
Semi-decidability (and hence decidability) of bisimilarity is then clear: generate successively all semilinear sets and verify for each of them if it relates the initial markings and is a bisimulation (deciding the relevant Presburger formula). □

Remark. The theorem could be generalized in an obvious way: "a semilinear bisimulation" is replaced by "a bisimulation from $\mathcal{C}$" where $\mathcal{C}$ is an effectively generable class of relations and where it is decidable for a given relation from $\mathcal{C}$ whether it is a bisimulation relating the initial markings (states).

The fact that Theorem 4.2. is really an extension of the mentioned result of [CHM93] follows from a theorem proved in [ES69].

Theorem.(Th.II in [ES69]) *Every congruence in a finitely generated commutative monoid M is a rational (or semilinear) subset of $M \times M$.*

The monoid in our case is the set $\mathcal{N}^n$ with (vector) addition.

5 Additional remarks

Recall that (*n-dimensional*) *vector addition systems* (*VASs*) are isomorphic to (reachability sets of) Petri nets (with n places) without self-loops (without both (p,t),(t,p) as arcs). Hopcroft and Pansiot in [HP79] introduce also $VASSs$ ($VASs$ with an additional finite state control); $n-dim$ $VASS$ are, in fact, Petri nets with (at most) n unbounded places. They show that any $2-dim$ $VASS$ (unlike $3-dim$) and any $5-dim$ VAS (unlike $6-dim$) is an effectively computable semilinear set; hence the equality problem is decidable for them. (The complexity of the problem for $2-dim$ $VASSs$ is studied in [HRHY86].) [HP79] also show how any $n-dim$ $VASS$ can be simulated by an $(n+3)-dim$ VAS; it can be done by a Petri net with $n+2$ places (using self-loops).

The proof of Theorem 3.2. can be easily modified to show undecidability for (very restricted subclasses of) $5-dim$ $VASS$ and $8-dim$ VAS (leaving the dimensions 3,4, resp. 6,7, open).

For bisimilarity and (prefix) language equivalence we have undecidability for (a very restricted subclass of) Petri nets with 2 unbounded places. My conjecture is that it is decidable for the case of 1 unbounded place. E.g. I think that in that case bisimulation equivalence is semilinear.

It might be also interesting to find out the relation betweeen the deterministic nets and "semilinear bisimulations" and, in the whole, better explore the "decidability border" for bisimilarity.

Acknowledgements.
I would like to thank J.Bradfield, P.Čížek, Y.Hirshfeld and C.Stirling for helpful discussions; special thanks to S.Christensen whose observation helped to simplify the construction used in the undecidability proof.
I also thank to the STACS program committee member who drew my attention to the paper [ES69].

References

[ABS91] Autant C., Belmesk Z., Schnoebelen Ph.: Strong bisimilarity on nets revisited; PARLE'91, LNCS 506, 295-312

[B73] Baker H.: Rabin's proof of the undecidability of the reachability set inclusion problem of vector addition systems; MAC - Memo 79, MIT (1973)

[CHS92] Christensen S., Hüttel H., Stirling C.: Bisimulation equivalence is decidable for all context-free processes; CONCUR'92, LNCS 630, 138-147

[CHM93] Christensen S., Hirshfeld Y., Moller F.: Bisimulation equivalence is decidable for all Basic Parallel Processes; CONCUR'93, LNCS 715, 143-157

[ES69] Eilenberg S., Schützenberger M.: Rational sets in commutative monoids; J. of Algebra 13, 1969, 173-191

[GS66] Ginsburg S.,Spanier E.: Semigroups, Presburger formulas, and languages; Pacific J.of Mathematics, 16(2),1966, 285-296

[H75] Hack M.: Decision problems for Petri nets and vector addition systems; MAC Tech.Memo 53, MIT (1975)

[H76] Hack M.: The equality problem for vector addition systems is undecidable; TCS 2, 1976, 77-95

[Hi93] Hirshfeld Y.: Petri nets and the equivalence problem; to appear in Proc. of Computer Science Logic '93; Swansea, Wales, September 13-17, 1993.

[HP79] Hopcroft J., Pansiot J.: On the reachability problem for 5-dimensional vector addition systems; TCS 8, 1979, 135-159

[HRHY86] Howell R., Rosier L., Huynh D., Yen H.: Some complexity bounds for problems concerning finite and 2-dimensional vector addition systems with states; TCS 46, 1986, 107-140

[J93] Jančar P.: Decidability questions for bisimilarity of Petri nets and some related problems; Techn. rep. ECS-LFCS-93-261, Univ. of Edinburgh, UK, April 1993

[M84] Mayr E.: An algorithm for the general Petri net reachability problem; SIAM J.Comput. 13(3), 1984, 441-460

[M67] Minsky M.: *Computation: Finite and Infinite Machines*; Prentice Hall, 1967

[M89] Milner R.: *Communication and Concurrency*; Prentice Hall 1989

[O78] Oppen D.C.: A $2^{2^{2^{pn}}}$ upper bound on the complexity of Presburger Arithmetic; JCSS 16, 1978, 323-332

[P81] Peterson J.L.: *Petri Net Theory and the Modeling of Systems*; Prentice Hall 1981

[VV81] Valk R., Vidal-Naquet G.: Petri nets and regular languages; JCSS 23, 1981, 299-325