Information Processing Letters 109 (2009) 850–855

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

EXPSPACE lower bounds for the simulation preorder between a communication-free Petri net and a finite-state system ☆

Sławomir Lasota

Institute of Informatics, Warsaw University, Warsaw, Poland

ARTICLE INFO

Article history:
Received 20 December 2008
Received in revised form 1 April 2009
Accepted 2 April 2009
Available online 5 April 2009
Communicated by A. Muscholl

Keywords:
Petri net
BPP-net
Simulation preorder and equivalence
Trace preorder and equivalence
Concurrency

ABSTRACT

We investigate the simulation preorder between finite-state systems and a simple subclass of BPP-nets (communication-free nets). We show EXPSPACE lower bounds for the simulation problems, in both directions, as well as for the simulation equivalence. Our results improve PSPACE and co-NP lower bounds for the simulation between finite-state systems and BPP-nets, given by Kučera and Mayr in [A. Kučera, R. Mayr, Simulation preorder over simple process algebras, Information and Computation 173 (2) (2002) 184–198].

Fix a finite labeling set $\mathtt{Act}$. A *transition system* $\mathcal{A}$ is determined by a (possibly infinite) set of states S, an initial state $s_{\mathtt{init}} \in S$, and a family of binary transition relations $\xrightarrow{a}$ over states, one for each $a \in \mathtt{Act}$. We define the simulation preorder in the standard way. Given two transition systems $\mathcal{A}$ and $\mathcal{B}$, consider a relation R between states of $\mathcal{A}$ and states of $\mathcal{B}$ such that for any $(q_\mathcal{A}, q_\mathcal{B}) \in R$, any $a \in \mathtt{Act}$, and any state $q'_\mathcal{A}$ of $\mathcal{A}$, with $q_\mathcal{A} \xrightarrow{a} q'_\mathcal{A}$, there is $q'_\mathcal{B}$ such that $q_\mathcal{B} \xrightarrow{a} q'_\mathcal{B}$ in $\mathcal{B}$ and $(q'_\mathcal{A}, q'_\mathcal{B}) \in R$. Each such R is called a *simulation relation* between $\mathcal{A}$ and $\mathcal{B}$. We write $\mathcal{A} \preccurlyeq \mathcal{B}$ if the initial states are related by some simulation relation; equivalently, we could require that the initial states are related by *the greatest* simulation relation. We write $\mathcal{A} \simeq \mathcal{B}$, and say that $\mathcal{A}$ and $\mathcal{B}$ are *simulation equivalent*, if $\mathcal{A} \preccurlyeq \mathcal{B}$ and $\mathcal{B} \preccurlyeq \mathcal{A}$.

We investigate in this paper certain variants of the simulation problems between a finite-state system on one side, and a Petri net on the other, in both directions:

Problem: $\mathtt{FS} \preccurlyeq \mathtt{PN}$ ($\mathtt{PN} \preccurlyeq \mathtt{FS}$, respectively)
INSTANCE: a finite-state transition system $\mathcal{A}$ and a labeled Petri net $\mathcal{N}$.
QUESTION: $\mathcal{A} \preccurlyeq \mathcal{N}$ ($\mathcal{N} \preccurlyeq \mathcal{A}$, respectively)?

It is known that both problems $\mathtt{FS} \preccurlyeq \mathtt{PN}$ and $\mathtt{PN} \preccurlyeq \mathtt{FS}$ are decidable [3] and EXPSPACE-hard (e.g., the first one is a generalization of the non-termination of Petri nets).

A special case of $\mathtt{FS} \preccurlyeq \mathtt{PN}$ ($\mathtt{PN} \preccurlyeq \mathtt{FS}$) is obtained by restricting the right-hand side net $\mathcal{N}$ to be a *BPP-net*, i.e., a net with each transition having precisely one incoming arc. Note that such nets do not exhibit synchronization. These restricted problems we call $\mathtt{FS} \preccurlyeq \mathtt{BPP}$ and $\mathtt{BPP} \preccurlyeq \mathtt{FS}$, respectively. In [5] it was shown that $\mathtt{FS} \preccurlyeq \mathtt{BPP}$ is PSPACE-hard and $\mathtt{BPP} \preccurlyeq \mathtt{FS}$ is co-NP-hard. We investigate in this paper some further simplifications of the latter two problems.

The asynchronous product of two transition systems $\mathcal{A}$ and $\mathcal{B}$, denoted $\mathcal{A} \otimes \mathcal{B}$, has as states pairs $(q_\mathcal{A}, q_\mathcal{B})$ where $q_\mathcal{A}$ is a state of $\mathcal{A}$ and $q_\mathcal{B}$ a state of $\mathcal{B}$. The initial state is a pair of initial states. There is a transition $(q_\mathcal{A}, q_\mathcal{B}) \xrightarrow{a} (q'_\mathcal{A}, q'_\mathcal{B})$ if either $q_\mathcal{A} \xrightarrow{a} q'_\mathcal{A}$ in $\mathcal{A}$ and $q'_\mathcal{B} = q_\mathcal{B}$, or $q_\mathcal{B} \xrightarrow{a} q'_\mathcal{B}$ in $\mathcal{B}$ and $q'_\mathcal{A} = q_\mathcal{A}$. For a finite-state system $\mathcal{A}$, with state-space S and initial state $s_{\mathtt{init}}$, by $\mathcal{A}!$

☆ This work has been partially supported by Polish government grant No. N206 008 32/0810.

E-mail address: sl@mimuw.edu.pl.

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2009.04.003

we mean the infinite asynchronous product. Its states are all infinite sequences of states from S. There is a transition $(s_i)_{i<\omega} \xrightarrow{a} (s'_i)_{i<\omega}$ in $\mathcal{A}!$ if $s_i \xrightarrow{a} s'_i$ in $\mathcal{A}$ for some i, and $s'_j = s_j$ for all $j \neq i$. The initial state of $\mathcal{A}!$ is given by $s_i = s_{\texttt{init}}$, for all $i < \omega$. Equivalently, $\mathcal{A}!$ may be understood as the smallest solution of the equation $\mathcal{A}! = \mathcal{A}! \otimes \mathcal{A}$.

Two core problems investigated in this paper are the following:

Problem: $\texttt{FS} \preccurlyeq \texttt{FS!}$ ($\texttt{FS!} \preccurlyeq \texttt{FS}$, respectively)
INSTANCE: finite-state transition systems $\mathcal{A}, \mathcal{B}$.
QUESTION: $\mathcal{A} \preccurlyeq \mathcal{B}!$ ($\mathcal{B}! \preccurlyeq \mathcal{A}$, respectively)?

It is easy to see that $\texttt{FS} \preccurlyeq \texttt{BPP}$ and $\texttt{BPP} \preccurlyeq \texttt{FS}$ subsume $\texttt{FS} \preccurlyeq \texttt{FS!}$ and $\texttt{FS!} \preccurlyeq \texttt{FS}$, respectively, as one may easily describe, using a BPP-net, the behavior of $\mathcal{B}!$, for a finite-state $\mathcal{B}$. It is sufficient to replace each transition $q_{\texttt{init}} \xrightarrow{a} q$ of $\mathcal{B}$, outgoing from its initial state $q_{\texttt{init}}$, by a BPP-transition:

$$q_{\texttt{init}} \xrightarrow{a} q \otimes q_{\texttt{init}}.$$

In this way, intuitively, whenever a transition leaves $q_{\texttt{init}}$, a fresh copy of $\mathcal{B}$ is spawn, in state $q_{\texttt{init}}$. All the other transitions in $\mathcal{B}$ remain unchanged.

Therefore any lower bound for the two core problems applies immediately to the more general problems as well. Our main results are:

Theorem 1. $\texttt{FS} \preccurlyeq \texttt{FS!}$ *is* EXPSPACE-*hard.*

Theorem 2. $\texttt{FS!} \preccurlyeq \texttt{FS}$ *is* EXPSPACE-*hard, even if both given finite-state systems are deterministic.*

By an elaboration of the proof of the last result, we show that the simulation equivalence problem $\texttt{FS} \simeq \texttt{FS!}$ is EXPSPACE-hard as well:

Theorem 3. *Deciding whether* $\mathcal{A} \simeq \mathcal{B}!$ *is* EXPSPACE-*hard, even if* $\mathcal{A}$ *is deterministic.*

Our results are an improvement of the lower bounds for $\texttt{FS} \preccurlyeq \texttt{BPP}$ and $\texttt{BPP} \preccurlyeq \texttt{FS}$ given in [5]:

Corollary 1. $\texttt{FS} \preccurlyeq \texttt{BPP}$, $\texttt{BPP} \preccurlyeq \texttt{FS}$ *and* $\texttt{FS} \simeq \texttt{BPP}$ *are* EXPSPACE-*hard.*

A trace of $\mathcal{A}$ is any finite sequence $a_1 \ldots a_n \in \texttt{Act}^*$ such that there is a sequence of transitions $s_{\texttt{init}} \xrightarrow{a_1} s_1 \cdots \xrightarrow{a_n} s_n$ in $\mathcal{A}$, for some states $s_1, \ldots, s_n$. The *trace inclusion* $\mathcal{A} \subseteq \mathcal{B}$ holds if any trace of $\mathcal{A}$ is also a trace of $\mathcal{B}$. $\mathcal{A}$ and $\mathcal{B}$ are *trace equivalent*, $\mathcal{A} \equiv \mathcal{B}$, if both $\mathcal{A} \subseteq \mathcal{B}$ and $\mathcal{B} \subseteq \mathcal{A}$. In general the trace inclusion is strictly coarser than the simulation preorder. However, they collapse when the "bigger" system is deterministic. Hence by Theorem 2 and by inspecting the proof of Theorem 3 we get:

Corollary 2. *Deciding whether* $\mathcal{B}! \sqsubseteq \mathcal{A}$ *is* EXPSPACE-*hard, for any relation* $\sqsubseteq$ *which lies between the simulation equivalence* $\simeq$ *and the trace equivalence* $\equiv$*, or between the simulation preorder* $\preccurlyeq$ *and the trace inclusion* $\subseteq$.

In Sections 1–3 we present the proofs of Theorems 1–3, by reduction from two EXPSPACE-complete questions concerning Petri nets: non-termination and place-markability (or control-state reachability). We prefer to work with nondeterministic counter machines without zero-tests, purely for technical convenience. Our crucial observation is that it is possible to separate the control states of a machine (described by $\mathcal{A}$, say) from its counters (described by $\mathcal{B}!$) in such a way that existence of a simulation between $\mathcal{A}$ and $\mathcal{B}!$ (or between $\mathcal{B}!$ and $\mathcal{A}$) corresponds faithfully to an answer to the above-mentioned EXPSPACE-complete questions. In this respect our approach is slightly related to those used in [1] and [9], to obtain the lower bound for LTL model-checking of BPP-nets, and for the simulation of a finite-state system by a product of such systems, respectively. However our technique is different from those used in the cited papers.

Related research. No upper bound, except for decidability, is known for the complexity of the simulation or trace inclusion between a finite-state system and a Petri net (or a BPP-net), in whatever direction. Decidability was shown in [3].

To our knowledge, the complexity of the core problems themselves was not studied before. Related problems are the simulation and trace pre-order between a given finite-state system $\mathcal{A}$ and a composition of given finite-states systems $\mathcal{B}_1 \ldots \mathcal{B}_n$:

INSTANCE: finite state transition systems $\mathcal{A}, \mathcal{B}_1, \ldots, \mathcal{B}_n$
QUESTION 1: $\mathcal{A} \preccurlyeq \mathcal{B}_1 \otimes \cdots \otimes \mathcal{B}_n$?
QUESTION 2: $\mathcal{B}_1 \otimes \cdots \otimes \mathcal{B}_n \preccurlyeq \mathcal{A}$?
QUESTION 3: $\mathcal{B}_1 \otimes \cdots \otimes \mathcal{B}_n \subseteq \mathcal{A}$?

Question 1 is trivially in EXPTIME; the EXPTIME lower bound was shown in [9]. Question 3 is trivially PSPACE-hard; the PSPACE-algorithm was given in [15] basing on an earlier insight of [13]. Question 2 may be thus answered in PSPACE, when $\mathcal{A}$ is assumed to be deterministic, but the exact complexity of the general case is not known. Our core problems may be readily seen as infinitary versions of Questions 1 and 2. E.g., $\mathcal{A} \preccurlyeq \mathcal{B}!$, for $\mathcal{B} = \mathcal{B}_1 + \cdots + \mathcal{B}_n$ the nondeterministic choice of $\mathcal{B}_1, \ldots, \mathcal{B}_n$, is the same as $\mathcal{A} \preccurlyeq \mathcal{B}_1! \otimes \cdots \otimes \mathcal{B}_n!$. It is however not clear whether the core problems subsume the above-mentioned questions. On the other hand, $\texttt{FS} \preccurlyeq \texttt{BPP}$ and $\texttt{BPP} \preccurlyeq \texttt{FS}$ certainly do subsume Questions 1 and 2.

The EXPTIME lower bound was shown for certain variants of Questions 1 and 2, where both sides are a composition of finite-state systems which moreover synchronize on certain actions [7], and hiding operator is allowed [12]. In [10] the EXPSPACE lower bound was given for the latter variant of trace equivalence.

As usually, the bisimulation equivalence is more tractable than the simulation equivalence (preorder). For instance, the bisimulation equivalence between a finite-state system and a BPP-net is in PTIME [4].

Game-theoretic characterization. A simulation relation witnessing $\mathcal{A} \preccurlyeq \mathcal{B}$ may be seen as a winning strategy of Duplicator in the *simulation game* over $\mathcal{A}$ and $\mathcal{B}$. The game

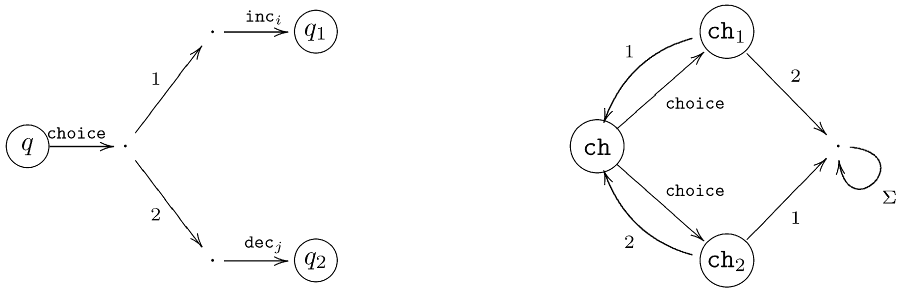

Fig. 1.

is played by two players, Spoiler and Duplicator, and proceeds in rounds. The first round starts in the configuration $(q_{\mathcal{A}}, q_{\mathcal{B}})$ consisting of initial states of $\mathcal{A}$ and $\mathcal{B}$, respectively. Each round consists of one Spoiler's move, followed by one Duplicator's move. Spoiler chooses a label $a \in \mathtt{Act}$ and a state $q'_{\mathcal{A}}$ in $\mathcal{A}$ such that $q_{\mathcal{A}} \xrightarrow{a} q'_{\mathcal{A}}$. Then, Duplicator answers by choosing a state $q'_{\mathcal{B}}$ in $\mathcal{B}$ such that $q_{\mathcal{B}} \xrightarrow{a} q'_{\mathcal{B}}$ in $\mathcal{B}$. Afterwards, the next round continues from the new configuration $(q'_{\mathcal{A}}, q'_{\mathcal{B}})$. One of the players wins, if a configuration is reached such that the other player has no possible move. Otherwise, the play is infinite – in this case Duplicator wins unconditionally. $\mathcal{A} \preccurlyeq \mathcal{B}$ if and only if Duplicator has a winning strategy in the simulation game over $\mathcal{A}$ and $\mathcal{B}$.

1. EXPSPACE lower bound for $\mathtt{FS} \preccurlyeq \mathtt{FS}!$

We provide a reduction from non-termination problem for nondeterministic counter machines without zero-tests. Instead, one could equivalently consider Petri nets, or vector addition systems with states. EXPSPACE-hardness of non-termination follows from [8]; it is EXPSPACE-complete [11]. The intuition is that the protocol between Spoiler and Duplicator during the simulation game enables them to imitate jointly a computation of a given counter machine.

A counter machine without zero-tests has a set of states Q, a distinguished initial state $q_{\mathtt{init}} \in Q$, a set of counters $C = \{c_1, \ldots, c_k\}$, and a finite set of increment or decrement transitions of the following form:

in state q, increment (decrement) c_i and go to state q'.

A counter can always be incremented, but can only be decremented if it is greater than zero. A computation of $\mathcal{M}$ is a (finite or infinite) sequence of states q_i and counter valuations $v_i : C \to \mathbb{N}$

$$(q_0, v_0) \quad (q_1, v_1) \quad (q_2, v_2) \quad \ldots$$

consistent with transitions of $\mathcal{M}$. We assume that $\mathcal{M}$ starts in $q_{\mathtt{init}}$, with all counters set to 0: $q_0 = g_{\mathtt{init}}$, $v_0(c_i) = 0$ for $i \leqslant k$. We say that $\mathcal{M}$ *terminates* if all computations of $\mathcal{M}$ are finite.

Assume, without loss of generality, that each state of $\mathcal{M}$ has precisely two outgoing transitions. (If there is only one, add a duplicate identical transition. If there are more than two, organize them into a binary tree, and use increments of an additional artificial counter on the branches. If there is no transition, add two decrementing self-loops. Note that this construction does not affect termination.)

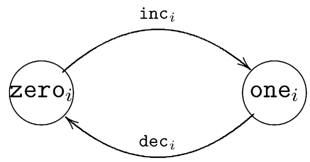

Fig. 2.

We describe now a construction that, for a given machine $\mathcal{M}$, yields two finite-state systems $\mathcal{A}$, $\mathcal{B}$ such that $\mathcal{M}$ does not terminate if and only if $\mathcal{A} \preccurlyeq \mathcal{B}!$. During the simulation game, $\mathcal{A}$ will keep track of the actual control state of $\mathcal{M}$ while the values of counters will be stored in $\mathcal{B}!$.

Let q be an arbitrary state of $\mathcal{M}$. To simplify the presentation assume that the two transitions from q are the following: increment c_i and go to state q_1; decrement c_j and go to q_2 (all other possibilities, i.e., two increments or two decrements, are dealt with in exactly the same way).

$\mathcal{A}$ will contain the transitions drawn on the left-hand side of Fig. 1, while the $\mathcal{B}$ system will contain the transitions drawn on the right-hand side (see Fig. 1).

As a whole, $\mathcal{A}$ will contain such transitions for each state q. Hence states of $\mathcal{A}$ will include the states of $\mathcal{M}$, and additional three temporary states for each state q of $\mathcal{M}$. The initial state is $q_{\mathtt{init}}$.

Σ stands for the whole alphabet hence the state of $\mathcal{B}$ with the Σ-loop is unconditionally winning for Duplicator. The Spoiler plays first with the choice-move, but the actual choice between transition 1 and 2 is done as a response of Duplicator. Then in his second move, Spoiler has to respect the Duplicator's choice, otherwise he loses.[1] Finally, in the third step, Spoiler checks whether the transition of $\mathcal{M}$ chosen by Duplicator is doable (this is relevant only in case of decrement, as increments are always doable). To answer this moves, in the $\mathcal{B}$ system there is also a small component for each counter c_i (see Fig. 2).

To make all the right-hand side components into a single finite-state $\mathcal{B}$, we simply collapse state $\mathtt{ch}$ and states

[1] This construction in an adaptation of the "Duplicator's choice" technique of [2,14]. While mostly used in the context of bisimulation, it was also applied for the simulation equivalence in [6].

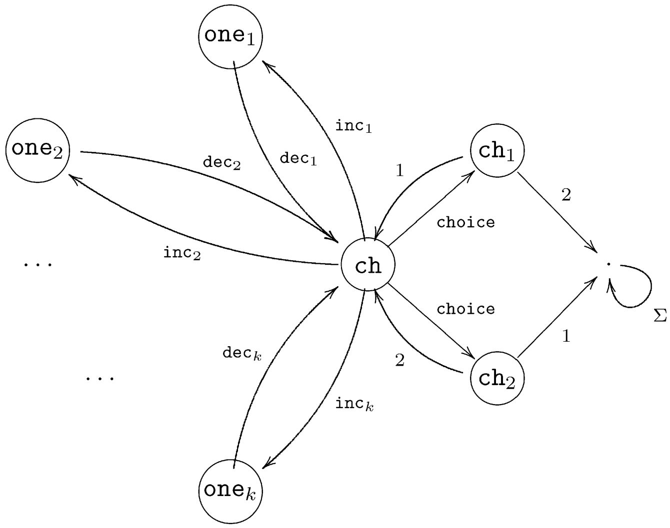

Fig. 3.

$\texttt{zero}_1, \ldots, \texttt{zero}_k$ into one initial state ch. It is irrelevant which copy of $\mathcal{B}$ will be used by Duplicator to decide between 1 and 2. The very construction of $\mathcal{A}$ guarantees that in each position in the game, at most one copy of $\mathcal{B}$ is in $\texttt{ch}_1$ or $\texttt{ch}_2$ state, as each Spoiler's choice move is immediately followed by his 1 or 2 move. The whole $\mathcal{B}$ system will look as shown in Fig. 3.

The states of $\mathcal{B}$ are $\{\texttt{ch}, \texttt{ch}_1, \texttt{ch}_2\} \cup \{\texttt{one}_i\}_{i=1\ldots k} \cup$ {the loop state}. The initial state is ch. $\Sigma = \{\texttt{choice}, 1, 2\} \cup \{\texttt{inc}_i, \texttt{dec}_i\}_{i=1\ldots k}$. As explained above, precisely one of copies of $\mathcal{B}$ will be used for making choices by Duplicator at each stage of the game; all the other copies will only be used to store the values of the counters.

We have: $\mathcal{M}$ does not terminate iff Duplicator wins, i.e., iff $\mathcal{A} \preccurlyeq \mathcal{B}!$.

Indeed, if $\mathcal{M}$ has an infinite computation, Duplicator's strategy is to follow precisely this computation. If Spoiler does not respect a Duplicator choice between 1 and 2, it is immediately punished by Duplicator in $\mathcal{B}$ by going to the loop state. Otherwise, assume that all computations of $\mathcal{M}$ are finite. Even if Spoiler has no actual initiative in the game, it is guaranteed that after a finite number of steps of $\mathcal{M}$ no further transition is doable. This means that Duplicator has no response to Spoiler's check (note that this can only take place in case of decrement). This completes the proof.

2. EXPSPACE lower bound for FS! $\preccurlyeq$ FS

We will give a reduction from the control state reachability of a nondeterministic counter machine without zero-tests (this is equivalent to markability of a given place of a Petri net, an EXPSPACE-complete problem [8,11]). As before we assume that in each state of $\mathcal{M}$ there are precisely two transitions, except for the distinguished state q_f, from which there are no outgoing transitions (hence $\mathcal{M}$ always stops when it enters q_f). Observe that the transformation sketched in the previous section, applied to all states except for q_f, does not affect control-state reachability of q_f.

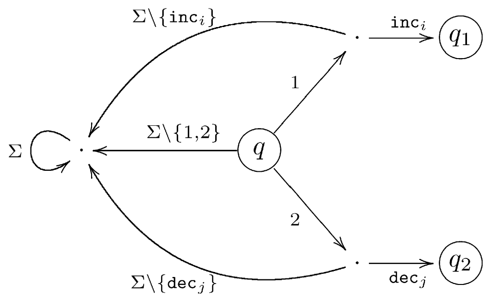

Fig. 4.

Assume additionally, without loss of generality, that the initial state $q_{\texttt{init}}$ is never revisited in any computation of $\mathcal{M}$.

Given such an $\mathcal{M}$, we construct finite-state systems $\mathcal{A}$ and $\mathcal{B}$ such that $\mathcal{M}$ has a finite computation that ends in q_f iff $\mathcal{B}! \not\preccurlyeq \mathcal{A}$.

Now the Spoiler plays in one of the (infinitely many) copies of $\mathcal{B}$ and Duplicator responds in $\mathcal{A}$. Let q be an arbitrary state of $\mathcal{M}$ different from q_f, and let the two transitions from q be the following: increment c_i and go to state q_1; decrement c_j and go to q_2 (again, the other cases are dealt analogously). $\mathcal{A}$ will contain the transitions shown in Fig. 4.

Thus states of $\mathcal{A}$ include the states of $\mathcal{M}$, an additional Σ-loop state, and two temporary states for each state of $\mathcal{M}$, except for q_f. Similarly as before, $q_{\texttt{init}}$ will be the initial state in $\mathcal{A}$. We additionally decide that transitions from $q_{\texttt{init}}$ will use different labels $\widetilde{1}$ and $\widetilde{2}$, instead of 1 and 2 (see Fig. 5).

Finally, there is no transition from q_f in $\mathcal{A}$.

The $\mathcal{B}$ system will contain the transitions shown in Fig. 6.

The initial state of $\mathcal{B}$ is $\widetilde{\texttt{ch}}$. $\Sigma = \{\widetilde{1}, \widetilde{2}, 1, 2\} \cup \{\texttt{inc}_i, \texttt{dec}_i\}$. $\mathcal{A}$ is constructed so that at each stage of the game, Duplicator narrows down the possible Spoiler moves. Spoiler may either choose between 1 or 2 only (except

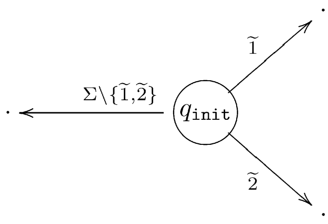

Fig. 5.

for the very first move, when the choice is between $\widetilde{1}$ and $\widetilde{2}$), which corresponds to a choice of the next transition of $\mathcal{M}$ to be executed; or may execute the appropriate single counter operation. As long as Spoiler correctly follows a run of $\mathcal{M}$ in that way, Duplicator can only stay passive. Hence, if q_f is reachable by a finite computation, this computation may be used as a Spoiler's winning strategy. If q_f is not reachable, no matter what computation is chosen by Spoiler, finite or infinite, the computation chosen never reaches q_f, and hence Duplicator wins.

Remark 1. Both $\mathcal{A}$ and $\mathcal{B}$ in the above reduction are deterministic. Since $\mathcal{A}$ is deterministic, all relations between the simulation preorder and the trace inclusion collapse, and therefore the hardness result applies to all such relations.

Remark 2. Similarly as in previous section, we could collapse states $\mathtt{ch}$, $\widetilde{\mathtt{ch}}$ and $\mathtt{zero}_1, \ldots, \mathtt{zero}_k$ into one initial state of $\mathcal{B}$, as it is again irrelevant in which copy of $\mathcal{B}$ Spoiler does a choice between 1 or 2. We prefer to keep the distinct $\widetilde{\mathtt{ch}}$ state, as it will make the analysis of the simulation equivalence simpler in the next section.

3. EXPSPACE lower bound for FS $\simeq$ FS!

We slightly elaborate on the proof from the last section to obtain EXPSPACE-hardness of the simulation equivalence problem $\mathtt{FS} \simeq \mathtt{FS}!$. The usual approach to reduction of the simulation preorder to equivalence is to use the equivalence $\mathcal{B}! \preccurlyeq \mathcal{A} \iff \mathcal{A} + \mathcal{B}! \simeq \mathcal{A}$, where $+$ stands for the nondeterministic choice. Indeed, one of the simulations, $\mathcal{A} \preccurlyeq \mathcal{A} + \mathcal{B}!$, is always true; and in the game for $\mathcal{A} + \mathcal{B}! \preccurlyeq \mathcal{A}$, Spoiler will surely play in $\mathcal{B}!$. However,

$$\mathcal{B}! \preccurlyeq \mathcal{A} \quad \iff \quad (\mathcal{A} + \mathcal{B})! \simeq \mathcal{A} \tag{1}$$

does not hold in general, as an unbounded number of copies of $\mathcal{A}$ is available to Spoiler in this case. Henceforth, reduction of $\mathtt{FS}! \preccurlyeq \mathtt{FS}$ to $\mathtt{FS} \simeq \mathtt{FS}!$ requires slightly more care.

It turns out that we are able to recover equivalence (1) when $\mathcal{A}$ and $\mathcal{B}$ are the specific finite-state systems constructed in the proof of the lower bound for $\mathtt{FS}! \preccurlyeq \mathtt{FS}$ in Section 2. We need to show:

$$q_f \text{ is not reachable in } \mathcal{M} \quad \iff \quad (\mathcal{A}+\mathcal{B})! \preccurlyeq \mathcal{A}. \tag{2}$$

Recall that the initial states $q_{\mathtt{init}}$ and $\widetilde{\mathtt{ch}}$ in $\mathcal{A}$ and $\mathcal{B}$, respectively, are never revisited, hence $\mathcal{A}+\mathcal{B}$ is obtained by simply gluing together the initial states. Recall also that the labels $\widetilde{1}$ and $\widetilde{2}$ lead in $\mathcal{A}$ to the Σ-loop state from each state different from $q_{\mathtt{init}}$. Hence, in the game for $(\mathcal{A}+\mathcal{B})! \preccurlyeq \mathcal{A}$, Spoiler can play in at most one copy of $\mathcal{A}$, as any further use of label $\widetilde{1}$ or $\widetilde{2}$ would be immediately punished by Duplicator.

Assume that in the very first move Spoiler chooses a move in $\mathcal{A}$. In the rest of the play, s(he) will be deprived of 1 and 2 actions from $\mathcal{B}$. Hence at each move, Spoiler has a choice to play in (the single copy of) $\mathcal{A}$, or to activate a copy of $\mathcal{B}$ by an $\mathtt{inc}_i$ move. But each usage of a transition of $\mathcal{B}$ will be immediately punished by Duplicator, as in the next round Spoiler would be incapable to do 1 or 2 as required. The only exception is when Duplicator is in state q_f.

Therefore, if state q_f is not reachable in $\mathcal{M}$, Duplicator wins both when Spoiler starts in $\mathcal{A}$ (and uses no $\mathcal{B}$ according to the above considerations); and also when Spoiler starts in $\mathcal{B}$ (in this case no copy of $\mathcal{A}$ can be activated by

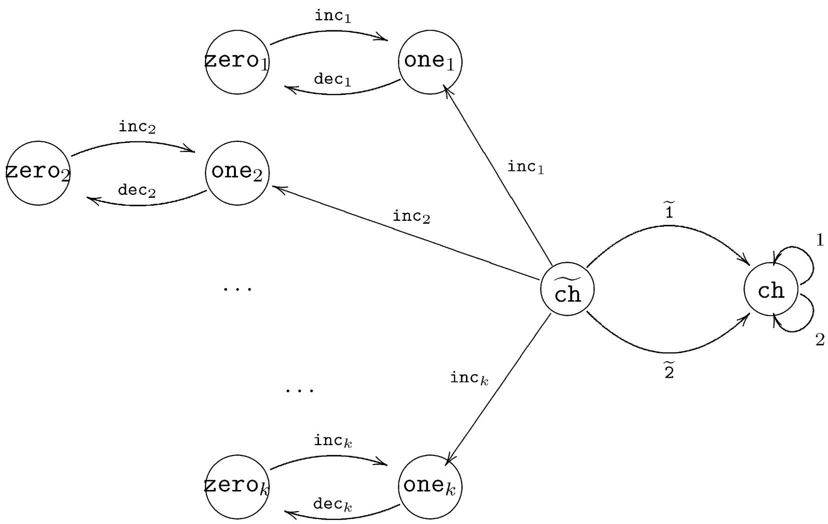

Fig. 6.

Spoiler). If q_f is reachable, Spoiler wins, for instance by playing solely in the copies of $\mathcal{B}$, without using $\mathcal{A}$ at all.

Remark 3. As $\mathcal{A}$ is deterministic, the right-hand side of (2) may be equivalently replaced by $(\mathcal{A}+\mathcal{B})! \subseteq \mathcal{A}$. Thus our proof applies to all relations between the simulation equivalence and the trace equivalence.

Acknowledgements

The author is grateful to an anonymous referee for the valuable comments, especially for a suggestion of simplification of the construction in the proof of Theorem 1 in Section 1.

References

[1] P. Habermehl, On the complexity of the linear-time μ-calculus for Petri nets, in: Proc. ICATPN'97, in: LNCS, vol. 1248, 1997, pp. 102–116.
[2] P. Jančar, High undecidability of weak bisimilarity for Petri nets, in: Proc. CAAP'95, in: LNCS, vol. 915, 1995, pp. 349–363.
[3] P. Jančar, F. Moller, Checking regular properties of Petri nets, in: Proc. CONCUR'95, in: LNCS, vol. 962, 1995, pp. 348–362.
[4] M. Kot, Z. Sawa, Bisimulation equivalence of a BPP and a finite-state system can be decided in polynomial time, ENTCS 138 (2005) 49–60.
[5] A. Kučera, R. Mayr, Simulation preorder over simple process algebras, Information and Computation 173 (2) (2002) 184–198.
[6] A. Kučera, R. Mayr, Why is simulation harder than bisimulation? in: Proc. CONCUR'02, in: LNCS, vol. 2421, 2002, pp. 594–609.
[7] F. Laroussinie, P. Schnoebelen, The state-explosion problem from trace to bisimulation equivalence, in: Proc. FoSSaCS'00, in: LNCS, vol. 1784, 2000, pp. 192–207.
[8] R. Lipton, The reachability problem requires exponential space, Tech. Rep. 62, Department of Computer Science, Yale University, 1976.
[9] A. Muscholl, I. Walukiewicz, A lower bound on web services composition, Logical Methods in Computer Science 4 (2–5) (2008) 1–14.
[10] A. Rabinowich, Complexity of equivalence problems for concurrent systems of finite agents, Information and Computation 139 (2) (1997) 111–129.
[11] C. Rackoff, The covering and boundedness problems for vector addition systems, Theoret. Comput. Sci. 6 (1) (1978) 223–231.
[12] Z. Sawa, Equivalence checking of non-flat systems is exptime-hard, in: Proc. CONCUR'03, in: LNCS, vol. 2761, 2003, pp. 237–250.
[13] S.K. Shukla, H.B. Hunt, I. Daniel, J. Rosenkrantz, R.E. Stearns, On the complexity of relational problems for finite state processes, in: Proc. ICALP'96, in: LNCS, vol. 1099, 1996, pp. 466–477.
[14] J. Srba, Strong bisimilarity of simple process algebras: Complexity lower bounds, Acta Informatica 39 (2003) 469–499.
[15] A. Valmari, A. Kervinen, Alphabet-based synchronisation is exponentially cheaper, in: Proc. CONCUR'02, in: LNCS, vol. 2421, 2002, pp. 161–176.