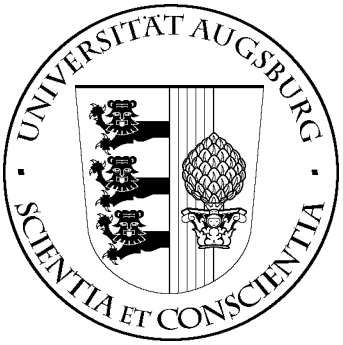
Concurrent implementation of asynchronous transition systems

Walter Vogler

Angaben zur Veröffentlichung / Publication details:

Vogler, Walter. 2005. "Concurrent implementation of asynchronous transition systems." Augsburg: Universität Augsburg.

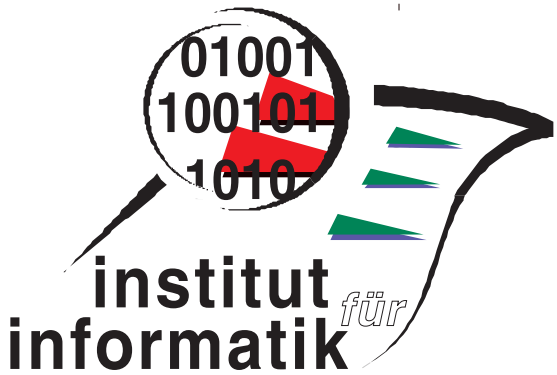
Universität Augsburg



Concurrent Implementation of Asynchronous Transition Systems

Walter Vogler

Report 1998-5 Dezember 1998



Institut für Informatik
D-86135 Augsburg

Concurrent Implementation of Asynchronous Transition Systems

Walter Vogler *

Institut für Informatik, Universität Augsburg
D-86135 Augsburg, Germany
email: vogler@informatik.uni-augsburg.de

Abstract

The synthesis problem is to decide for a deterministic transition system whether a Petri net with an isomorphic reachability graph exists and in case to find such a net (which must have the arc-labels of the transition system as transitions). In this paper, we weaken isomorphism to some form of bisimilarity that also takes concurrency into account and we consider safe nets that may have additional internal transitions. To speak of concurrency, the transition system is enriched by an independence relation to an asynchronous transition system.

Given an arbitrary asynchronous transition system, we construct an ST-bisimilar net. We show how to decide effectively whether there exists a bisimilar net without internal transitions, in which case we can also find a history-preserving bisimilar net without internal transitions. Finally, we present a construction that inserts a new internal event into an asynchronous transition system such that the result is history-preserving bisimilar; this construction can help to find a history-preserving bisimilar net (with internal transitions).

1 Introduction

One methodology for the design of asynchronous circuits takes a transition system (whose arcs are labelled with what we call events) as specification of the desired behaviour, gives it a distributed implementation as a safe Petri net and transforms the latter stepwise into a circuit, see e.g. [CKLY98]; [Yak98] gives in detail a practical example for such a development. In the synthesis of the net, it is desirable that each event of the transition system corresponds to a unique transition of the net: as [Yak98] shows, the transformation of the net may involve event refinement, which is much easier when each event is represented by one transition; only with this property, it is e.g. easy to refine an event e such that each occurrence of e in a run of the original net is replaced alternatingly by an occurrence of e_1 or one of e_2 in a run of the refined

*This work was partially supported by the DFG-project 'Halbordnungstesten'.

net. Also some existing procedures for efficient direct compilation of a net into an asynchronous circuit rely on this property.[1] Since we desire this property, it is natural that we restrict attention to deterministic transition systems.

A pivotal contribution to the synthesis problem is the theory of regions, see e.g. [NRT92]; it allows a characterization of those transition systems TS for which an elementary Petri net exists whose reachability graph is isomorphic to TS. Elementary nets are (almost) the same as safe nets without loops; a nice feature is that for such nets independence of transitions corresponds exactly to 'diamonds' in the transition system. But [PKY95] points out, that loops are very natural in particular in the context of circuits and allow to implement additional transition systems; the theory of regions is extended accordingly (almost) to general safe nets; the results can be seen as a specialization of the parametric results in [DS93]. On the other hand, [CKLY98] weakens the requirements of [NRT92] and shows that for each transition system TS satisfying the weaker requirements there exists an elementary net whose reachability graph is bisimilar to TS. For nets or transition systems (without internal events), bisimilarity and language equivalence coincide; based on [HKT92], language equivalent realization of transition systems is studied in [BBD95] for bounded nets and in [Dar98] for unbounded nets.

Finally, [Yak98] is confronted with a transition system that cannot be realized by a net in any of these approaches; since the reaction to such a situation can hardly be simply to give up, he first inserts an internal event into the transition system 'in a harmless way', i.e. he achieves in his example an implementation with a net that has an additional internal transition; in fact, the net (i.e. its reachability graph) is bisimilar to the original transition system. Implementation of transition systems in this sense is the topic of the present paper.

Thus, we are given a deterministic transition system TS as a specification of some behaviour and we want to find a (general) safe net whose transitions are the events of TS and possibly some additional internal events, and whose reachability graph is (weakly) bisimilar to TS. This ensures (but see below) that the net has essentially the desired behaviour; also, properties formulated e.g. in Hennessy-Milner logic and checked for the transition system will also hold for the net, see [Yak98] for an informal check of this type. Note that internal events could lead to a new and usually unwanted behaviour that is ignored by bisimulation, namely divergence, i.e. infinite internal computation; hence, we additionally require the net to be divergence-free.

It should be mentioned that it is easy to turn an arbitrary transition system into a Petri net when labelling of the transitions is allowed. Furthermore, constructions are known – see e.g. [BDKP91] – that turn a labelled Petri net into one where there are additional internal transitions, but otherwise each label occurs only once; the latter is the kind of net we are looking for. To the best of the author's knowledge, the known constructions either do not give a bisimilar net or introduce divergence.

We will show that it is also not too difficult to find a bisimilar divergence-free implementation for each transition system; but it turns out that it is completely sequential. This is undesirable e.g. for performance reasons; hence, an additional requirement is to preserve concurrency – for which concurrency must be specified in

[1] Thanks go to Alex Yakovlev for pointing this out to me.

the first place. We will therefore in fact start from an asynchronous transition system (ATS) which is a deterministic transition system with an additional independence relation on the events. In a categorical setting, asynchronous and similar transition systems are related to nets by giving a characterization of those ATS for which a net exists whose reachability graph (with independence relation) is isomorphic to the ATS in [DS93, NW95]; instead of requiring isomorphism, we want to compare the behaviour of an ATS and a net.

Preservation of concurrency could mean that the net should have the same step sequences or the same partial order semantics as the ATS, where the latter semantics could be defined via Petri net processes or equivalently as (Mazurkiewicz) traces; see e.g. [NW95] – also for ATS. Or one could combine this with bisimilarity and require step or history-preserving bisimilarity; for our first result, we will consider something in between: ST-bisimilarity, which combines bisimulation with a partial order semantics based on so-called interval orders, see e.g. [Vog92]. We will show that each ATS (with a weak requirement for independence) can be implemented by an ST-bisimilar divergence-free safe net.

Then, we will consider ATS with the usual strong requirement for independence; we will show how to decide whether for such an ATS there exists a safe net without internal events with a bisimilar reachability graph, and we will prove that one such net is in fact history-preserving bisimilar to the ATS. Finally, we will mention an ATS-modification that sometimes helps to find a history-preserving bisimilar divergence-free safe net with internal events; a special case is the modification used in [Yak98] and mentioned above.

We conclude this introduction by mentioning another two very interesting contributions to the synthesis problem. [Sun98, Chapter 5] considers the case that the specification of the desired behaviour is given as a temporal logic formula and presents an effective decision procedure whether there exists a safe net with possibly some additional internal events meeting the specification. [Dar98] generalizes the synthesis problem in two other ways: firstly, two regular languages are given and a (possibly unbounded) net is sought for with a language between the given ones; secondly, the problem of realizing a deterministic context-free language with a net is considered.

2 Petri nets and ST-bisimulation

This section gives a short introduction to safe Petri nets (place/transition-nets). For general information on nets, the reader is referred to e.g. [Pet81, Rei85].

In this paper, a *safe Petri net* N (or just a *net* for short) is a tuple (S, E_v, E_i, F, M_N) satisfying a number of requirements explained in the following. S, E_v and E_i, are finite disjoint sets of *places* and *visible* and *internal events*; thus, we call the elements of $E = E_v \cup E_i$ events instead of transitions. N is called *visible* if E_i is empty. $F \subseteq S \times E \cup E \times S$ is the set of *arcs* (which all have weight 1), and M_N is the *initial marking*, which is as any *marking* a subset of S. When we introduce a net N or N' etc., then we assume that implicitly this introduces its components S, E_v, E, ... or S', E'_v, ..., etc. and similarly for other tuples later on.

For each $x \in S \cup E$, the *preset* of x is ${}^\bullet x = \{y \mid (y,x) \in F\}$ and the *postset* of x is $x^\bullet = \{y \mid (x,y) \in F\}$. These notions are extended pointwise to sets, e.g.

${}^\bullet X = \bigcup_{x \in X} {}^\bullet x$. If $x \in {}^\bullet y \cap y^\bullet$, then x and y form a *loop*.

- An event a is *enabled* under a marking M, denoted by $M[a\rangle$, if ${}^\bullet a \subseteq M$.

  If $M[a\rangle$ and $M' = (M \setminus {}^\bullet a) \cup a^\bullet$, then we denote this by $M[a\rangle M'$ and say that a can *occur* or *fire* under M yielding the marking M'. (This rule is in fact a bit unusual, since $M \setminus {}^\bullet a$ and $a^\bullet$ may overlap, but even then there is only at most one token on a place under M', since markings are sets. Our rule coincides with the usual definition for the nets we will consider in the following.)

- This definition of enabling and occurrence can be extended to sequences as usual: a sequence w of events is *enabled* under a marking M, denoted by $M[w\rangle$, and yields the follower marking M' when *occurring*, denoted by $M[w\rangle M'$, if w is the empty sequence λ and $M = M'$ or $w = w'a$, $M[w'\rangle M''$ and $M''[a\rangle M'$ for some marking M'' and some event a. If w is enabled under the initial marking, then it is called a *firing sequence*.

  A marking M is called *reachable* if $\exists w \in E^* : M_N[w\rangle M$. The net is *safe* if for all reachable markings M and events a, $M[a\rangle$ implies $(M \setminus {}^\bullet a) \cap a^\bullet = \emptyset$.

General assumption All nets considered in this paper are safe and have only events with nonempty presets. (The latter is not much of a restriction since one can add to each violating event a a new marked place on a loop with a. The former can be ensured by adding complementary places, see e.g. [Dev90].) For convenience, we also assume that for each event a there is some reachable marking that enables a.

It is obvious how to generalize the firing rule to infinite sequences. It is usually desirable that a net be *divergence-free*, i.e. that no reachable marking enables an infinite sequence of internal events. Next, we lift the enabledness and firing definitions to the level of visibility:

- A sequence $v \in E_v^*$ is *visibly enabled* under a marking M, denoted by $M[v\rangle\rangle$, if there is some sequence $w \in E^*$ with $M[w\rangle$ such that v is obtained from w by deleting all internal events. If $M = M_N$, then v is called a *visible firing sequence*.

To each net N we associate an *independence relation* $I(N)$ on its events, where $a\, I(N)\, b$ if ${}^\bullet a \cup a^\bullet$ and ${}^\bullet b \cup b^\bullet$ are disjoint. Note that $I(N)$ is irreflexive and symmetric. For a marking M, $\mu \subseteq E$ is an *M-step* if the events in μ are enabled under M and pairwise independent; in this case, the events can fire in any order under M and, intuitively, also simultaneously.

We are interested in behaviour notions that capture choice *and* concurrency in a strong sense, hence in variants of bisimulation that also consider concurrency. (The basic bisimulation will be treated in the next section.) One such variant is called ST-bisimulation; its key idea is that the firing of a visible event a consists of a beginning a^+ and an end a^-, where a^+ checks the enabledness of a and consumes the input of a, while a^- produces the output. Thus, concurrency in the sense of overlapping occurrences can be observed for visible events – while internal events cannot be observed at all. This is a stronger notion of concurrency than e.g. steps; it corresponds to a partial order semantics that is weaker than causality as captured by net processes or

Mazurkiewicz traces, but is instead based on so-called interval orders (see e.g. [Vog92]) and suitable to judge temporal efficiency when events take time, see [Vog95].

If events have a beginning and an end, a system state cannot adequately be described by a marking alone; instead, it consists of a marking together with some events that have started, but have not finished yet, and it is called an ST-marking. In the corresponding firing rule, the preset of a starting event is usually subtracted from the marking immediately [GV87]; for compatibility with the next section, we use an alternative but equivalent formulation.

- An *ST-marking* (M, μ) of a net N consists of a reachable marking M and an M-step $\mu \subseteq E_v$. The *initial ST-marking* is $(M_N, \emptyset)$.
- For a visible event a, we write $(M, \mu)[a^+\rangle(M, \mu \cup \{a\})$ if $M[a\rangle$ and a is independent to all $b \in \mu$ – which in particular implies $a \notin \mu$. For a visible event a, we write $(M, \mu)[a^-\rangle(M', \mu \setminus \{a\})$ if $a \in \mu$ and $M[a\rangle M'$. Finally, for an internal event c, we write $(M, \mu)[c\rangle(M', \mu)$ if c is independent to all $b \in \mu$ and $M[c\rangle M'$. Note that in all three cases the pair reached is an ST-marking again.
- We again extend this definition to sequences and, by suppressing internal events, to visible sequences.

Two nets N and N' with the same visible events are *ST-bisimilar*, if there exists an *ST-bisimulation* between them, i.e. a relation $\mathcal{B}$ between the ST-markings of N and N' such that:

1. $\mathcal{B}$ relates the initial ST-markings.
2. If $a \in E_v$, $c \in E_i$ and $(M, \mu)\mathcal{B}(M', \mu)$ (with the same μ), then we have:
   (a) $(M, \mu)[a^+\rangle(M, \mu \cup \{a\})$ implies that for some M'' $(M', \mu)[a^+\rangle\rangle(M'', \mu \cup \{a\})$ and $(M, \mu \cup \{a\})\mathcal{B}(M'', \mu \cup \{a\})$
   (b) $(M, \mu)[a^-\rangle(M_1, \mu \setminus \{a\})$ implies that for some M_1' $(M', \mu)[a^-\rangle\rangle(M_1', \mu \setminus \{a\})$ and $(M_1, \mu \setminus \{a\})\mathcal{B}(M_1', \mu \setminus \{a\})$
   (c) $(M, \mu)[c\rangle(M_1, \mu)$ implies that $(M', \mu)[\rangle\rangle(M_1', \mu)$ and $(M_1, \mu)\mathcal{B}(M_1', \mu)$ for some M_1'
3. vice versa

In the case of general labelled nets, ST-bisimulations do not consist simply of pairs $((M, \mu), (M', \mu'))$; instead, there is an additional component that matches for each visible label a the a-labelled transitions in μ to the a-labelled transitions in μ'. This is not necessary in our setting, since here a step can contain at most one a – whereas in general there can be several a-labelled transitions in a step, i.e. there can be autoconcurrency. (What we have defined is really split-bisimilarity – see e.g. [GV97] –, which coincides with ST-bisimilarity in our setting.)

3 Asynchronous transition systems and history-preserving and ST-bisimulation

An asynchronous transition system (an ATS) A (and, more generally, a weak asynchronous transition system, a wATS) is a tuple (Q, E_v, E_i, T, q_0, I) satisfying a number of requirements explained in the following. Q is the finite set of *states* containing the *initial state* q_0; E_v and E_i are finite disjoint sets of *visible* and *internal events* and I is the irreflexive and symmetric *independence relation* on $E = E_v \cup E_i$; events a and b are *dependent* if they are not independent. A is called *visible* if E_i is empty. We speak of a *transition system*, if we are not interested in I – formally, a transition system would not have an independence relation and each wATS would have an underlying transition system.

The transition relation T is a partial function from $Q \times E$ to Q, i.e. A is deterministic over the alphabet E. We say a is *enabled* under q or can *occur* from q and write $q \xrightarrow{a}$, if T is defined for (q, a); we write $q \xrightarrow{a} p$, if $T(q, a) = p$, and speak of an (a-labelled) *arc* from q to p; q and a form a *loop* in A if $q \xrightarrow{a} q$. Similarly to the last section, $\xrightarrow{a}$ is generalized to $\xrightarrow{w}$ for $w \in E^*$ (asserting the existence of a w-labelled *path*); if $q_0 \xrightarrow{w}$, then w is called an *occurrence sequence* of A. Also in direct analogy to the last section, we define *divergence-freeness* of wATS. We write $q \stackrel{v}{\Rightarrow} q'$ if $q \xrightarrow{w} q'$ and v is the sequence of visible events obtained from w by deleting all internal events. We require that all states of A are *reachable*, i.e. for all $q \in Q$ there is some $w \in E^*$ with $q_0 \xrightarrow{w} q$.

For $q \in Q$, $\mu \subseteq E$ is a *q-step* if the events in μ are enabled under q and pairwise independent. A minimal requirement is in this case, that these events can occur in any order from q reaching the same state independently of the order. To guarantee this, we define:

A *weak asynchronous transition system* (wATS) satisfies for all independent events a and b and states q: if $q \xrightarrow{a}$ and $q \xrightarrow{b}$, then there exists some q' with $q \xrightarrow{ab} q'$ and $q \xrightarrow{ba} q'$. (This 'forward-diamond-property' only is e.g. also required in [DS93].) For an *asynchronous transition system* (ATS) such a q' also exists if $q \xrightarrow{ab}$. For convenience, we assume that for each event a of a wATS or ATS there exists some q with $q \xrightarrow{a}$. For an ATS, we furthermore assume that aIb implies that there exists some q with $q \xrightarrow{a}$ and $q \xrightarrow{b}$.

We call wATS A and A' *isomorphic* if one is obtained from the other by a bijective renaming of states that preserves initial states and transition relations.

For comparison of wATSs, we first define ordinary bisimulation: two wATSs A and A' with the same visible events are *(weakly) bisimilar*, if there exists a *bisimulation* between them, i.e. a relation $\mathcal{B} \subseteq Q \times Q'$ such that:

1. $\mathcal{B}$ relates the initial states.
2. If $a \in E_v$, $c \in E_i$ and $q \mathcal{B} q'$, then we have:
   (a) $q \xrightarrow{a} q_1$ implies that for some q_1' $q' \stackrel{a}{\Rightarrow} q_1'$ and $q_1 \mathcal{B} q_1'$
   (b) $q \xrightarrow{c} q_1$ implies that for some q_1' $q' \Rightarrow q_1'$ and $q_1 \mathcal{B} q_1'$

3. vice versa

Note that in our implementation problem the desired behaviour will be given as a visible wATS A; thus, internal events in a bisimilar A' are simulated in A by doing nothing, i.e. internal events never decide any choices and we are in fact working with the slightly stronger equivalence of branching bisimilarity, see [GW89].

A bisimulation *on* A is one between A and A, and states of A (or similarly reachable markings of a net) are *bisimilar* if they are related by a bisimulation on A.

For wATS, we also define ST-bisimulations and related notions:

- An *ST-state* (q, μ) of a wATS A consists of a state q and some q-step $\mu \subseteq E_v$. The *initial ST-state* is $(q_0, \emptyset)$.

- For a visible event a, we write $(q, \mu) \stackrel{a^+}{\rightarrow} (q, \mu \cup \{a\})$ if $q \stackrel{a}{\rightarrow}$ and a is independent to all $b \in \mu$ – which again implies $a \notin \mu$. Note that in the case of wATS, we cannot change the state component in a way that would somehow reflect just the starting of a; therefore, we have adapted the notation for nets accordingly in the previous section.

  We write $(q, \mu) \stackrel{a^-}{\rightarrow} (q', \mu \setminus \{a\})$ if $a \in \mu$ and $q \stackrel{a}{\rightarrow} q'$. Finally, for an internal event c, we write $(q, \mu) \stackrel{c}{\rightarrow} (q', \mu)$ if c is independent to all $b \in \mu$ and $q \stackrel{c}{\rightarrow} q'$. Note that in all three cases the pair reached is an ST-state again.

- We again extend this definition to sequences and, by suppressing internal events, to visible sequences.

Two wATSs A and A' with the same visible events are *ST-bisimilar*, if there exists an *ST-bisimulation* between them, i.e. a relation $\mathcal{B}$ between the ST-states of A and A' such that:

1. $\mathcal{B}$ relates the initial ST-states.

2. If $a \in E_v$, $c \in E_i$ and $(q, \mu)\mathcal{B}(q', \mu)$ (with the same μ), then we have:

   (a) $(q, \mu) \stackrel{a^+}{\rightarrow} (q, \mu \cup \{a\})$ implies that for some q'' $(q', \mu) \stackrel{a^+}{\Rightarrow} (q'', \mu \cup \{a\})$ and $(q, \mu \cup \{a\})\mathcal{B}(q'', \mu \cup \{a\})$

   (b) $(q, \mu) \stackrel{a^-}{\rightarrow} (q_1, \mu \setminus \{a\})$ implies that for some q_1' $(q', \mu) \stackrel{a^-}{\Rightarrow} (q_1', \mu \setminus \{a\})$ and $(q_1, \mu \setminus \{a\})\mathcal{B}(q_1', \mu \setminus \{a\})$

   (c) $(q, \mu) \stackrel{c}{\rightarrow} (q_1, \mu)$ implies that for some q_1' $(q', \mu) \Rightarrow (q_1', \mu)$ and $(q_1, \mu)\mathcal{B}(q_1', \mu)$

3. vice versa

In this paper, we define the *reachability graph* of a net N to be an ATS: its states are the reachable markings, it has the same visible and internal events as N, the transition relation is given by the firing rule, M_N is the initial state and the independence relation is $I(N)$ restricted to those (a, b) that are enabled under a common reachable marking. With this in mind, the above definition of ST-bisimulation extends the one from the

last section, since nets are ST-bisimilar if and only if their reachability graphs are ST-bisimilar. In this sense, we can also speak of a net being ST-bisimilar to an ATS or wATS.

Similarly, we call two nets *bismilar* if their reachability graphs are bismilar, and this way we can also speak of a net being bisimilar to an ATS or wATS.

History-preserving bisimulations, or hp-bisimulations for short, are usually defined for nets based on the partial orders induced by net processes. These partial orders can alternatively be obtained as Mazurkiewicz traces. Since the latter can be naturally defined for ATS as well, we define hp-bisimulation for ATS in this way; via the reachability graph, this also defines when two nets or a net and an ATS are *hp-bisimilar*.

For an ATS A and event sequences v and w, we write $v \sim w$ if there are independent events a and b such that $v = uabu'$ and $w = ubau'$. If v and w are related by the reflexive-transitive closure of $\sim$, we call them *equivalent*, write $[v]$ for the equivalence class of v and call it a *trace* – and a *trace of* A, if v is an occurrence sequence of A. Note that, due to the stronger independence requirement for ATS, all elements of a trace of A are occurrence sequences reaching the same state.

To each trace $[a_1 \dots a_n]$ we can associate a labelled partial order on $\{1, \dots, n\}$: each i is labelled with a_i and the order is the least transitive relation where i is 'less than' j if $i < j$ and a_i and a_j are dependent. $[a_1 \dots a_n]$ is exactly the set of linearizations of this labelled partial order, which is up to isomorphism independent of the representative $a_1 \dots a_n$. (Hence, strictly speaking, we consider labelled partial orders only up to isomorphism.) If we restrict the labelled partial order to the visible events, we obtain the *visible po* of $[a_1 \dots a_n]$.

Two ATSs A and A' with the same visible events are *hp-bisimilar*, if there exists an *hp-bisimulation* between them, i.e. a relation $\mathcal{B}$ between the traces of A and A' such that:

1. $[\lambda]\mathcal{B}[\lambda]$.
2. If $[v]\mathcal{B}[w]$, then $[v]$ and $[w]$ have the same visible po (up to isomorphism).
3. If $[v]\mathcal{B}[w]$ and $[va]$ is a trace of A for some event a, then there exists some trace $[wu]$, $u \in E'^*$, with $[va]\mathcal{B}[wu]$.
4. vice versa

Again, for general labelled nets, the elements of an hp-bisimulation are in fact triples where the additional component is an explicit isomorphism between the visible partial orders; again this is not necessary here where elements with the same label are always ordered such that the required isomorphism is unique.

4 ST-bisimilar implementations of weak ATSs

Assume a visible weak ATS $A = (Q, E_v, \emptyset, T, q_0, I)$ is given. In this section, we will construct a net N that is ST-bisimilar to A. First of all, we find a family of cliques covering the dependence graph of I, i.e. a family (D_i) of nonempty subsets of E_v such

that the elements of each D_i are pairwise dependent and such that for any dependent events a and b there exists a D_i containing a and b; in particular, the union of the D_i is E_v since possibly $a = b$. If one is not interested in concurrency, one can choose $I = \emptyset$ and E_v as the only D_i; Figure 1 shows a transition system and part of our net construction for this simple case, which yields a sequential net.

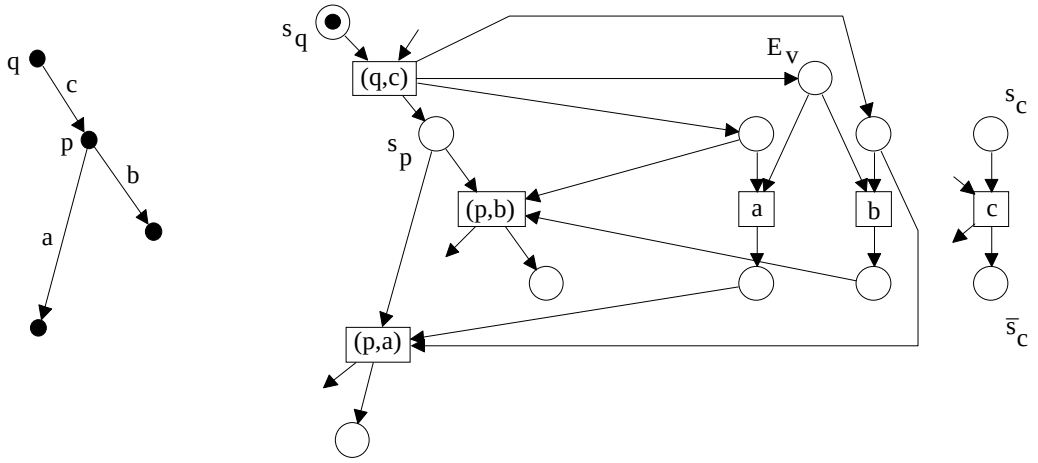


Figure 1

N has places s_q for $q \in Q$, s_a and $\bar{s}_a$ for $a \in E_v$, and D_i. It has the visible events in E_v, and $E_i = \{(q,a) \mid q \in Q,\ a \in E_v,\ q \xrightarrow{a}\}$. We define F by giving the pre- and postsets of the events. For $a \in E_v$, ${}^\bullet a = \{s_a\} \cup \{D_i \mid a \in D_i\}$ and $a^\bullet = \{\bar{s}_a\}$. For an internal event (q,a) with $q \xrightarrow{a} p$, we define:

$$\begin{aligned} {}^\bullet(q,a) &= \{s_q\} \cup \{\bar{s}_a\} \cup \{s_b \mid a \neq b \ \wedge\ q \xrightarrow{b} \ \wedge \neg p \xrightarrow{b}\} \\ &\quad \cup \{D_i \mid (\exists b \in D_i :\ q \xrightarrow{b}) \ \wedge\ (\neg\exists b \in D_i :\ p \xrightarrow{b}) \ \wedge\ a \notin D_i\} \end{aligned}$$

$$\begin{aligned} (q,a)^\bullet &= \{s_p\} \cup \{s_b \mid p \xrightarrow{b} \ \wedge (a = b \vee \neg q \xrightarrow{b})\} \\ &\quad \cup \{D_i \mid (\exists b \in D_i :\ p \xrightarrow{b}) \ \wedge\ (a \in D_i \vee \neg\exists b \in D_i :\ q \xrightarrow{b})\} \end{aligned}$$

It will turn out that the reachable markings have the form $M(q,\nu)$ where $\nu \subseteq E_v$ is a visible q-step; we define:

$$\begin{aligned} M(q,\nu) &= \{s_q\} \cup \{s_b \mid q \xrightarrow{b} \ \wedge\, b \notin \nu\} \cup \{\bar{s}_a \mid a \in \nu\} \\ &\quad \cup \{D_i \mid (\exists b \in D_i :\ q \xrightarrow{b}) \ \wedge\ D_i \cap \nu = \emptyset\} \end{aligned}$$

With this definition, the initial marking of N is $M(q_0, \emptyset)$.

Lemma 4.1 *N is safe and divergence-free, and the reachable markings of N are the markings $M(q,\nu)$ where $\nu \subseteq E_v$ is a visible q-step. More in detail:*

1. *$M(q,\nu)$ enables $a \in E_v$ iff $q \xrightarrow{a}$ and a is independent of each event in ν; then, $M(q,\nu)[a\rangle M(q,\nu \cup \{a\})$.*

2. *$M(q,\nu)$ enables an internal event iff it has the form (q,a) with $a \in \nu$ and $q \xrightarrow{a} p$ for some p; then, $M(q,\nu)[(q,a)\rangle M(p,\nu \setminus \{a\})$.*

Proof: First note that divergence-freeness will follow from statement 2. We will consider some $M(q,\nu)$ and show that firing any enabled transition does not violate safety and reaches again a marking of the desired form. Since the initial marking is $M(q_0,\emptyset)$, this shows that N is safe and the reachable markings of N are of the desired form. From our considerations, it will be easy to see that any $M(q,\nu)$ can be reached in N by taking a sequence reaching q in A, inserting after each a occurring from some p in A the internal transition (p,a) and finally adding the events in ν in some order.

If $a \in E_v$ is enabled under $M(q,\nu)$, then a needs a token from s_a, hence $q \xrightarrow{a}$, and a needs a token from all D_i, where $a \in D_i$; thus, all these D_i have an empty intersection with ν such that a is independent of each event in ν by choice of the D_i. Vice versa, each a with $q \xrightarrow{a}$ that is independent of each event in ν is easily seen to be enabled under $M(q,\nu)$. Firing such an a gives $M(q,\nu \cup \{a\})$ and does not violate safety.

An internal event enabled under $M(q,\nu)$ needs a token from s_q and from some $\bar{s}_a$, and thus must have the form (q,a) with $a \in \nu$ and $q \xrightarrow{a} p$ for some p. By considering the different types of places separately, we will show that each such (q,a) is in fact enabled and that firing it does not violate safety and reaches $M(p,\nu \setminus \{a\})$.

Firing (q,a) as above removes $\bar{s}_a$ and replaces s_q by s_p observing safety.

For the places s_b, $b \in E_v$, observe that $b \in \nu$ implies $a = b \vee p \xrightarrow{b}$. Now, $s_b \in {}^\bullet(q,a)$ implies $a \neq b \wedge \neg p \xrightarrow{b}$, hence $b \notin \nu$; since also $q \xrightarrow{b}$, we see that $M(q,\nu)$ marks $s_b \in {}^\bullet(q,a)$ and (q,a) is enabled w.r.t. the s_b. If $M(q,\nu)$ would mark some $s_b \in (q,a)^\bullet$, then $q \xrightarrow{b}$ and thus $b = a \in \nu$, a contradiction. Hence, firing (q,a) does not violate safety w.r.t. the s_b and s_b is marked after firing (q,a) iff $s_b \in (q,a)^\bullet$ or $s_b \notin {}^\bullet(q,a) \wedge s_b \in M(q,\nu)$. The latter disjunct means $(a = b \vee \neg q \xrightarrow{b} \vee\ p \xrightarrow{b}) \wedge q \xrightarrow{b} \wedge\ b \notin \nu$, i.e. $p \xrightarrow{b} \wedge\ q \xrightarrow{b} \wedge\ b \notin \nu$, since $a = b$ contradicts $b \notin \nu$. Expanding $s_b \in (q,a)^\bullet$, we get that s_b is marked after firing (q,a) iff $p \xrightarrow{b}$ and $a = b \vee \neg q \xrightarrow{b} \vee\ (q \xrightarrow{b} \wedge\ b \notin \nu)$. Since $b \in \nu$ implies $q \xrightarrow{b}$, the latter conjunct means $a = b \vee b \notin \nu$, i.e. $b \notin \nu \setminus \{a\}$. Thus, s_b is marked after firing (q,a) iff $p \xrightarrow{b}$ and $b \notin \nu \setminus \{a\}$ iff $M(p,\nu \setminus \{a\}$ marks s_b.

It remains to check the D_i, so let us fix one of them; we will write $A(q)$ for $\exists b \in D_i :\ q \xrightarrow{b}$ and analogously $A(p)$. Furthermore, B will stand for $a \in D_i$ and C for $D_i \cap (\nu \setminus \{a\}) = \emptyset$. With this, we have

$D_i \in {}^\bullet(q,a)$ iff $A(q)\ \wedge\ \neg A(p)\ \wedge\ \neg B$,
$D_i \in (q,a)^\bullet$ iff $A(p)\ \wedge\ (B \vee \neg A(q))$,
$D_i \in M(q,\nu)$ iff $A(q)\ \wedge\ \neg B\ \wedge\ C$,
$D_i \in M(p,\nu \setminus \{a\}$ iff $A(p)\ \wedge\ C$.

We list a number of properties:

i) $\neg A(p)$ implies C. If $\neg C$, pick some $b \in D_i \cap (\nu \setminus \{a\})$; then aIb and $q \xrightarrow{b}$, thus $q \xrightarrow{a} p$ implies $p \xrightarrow{b}$ and, hence, $A(p)$.

ii) $D_i \in {}^\bullet(q,a)$ implies $D_i \in M(q,\nu)$ (by i)). Hence, (q,a) is enabled w.r.t. the D_i.

iii) $D_i \in M(q,\nu)$ implies $D_i \notin (q,a)^\bullet$ (because of $A(q) \wedge \neg B$). Hence, firing (q,a) does not violate safety w.r.t. the D_i.

iv) B implies C. (a is independent to all other events in ν.)

v) $\neg A(q)$ implies C, since $b \in D_i \cap \nu$ implies $q \xrightarrow{b}$.

Now D_i is marked after firing (q,a) iff $D_i \in (q,a)^\bullet$ or $D_i \notin {}^\bullet(q,a) \wedge D_i \in M(q,\nu)$. The latter disjunct means $(\neg A(q) \vee A(p) \vee B) \wedge A(q) \wedge \neg B \wedge C$, which can be simplified to $A(p) \wedge A(q) \wedge \neg B \wedge C$. Expanding $D_i \in (q,a)^\bullet$, we get that D_i is marked after firing (q,a) iff $A(p)$ and $B \vee \neg A(q) \vee (A(q) \wedge \neg B \wedge C)$. The latter conjunct can be simplified to $B \vee \neg A(q) \vee C$ and by iv) and v) to C. Thus, D_i is marked after firing (q,a) iff $A(p)$ and C iff $M(p, \nu \setminus \{a\}$ marks D_i. □

Now we will show that $\mathcal{B}$ is an ST-bisimulation between A and N, where $\mathcal{B}$ relates (q',μ) to $(M(q,\nu),\mu)$ if $\mu \cup \nu$ is a visible q-step, $\mu \cap \nu = \emptyset$ and from q the step ν reaches q' (which then allows the step μ). This is clearly satisfied for $(q_0,\emptyset)$ and $(M(q_0,\emptyset),\emptyset)$, so let us assume $(q',\mu)\mathcal{B}(M(q,\nu),\mu)$.

We will first show how A simulates N, so consider $(M(q,\nu),\mu)[a^+\rangle(M(q,\nu),\mu \cup \{a\})$, hence a is enabled under $M(q,\nu)$ and independent to all events in μ. a is also independent to all events in ν, since otherwise some $D_i \in {}^\bullet a$ would be missing in $M(q,\nu)$ – in particular, $a \notin \nu$. Since s_a must be marked under $M(q,\nu)$, we get $q \xrightarrow{a}$; thus, $\mu \cup \nu \cup \{a\}$ is a visible q-step and $\mu \cup \{a\}$ is a visible q'-step; in particular, $(q',\mu) \xrightarrow{a^+} (q',\mu \cup \{a\})$. Furthermore, $(q',\mu \cup \{a\})\mathcal{B}(M(q,\nu),\mu \cup \{a\})$.

Next, consider $(M(q,\nu),\mu)[a^-\rangle(M(q,\nu \cup \{a\}),\mu \setminus \{a\})$ – see Lemma 4.1. Since q' enables $a \in \mu$, take p with $q' \xrightarrow{a} p$, i.e. $(q',\mu) \xrightarrow{a^-} (p,\mu\setminus\{a\})$. Now $(\nu\cup\{a\})\cup(\mu\setminus\{a\}) = \mu\cup\nu$ is a visible q-step, from q the step $\nu\cup\{a\}$ reaches p and $(\nu\cup\{a\})\cap(\mu\setminus\{a\}) = \emptyset$; thus, $(p,\mu \setminus \{a\})\mathcal{B}(M(q,\nu \cup \{a\}),\mu \setminus \{a\})$.

To conclude with this simulation, consider $(M(q,\nu),\mu)[(q,a)\rangle(M(p,\nu \setminus \{a\}),\mu)$ where $a \in \nu$ and $q \xrightarrow{a} p$ – see Lemma 4.1. Since $(q',\mu)\mathcal{B}(M(q,\nu),\mu)$, $\mu \cup \nu \setminus \{a\}$ is a visible p-step, where $\mu \cap (\nu \setminus \{a\}) \subseteq \mu \cap \nu = \emptyset$, and from p the step $\nu \setminus \{a\}$ reaches q'; thus, $(q',\mu)\mathcal{B}(M(p,\nu \setminus \{a\}),\mu)$.

Now, we show how N simulates A; given $(q',\mu)\mathcal{B}(M(q,\nu),\mu)$, we can conclude from the last paragraph that N can first fire internal events reaching $(M(q',\emptyset),\mu)$, which is $\mathcal{B}$-related to (q',μ) as well. For this, we have to show that (q,a) with $a \in \nu$ and $q \xrightarrow{a} p$ for some p is independent of the events in μ, so take some $b \in \mu$, i.e. $a \neq b$. The places in ${}^\bullet b \cup b^\bullet$ are $\bar{s}_b$, which is not in ${}^\bullet(q,a) \cup (q,a)^\bullet$, s_b and each D_i with $b \in D_i$. Since $\mu \cup \nu$ is a q-step, we have $q \xrightarrow{b}$ and $p \xrightarrow{b}$; thus, s_b is not in ${}^\bullet(q,a) \cup (q,a)^\bullet$, either. Since $p \xrightarrow{b}$, a D_i containing b is not in ${}^\bullet(q,a)$. If such a D_i were in $(q,a)^\bullet$, $q \xrightarrow{b}$ would imply $a \in D_i$, but a and b are independent. Thus, we have shown the independence of (q,a) and b.

Hence, we only have to consider the case $(q',\mu)\mathcal{B}(M(q',\emptyset),\mu)$. On the one hand, if $(q',\mu) \xrightarrow{a^+} (q',\mu \cup \{a\})$, then $a \notin \mu$ and $\mu \cup \{a\}$ is a q'-step, i.e. $q' \xrightarrow{a}$ and a is independent of the events in μ. With Lemma 4.1 we get that $M(q',\emptyset)$ enables a, thus $(M(q',\emptyset),\mu)[a^+\rangle(M(q',\emptyset),\mu \cup \{a\})$ and the latter is $\mathcal{B}$-related to $(q',\mu \cup \{a\})$.

On the other hand, if $(q',\mu) \xrightarrow{a^-} (p,\mu \setminus \{a\})$ with $a \in \mu$ and $q' \xrightarrow{a} p$, we have

$M(q',\emptyset)[a\rangle M(q',\{a\})$ by Lemma 4.1. Hence, $(M(q',\emptyset),\mu)[a^-\rangle(M(q',\{a\}),\mu\setminus\{a\})$ and the latter ST-marking is $\mathcal{B}$-related to $(p,\mu\setminus\{a\})$.

Thus we have shown:

Theorem 4.2 *For each visible weak ATS A there exists a divergence-free net N that is ST-bisimilar to A.*

Note that N constructed from A above contains a loop if and only if A contains a loop. Such a loop in N consists of (q,a) and s_q with $q \xrightarrow{a} q$.

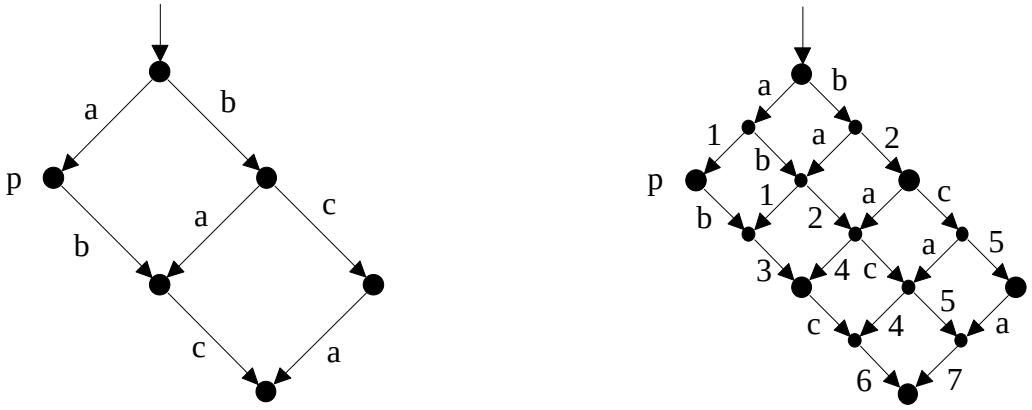


Figure 2

Figure 2 shows an ATS, where a is independent of b and c, and the reachability graph of the net that results from our construction, where the internal events have simply been numbered and $1 = (q_0,a)$, $2 = (q_0,b)$ and $3 = (p,b)$. This example demonstrates the transformation of the given ATS that is implied by our construction; one could call it τ-splitting of events where τ stands for an internal event. But it also makes clear that this construction does not always give a 'good' result: the occurrence sequence $ab13c$ gives a trace where, in the visible po, c comes 'after', i.e. depends causally on, both a and b. Of course, it is straightforward to find a net N with the given ATS as reachability graph, such that in N c is completely independent of a.

But note that the construction of this section works for any *weak* asynchronous transition system; indeed, Figure 3 shows a transition system and a net implementation (not obtained by our construction) where a and b are independent and state q satisfies $q \xrightarrow{ab}$ but not $q \xrightarrow{ba}$. This is only possible when using internal events in the net.

One could think of an even more general specification of independence where events a and b could be independent under some state q, i.e. $q \xrightarrow{ab} q'$ and $q \xrightarrow{ba} q'$ for some q', but dependent under some other state p, indicated e.g. by $p \xrightarrow{a}$ and $p \xrightarrow{b}$ but not $p \xrightarrow{ab}$. Such transition systems cannot be sensibly implemented by safe nets: we would need a reachable marking bisimilar to p; firing internal events from this marking, we would reach a stable marking M (i.e. one not enabling an internal event) by divergence-freeness, and M would also be bisimilar to p, since the transition system does not have internal events and thus internal events of the net do not make any

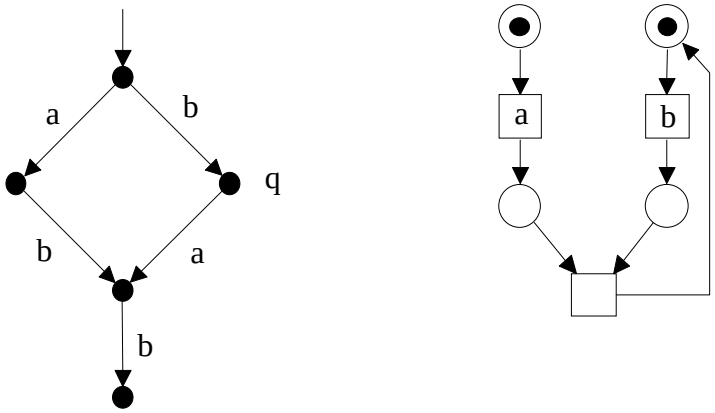


Figure 3

choices. Since a and b, but not ab would be enabled under M, a and b would have a common place in their presets and could never be independent.

5 Safe and semi-safe transition systems

For the next section, we need the extension of the classical theory of regions from [PKY95, DS93] and a variation of the classical theory given in [CKLY98]. [PKY95] deals with safe nets instead of elementary ones, but (in contrast to [DS93]) forbids transitions that are only on loops, i.e. have coinciding pre- and postset. Correspondingly, [PKY95] only considers transition systems without loops, which is usual and also applies to [CKLY98]. Since we will consider the problem of realizing a visible transition system with a bisimilar visible net, and since transition systems without loops can be bisimilar to transition systems with loops, this restriction does not seem natural in our context. Hence, we take the opportunity to formulate the theory of regions and its variation according to [CKLY98] for general safe nets, where the first part simply means to spell out a special case of the results in [DS93].

In this section, all events are visible, i.e. the additional feature of internal events is of no importance and we are in the more usual setting. Also, our results do not depend on independence; independence is just an additional feature that makes Theorem 5.4 below stronger using Proposition 5.3.

For a visible ATS A, a *region* is a nonempty, proper subset R of Q satisfying for each event a one of the following cases (where the third is a subcase of the fourth):

- R is a *pre-region* of a, $R \in {}^\circ a$, i.e. $q \xrightarrow{a} p$ implies $q \in R$ and $p \notin R$.
- R is a *post-region* of a, $R \in a^\circ$, i.e. $q \xrightarrow{a} p$ implies $q \notin R$ and $p \in R$.
- R is a *co-region* of a, $R \in \overset{\circ}{a}$, i.e. $q \xrightarrow{a} p$ implies $q \in R$ and $p \in R$.
- a is *not crossing* R, i.e. $q \xrightarrow{a} p$ implies $q, p \in R$ or $q, p \notin R$.

A visible ATS is *safe*, if it satisfies the following two properties:

- *event separation*: for all $a \in E$ and $q \in Q$, $\neg q \xrightarrow{a}$ implies that there exists a region $R \in {}^\circ a \cup \overset{\circ}{a}$ with $q \notin R$.
- *state separation*: for all $p, q \in Q$ with $p \neq q$, there exists a region R such that $p \in R$ iff $q \notin R$.

We will show that safe ATS are (up to isomorphism) just the reachability graphs of general safe nets, where one implication is quite obvious.

Proposition 5.1 *If N is a visible net, then its reachability graph is a safe ATS.*

Proof: Let s be a place of N; then the reachable markings of N that contain s form a region R – provided this set is nonempty and does not contain all reachable markings: if $s \in {}^\bullet a \cap a^\bullet$, then R is a co-region; otherwise, if $s \in {}^\bullet a$, then R is a pre-region, if $s \in a^\bullet$, then R is a post-region; finally, if $s \notin {}^\bullet a \cup a^\bullet$, a is not crossing R.

If a reachable marking M does not enable a, then there is some $s \in {}^\bullet a$ with $s \notin M$; since a is enabled under some reachable marking, which thus contains s, the set R associated to s is indeed a region and does not contain M.

For different reachable markings, there exists a place s in one not contained in the other. The associated R is again a region and contains the first, but not the latter marking.

The independence requirements are obvious. □

For the other implication, we first note:

Lemma 5.2 *If A is a safe ATS and a an event that is enabled under all states, then $q \xrightarrow{a} q$ for all states q.*

Proof: Otherwise, take different p and q with $p \xrightarrow{a} q \xrightarrow{a}$; one sees easily that any region containing p or q is not a pre- or postregion of a, hence does not separate p and q; thus, p and q violate state separation. □

We call an event a with $q \xrightarrow{a} q$ for all states q a *loop-event*.

Now assume we are given a safe ATS A; let $\mathcal{R}$ be a set of regions that are sufficient to satisfy event and state separation. We construct a net $N = (\mathcal{R} \cup \{s\}, E_v, \emptyset, F, M_N)$ by letting $(R, a) \in F$ if $R \in {}^\circ a \cup \overset{\circ}{a}$, $(a, R) \in F$ if $R \in a^\circ \cup \overset{\circ}{a}$, $(a, s), (s, a) \in F$ if a is a loop-event, and finally $M_N = \{R \in \mathcal{R} \mid q_0 \in R\} \cup \{s\}$. First note, that each event a has a nonempty preset in N since it is either a loop-event or it has a region R with $(R, a) \in F$ by event separation.

We now show that the reachability graph of N is isomorphic to A except for the independence relation, where $q \in Q$ corresponds to $\{R \in \mathcal{R} \mid q \in R\} \cup \{s\}$, which is injective by state separation of $\mathcal{R}$ and obviously preserves initial states.

We will show that this correspondence preserves the transition relation, too, and also check that safety is not violated in N. Since all states are reachable, this shows at the same time that our correspondence is a bijection onto the reachable markings. Observe that a loop-event has only s in its pre- and postset, which is always marked; thus, we can ignore s and any loop-events in the following. Now, take corresponding q and M and first assume $q \xrightarrow{a} q'$. If $(R, a) \in F$, then $q \in R$ and R is marked under

M; thus, a is enabled under M; we define $M' = (M \setminus {}^\bullet a) \cup a^\bullet$. We check the different possibilities for a region $R \in \mathcal{R}$ w.r.t. a. If $R \in {}^\circ a$, then on the one hand $q \in R$ and $q' \notin R$, while on the other hand firing a empties R; hence $R \notin M'$. If $R \in a^\circ$, then on the one hand $q \notin R$ and $q' \in R$, while on the other hand firing a marks R; hence $R \notin M$, such that safety is not violated, and $R \in M'$. If $R \in \overset{\circ}{a}$, then on the one hand $q \in R$ and $q' \in R$, while on the other hand a and R form a loop; hence safety is not violated and $R \in M'$. If none of these cases applies, then on the one hand $q \in R$ iff $q' \in R$, while on the other hand firing a does not change the marking of R; hence, $R \in M$ iff $R \in M'$. In any case, q' and M' correspond (using the correspondence of q and M in the last case).

Second, assume $M[a\rangle M'$. If $\neg q \xrightarrow{a}$, then there is some $R \in \mathcal{R}$ with $q \notin R$ and $R \in {}^\circ a \cup \overset{\circ}{a}$; this implies $R \notin M$ and $(R, a) \in F$, a contradiction. Hence, $q \xrightarrow{a} q'$ for some q', which by the last paragraph corresponds to M'.

If we are not interested in the independence relation, N realizes A. Otherwise, note that our correspondence is an isomorphism (up to the independence relation) and hence also a bisimulation, and apply the following new result.

Proposition 5.3 *Let A be a visible ATS and N a visible net bisimilar to A. Then there exists a net N' whose reachability graph is isomorphic to that of N except that its independence is I (i.e. that of A).*

Proof: We might have events a and b that are independent according to $I(N)$ but not according to I. In this case, we can add a common marked loop place to a and b. This does not really change the reachability graph of N and makes a and b dependent.

We also could have events a and b that are independent according to I but not according to $I(N)$. Consider some $q_1 \in Q$ with $q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_4$ and $q_1 \xrightarrow{b} q_3 \xrightarrow{a} q_4$. Due to bisimilarity, there is a reachable marking M_1 of N with $M_1[a\rangle M_2[b\rangle M_4$ and $M_1[b\rangle M_3[a\rangle M_4'$. The effect of firing a and b does not depend on their order; hence $M_4 = M_4'$. Furthermore, a place in $({}^\bullet a \cup a^\bullet) \cap ({}^\bullet b \cup b^\bullet)$ can only be a common loop place of a and b. In this situation, we duplicate each such common loop place s such that both copies are connected by arcs to all other events in the same way as s, but one copy is on a loop with a (and not with b) while the other is on a loop with b (and not with a). This does not really change the reachability graph of N and makes a and b independent. □

We conclude:

Theorem 5.4 *If A is a safe ATS, then there effectively exists a (visible) net N whose reachability graph is isomorphic to A.*

As a next step, we weaken the definition of a safe ATS. A visible ATS is *semi-safe* if it satisfies event separation, but not necessarily state separation. Thus, a semi-safe ATS is up to the treatment of loops and up to independence what [CKLY98] calls an excitation-closed transition system. Following [CKLY98], we will show that a semi-safe ATS A can be realized by a visible net up to bisimilarity by transforming A to a bisimilar safe ATS and applying the above theorem. Semi-safety on languages, i.e. on infinite tree-shaped transition systems is also considered in [BBD95] and in the context of trace languages in [HKT92]. We note a lemma first.

Lemma 5.5 *Let A be a visible ATS, $\mathcal{B}$ an equivalence on Q that is also a bisimulation on A, and denote the equivalence class of $q \in Q$ by $[q]$. Then the $\mathcal{B}$-quotient $A' = (\{[q] \mid q \in Q\}, E_v, \emptyset, T', [q_0], I)$ is a (visible) ATS bisimilar to A, where $T'([q], a) = [q']$ if $q \xrightarrow{a} q'$.*

Proof: This is easy and well-known for non-deterministic transition systems; the proof involves the fact that $p \in [q]$ and $q \xrightarrow{a} q'$ implies $p \xrightarrow{a} p'$ for some $p' \in [q']$. Due to determinism, this shows that T' is indeed a partial function; also, the independence requirements follow easily. □

Now assume a semi-safe ATS A is given. Define a relation $\mathcal{B}$ on Q by $q\mathcal{B}p$ if q and p are contained in the same regions. Clearly, $\mathcal{B}$ is an equivalence.

Let $q \xrightarrow{a} q'$ and $q\mathcal{B}p$; if $\neg p \xrightarrow{a}$, then due to event separation there would be a pre-region of a – hence containing q – not containing p, a contradiction. Thus, there is some p' with $p \xrightarrow{a} p'$. If a region R contains q' and q, hence p, then a is not crossing R, i.e. $p' \in R$. If R contains q' but neither q nor p, then R is a post-region of a and contains p'. Vice versa, each region containing p' contains q', too; thus, $q'\mathcal{B}p'$. Therefore, $\mathcal{B}$ is a bisimulation on A. By the above lemma, A' as defined there is an ATS bisimilar to A.

It remains to show that in our case A' is a safe ATS. For a region R of A, we define $[R] = \{[q] \mid q \in R\}$. If R is a region of A and $q \in R$, then clearly $[q] \subseteq R$. With this and the above considerations, it is not hard to see that $[R]$ is a region of A' and, more precisely, a pre-, co- or post-region for some a if R is a pre-, co- or post-region for this a in A. Thus, if some $[q]$ does not enable some a in A', then neither does q in A and there is some pre- or co-region R of a not containing q; hence, $[R]$ is a pre- or co-region of a not containing $[q]$. If $[q] \neq [p]$, then $\neg q\mathcal{B}p$ and there is some region R containing exactly one of p and q, thus $[R]$ is a region containing exactly one of $[p]$ and $[q]$.

We conclude:

Theorem 5.6 *If A is a semi-safe ATS, then there effectively exists a bisimilar safe ATS and a visible net N bisimilar to A.*

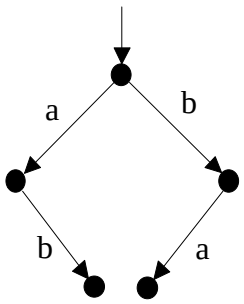


Figure 4

Figure 4 shows a semi-safe ATS (with dependent a and b). Clearly, it cannot be the reachability graph of a net, but identification of the two 'terminal' states gives such an ATS.

6 Results on bisimilar and hp-bisimilar implementations of ATSs

Largely repeating from the literature, we have seen in the last section that semi-safe ATSs can be implemented as visible nets up to bisimilarity. We started out with the aim to use behaviour notions that also consider concurrency. With the following easy lemma, it becomes obvious from Proposition 5.3 that, whenever we can realize a visible ATS by a bisimilar visible net, we can also realize it by a hp-bisimilar visible net; this applies in particular to semi-safe ATSs.

Lemma 6.1 *If two bisimilar visible ATSs have the same independence relation, then they are hp-bisimilar.*

Proof: By bisimilarity, the two visible ATSs have the same occurrence sequences, hence the same traces, and the identity is an hp-bisimulation. □

Corollary 6.2 *Let A be a visible ATS and N a visible net bisimilar to A. Then there also exists a visible net hp-bisimilar to A. In particular, if A is a semi-safe ATS, then there exists a visible net hp-bisimilar to A.*

While we have seen in the last section that identifying bisimilar states in an ATS can help to find a net implementation, Figure 5 shows an ATS that violates event separation for d and q; since there are no bisimilar states, identification of bisimilar states cannot help here. Nevertheless, the ATS has a bisimilar net implementation as shown where the two occurrences of c lead to different markings. So splitting a state into bisimilar copies can also help.

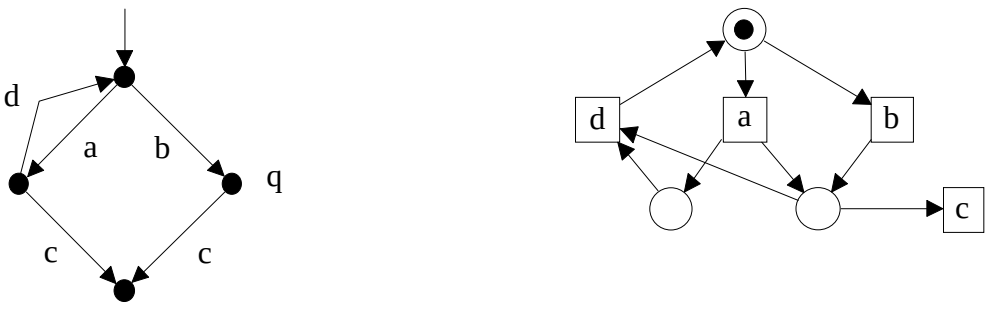


Figure 5

While semi-safety is sufficient to ensure that a bisimilar visible net exists, we will now describe an effective algorithm to decide whether to a given visible ATS there exists a bisimilar (or hp-bisimilar) visible net. We list some easy lemmata first.

Lemma 6.3 *Let A' and A be ATSs and h a morphism from A' to A, i.e. a function from Q' to Q with $h(q'_0) = q_0$ such that $p \xrightarrow{a} q$ in A' implies $h(p) \xrightarrow{a} h(q)$ in A. If R is a region of A (and a pre-/post-/co-region of some a), then $h^{-1}(R)$ is a region of A' (and a pre-/post-/co-region of a).*

Proof: Let R be a pre-region of a and $p \xrightarrow{a} q$ in A'. Then $h(p) \xrightarrow{a} h(q)$, $h(p) \in R$ and $h(q) \notin R$ and hence $p \in h^{-1}(R)$ and $q \notin h^{-1}(R)$. The other cases are similar. □

Lemma 6.4 *Let A and A' be bisimilar visible ATSs and define $q\mathcal{B}q'$ if $q_0 \xrightarrow{w} q$ in A and $q'_0 \xrightarrow{w} q'$ in A' for some $w \in E^*$. Then $\mathcal{B}$ is a bisimulation.*

Proof: Clearly, $q_0\mathcal{B}q'_0$. If $q\mathcal{B}q'$, then by induction on the related w each bisimulation between A and A' must relate q and q' due to determinism. Thus, if $q \xrightarrow{a} p$ then $q' \xrightarrow{a} p'$ for some p' and $p\mathcal{B}p'$ by definition of $\mathcal{B}$. □

Lemma 6.5 *Let N be a visible net and M, M' be reachable markings with $M[w\rangle M'$ for some $w \in E^*$. If $M'[w\rangle$ or M and M' are bisimilar, then $M = M'$.*

Proof: By safety, $M[w\rangle M'[w\rangle$ implies that firing w cannot change a marking. If M and M' are bisimilar, then $M'[w\rangle$. □

Let A be an ATS. States p and q of A are *strongly connected*, if there are paths from p to q and from q to p (i.e. $p \xrightarrow{v} q$ and $q \xrightarrow{w} p$ for some $v, w \in E^*$). Being strongly connected is an equivalence relation; a *strongly connected component* (*scc* for short) consists of an equivalence class together with the arcs between any two of its elements. An arc $p \xrightarrow{a} q$ is a *tree-edge*, if it does not belong to any *scc*. Note that no path in A can use a tree-edge twice, because then we would have a cycle containing the tree-edge, and this cycle would be contained in a *scc*.

We will now define the *scc-tree* of A, denoted *scc-tree*(A). The idea is to unfold A into a tree-like ATS, where the *scc*s are left intact but possibly get duplicated, and form a tree with the tree-edges. This unfolding gives a finite ATS (in contrast with the complete unfolding into a tree), but all states of A are split as much as it could be helpful for finding a suitable net.

Let $q_0 \xrightarrow{a_1} q_1$, $q_1 \xrightarrow{a_2} q_2$, ..., $q_{n-1} \xrightarrow{a_n} q_n$ in A; taking the subsequence of tree-edges and representing each such tree-edge $q \xrightarrow{a} p$ as (q, a, p), we obtain a sequence σ which we call a *tree-path to q_n*. Note that σ cannot contain a repetition, and thus there can only be finitely many tree-paths to any q.

We define *scc-tree*$(A) =: A'$ as follows: $Q' = \{(q, \sigma) \mid \sigma \text{ a tree-path to } q \in Q\}$, $E'_v = E_v$, $E'_i = E_i$, $q'_0 = (q_0, \lambda)$ and $I' = \emptyset$; $T'((q, \sigma), a)$ is (p, σ) if $q \xrightarrow{a} p$ and q and p belong to the same *scc*, and it is $(p, \sigma(q, a, p))$ if $q \xrightarrow{a} p$ is a tree-edge. Observe that both images are indeed in Q' and that the transition relation is deterministic as required.

Occurrence of an event changes the first component of a state of A' as in A, while the other component gives just additional information splitting q into several copies. Hence, it is clear that – just as in A – each event is enabled under some state of A' and that A and A' are bisimilar; since the independence relation is empty, A' is an ATS.

Lemma 6.6 *scc-tree(A) is an ATS and bisimilar to A.*

The following theorem shows that by constructing *scc-tree*(A) we have performed all state splittings that can possibly help to implement A.

Theorem 6.7 *There exists a visible net N (hp-)bisimilar to a visible ATS A if and only if scc-tree(A) is a semi-safe ATS.*

Proof: To see the reverse implication, apply Theorem 5.6, Lemma 6.6 (and Corollary 6.2).

For the other implication, *scc-tree*(A) and N are bisimilar by Lemma 6.6, and the relation $\mathcal{B}$ in Lemma 6.4 relates each state (q,σ) to some reachable marking M; we show that this M is unique, i.e. $\mathcal{B}$ is a function.

If $\sigma = (q_1, a_1, q_1') \dots (q_k, a_k, q_k')$, then each occurrence sequence to (q,σ) must use each arc from $(q_i, (q_1, a_1, q_1') \dots (q_{i-1}, a_{i-1}, q_{i-1}'))$ to $(q_i', (q_1, a_1, q_1') \dots (q_i, a_i, q_i'))$ for $i = 1, \dots, k$. If (q,σ) is related to M and M' due to some w and w', then both of the latter must use these arcs. Thus, we can apply induction if we show:

$(*)$ Assume that $(q_0,\lambda) \stackrel{w}{\to} (q,\sigma) \stackrel{w'}{\to} (q',\sigma)$ and $(q,\sigma) \stackrel{w''}{\to} (q',\sigma)$ in *scc-tree*(A) and that $M_N[w\rangle M[w'\rangle M'$ and $M[w''\rangle M''$ in N; then $M' = M''$.

Proof of $(*)$: Since q and q' belong to the same *scc*, there is some v with $q' \stackrel{v}{\to} q$, i.e. $(q',\sigma) \stackrel{v}{\to} (q,\sigma)$. Since $\mathcal{B}$ is a bisimulation, this shows $M[w'\rangle M'[v\rangle M'''$ for some M''' bisimilar to (q,σ), which implies $M'''[w'v\rangle$ and with Lemma 6.5 $M = M'''$. Thus $M'[vw''\rangle M''$.

Since M' and M'' are related to the same (q',σ) by the bisimulation $\mathcal{B}$, they are bisimilar and hence equal by Lemma 6.5 again. $\Box(*)$

Thus, $\mathcal{B}$ is a function h and in fact a morphism as defined in Lemma 6.3, which gives us event separation: If $\neg(q,\sigma) \stackrel{a}{\to}$ in *scc-tree*(A), then $\neg h(q,\sigma)[a\rangle$ in N, since h is a bisimulation. Hence, there is some pre- or co-region R of a in the reachability graph of N with $h(q,\sigma) \notin R$. Thus, $h^{-1}(R) \in {}^\circ a \cup \overset{\circ}{a}$ in *scc-tree*(A) with $(q,\sigma) \notin h^{-1}(R)$. We conclude that *scc-tree*(A) is semi-safe. $\Box$

Without internal events, finding a bisimilar net is the same problem as finding a language equivalent net. This problem has been considered for unbounded nets in [HKT92] (in a trace setting), for bounded nets (without loops) in [BBD95] and, very recently, for unbounded nets (without loops) in [Dar98]. It is shown that the language, i.e. the complete unfolding of the transition system, has to satisfy event separation. So the important point of our result is that *scc-tree*(A) is finite. [BBD95, Dar98] give effective results, where [BBD95] works on regular expressions, while [Dar98] uses a finite tree-like unfolding of the transition system; this unfolding seems to be much more complicated than ours – where of course the problem considered is different and unbounded nets are more involved than safe ones.

There can be exponentially many tree-paths in a visible ATS A, e.g. if it consists of a sequence of states each being connected to the next by *two* arcs. On the other hand, if A has n states, there are at most n^n tree-paths, so *scc-tree*(A) can be at most of exponential size compared to A.

To make A and, thus, *scc-tree*(A) smaller, one can first of all replace A by its bisimilarity-quotient A'. Since bisimilarity on A is a bisimulation and an equivalence, A' is a visible ATS bisimilar to A by Lemma 5.5, so the existence of a bisimilar net can be checked for A' instead of A.

In fact, I assume that in practice the simplicity of the *scc-tree* will help a lot. In many cases, A or at least its bisimilarity-quotient A' will be strongly connected or

consist of a path from the initial state to some *scc* (which is then the only one which may contain more than one state). Then A' is isomorphic to *scc-tree*(A'), and we simply have to check whether A' is semi-safe. Since there are no bisimilar states in A', this is the same as checking safety by the proof of Theorem 5.6. Of course, semi-safety is most likely easier to check than safety; at least in this case, it is sufficient to find enough regions to guarantee event-separation – also for the construction of a suitable net. We summarize:

Corollary 6.8 *Let A be a visible ATS and A' be its bisimilarity-quotient.*

i) If A is strongly connected or consists of a path from the initial state to some scc, the same applies to A'. If A' is strongly connected or consists of a path from the initial state to some scc, then scc-tree(A') is isomorphic to A'.

ii) Assume scc-tree(A') is isomorphic to A'. Then there exists a visible net N (hp-)bisimilar to A if and only if A' is safe if and only if A' is semi-safe, where N can directly be constructed from some regions guaranteeing event-separation.

In the cases not covered by this corollary, there are possibilities for further improvements. First, when minimizing A one possibly merges states that are then split again in *scc-tree*(A); one could try to avoid this, but this would need careful consideration: e.g. just merging bisimilar states that belong to the same *scc* could create non-determinism, hence further merging could be required. Second, if there is some 'diamond' – i.e. $q \xrightarrow{ab} p$ and $q \xrightarrow{ba} p$ – and some arc involved is a tree-edge, then the diamond would be split in *scc-tree*(A); this cannot help to find a net, since in a net firing ab or ba leads to the same marking; thus, splitting in the construction of *scc-tree*(A) could be reduced such that diamonds are kept intact.

We now come to the last contribution of this section: we will exhibit a construction on ATSs that preserves hp-bisimilarity and involves internal events. The example treated in [Yak98] demonstrates that this can turn a visible ATS that is not semi-safe into one that is safe if one regards the additional internal event as visible. This shows how to use our construction: the new ATS is isomorphic to the reachability graph of a net N which is therefore hp-bisimilar to the original ATS if we regard the additional event of N as internal again. (In formal words, this is true because hp-bisimilarity is a congruence for hiding.) Later, we will give an example showing that the construction is not always helpful. Thus, for the time being, we can only offer a trial-and-error method that may help to find a hp-bisimilar net.

Let A be an ATS; a *τ-state-splitting* A' of A is obtained by choosing a state q and a family $\{q_a \xrightarrow{a} q\}$ of arcs such that a and b are dependent, whenever $q_a \xrightarrow{a} q$ is in this family and we have $p \xrightarrow{b} q$ outside the family or $q \xrightarrow{b} p$ for some state p. Then, A' has a new state q', a new internal event τ dependent to all other events, and a new arc $q' \xrightarrow{\tau} q$; each arc $q_a \xrightarrow{a} q$ of the family is redefined to $q_a \xrightarrow{a} q'$.

Theorem 6.9 *Let A be an ATS and A' be a τ-state-splitting of A. Then A and A' are hp-bisimilar and one is divergence-free if the other one is.*

Proof: The claim about divergence-freeness should be clear. We will show the theorem for a visible ATS A; this is sufficient, since turning visible events into internal ones preserves hp-bisimilarity. Assume we are given q and the family of arcs as above.

First, we have to show that A' is an ATS, i.e. satisfies the independence requirements. So assume that in A' $p \xrightarrow{a} p_1$ and for some b independent of a $p \xrightarrow{b}$ or $p_1 \xrightarrow{b}$. Clearly, $p \neq q'$, since τ is not independent of any event. If $p_1 = q'$, we would have $p \xrightarrow{a} q$ and $p \xrightarrow{b}$ in A, hence $q \xrightarrow{a}$ contradicting the choice of the arc family. Thus, we have a diamond $p \xrightarrow{a} p_1 \xrightarrow{b} p_3$ and $p \xrightarrow{b} p_2 \xrightarrow{a} p_3$ in A, and by the above (and symmetry) $p \xrightarrow{a} p_1 \xrightarrow{b}$ and $p \xrightarrow{b} p_2 \xrightarrow{a}$ in A'. If we do not have the same diamond in A', then w.l.o.g. $p_1 \xrightarrow{b} q'$ in A' and this is one of the redefined arcs; by aIb and choice of the arc family, then $p_2 \xrightarrow{a} p_3$ also belongs to the family, i.e. $p_2 \xrightarrow{a} q'$ in A'.

The hp-bisimulation matches each occurrence sequence w (or its trace) in A with the same sequence in A', where we insert a τ after each a arising from some $q_a \xrightarrow{a} q$ in A from the chosen family; clearly, this is an occurrence sequence in A' ending in the same state, and we only have to check that it has the same visible po. So assume that $w = uabu'$ and we have to insert a τ after a. On the A'-side, each visible event before this τ is less than this τ which in turn is less than each visible event after this τ in the full labelled partial order defined from the extended occurrence sequence; we have to check that on the A-side each event in ua is less than each event in bu', too and we will write here $<$ for less than. The a arises from an arc into q, the b from an arc out of q, hence a and b are dependent and $a < b$; so choose some (occurrence of some) c in u and c' in u'. If $c < a$, also $c < b$; otherwise, we can commute the c behind a, i.e. there is a c-labelled arc into q and aIc; by choice of the arc family, this arc belongs to it, b and c are dependent and $c < b$. If $b < d$ we are done; otherwise, we find a d-labelled arc from q and by choice of the family $a < d$; if now $\neg c < a$, then we have a c-labelled arc into q belonging to the family and a d-labelled arc from q, thus by choice of the family $c < d$. □

As a negative example, consider simply a diamond with arcs $p \xrightarrow{a} q$ and $q \xrightarrow{b} p'$, but with a and b being dependent. This can easily be realized by a net, but inserting a τ between the two arcs makes it impossible – the τ-transition would have to have a zero-effect, but would be required to change the marking. Of course, this can be remedied by inserting another internal event on the other side of the diamond.

We conclude by the remark that our construction above coincides with the one in Section 4 in an extreme case, namely if all events are dependent and we apply the former to each arc separately.

7 Conclusion

We have considered the problem of finding a safe net (possibly with internal events) that implements the behaviour described by an asynchronous transition system without internal events, i.e. the net must be divergence-free (since clearly the transition system is) and bisimilar to the transition system in a way that does not only consider nondeterministic choices but also concurrency. We have shown how to construct an ST-bisimilar implementation and have also considered the problem of a history-preserving bisimilar implementation.

The results in this paper are only a beginning. Clearly, the problem of finding a history-preserving bisimilar implementation for an asynchronous transition system

has only been touched upon, although the author does not know of any asynchronous transition system that does not have such an implementation. Furthermore, for realistic application it is necessary to optimize the implementation by minimizing the number of places (as it is e.g. done in [CKLY98]) or – most of all – the number of additional internal transitions.

References

[BBD95] E. Badouel, L. Bernardinello, and P. Darondeau. Polynomial algorithms for the synthesis of bounded nets. In P. Mosses et al., editors, *TAPSOFT 95*, Lect. Notes Comp. Sci. 915, 364–378. Springer, 1995.

[BDKP91] E. Best, R. Devillers, A. Kiehn, and L. Pomello. Concurrent bisimulations in Petri nets. *Acta Informatica*, 28:231–264, 1991.

[CKLY98] J. Cortadella, M. Kishinevsky, L. Lavagno, and A. Yakovlev. Deriving Petri nets from finite transition systems. *IEEE Transactions on Computers*, 47:859–882, 1998.

[Dar98] P. Darondeau. Deriving unbounded Petri nets from formal languages. In D. Sangiorgi and R. de Simone, editors, *CONCUR 98*, Lect. Notes Comp. Sci. 1466, 533–548. Springer, 1998.

[Dev90] R. Devillers. The semantics of capacities in P/T-nets. In G. Rozenberg, editor, *Advances in Petri Nets 1989*, Lect. Notes Comp. Sci. 424, 128–150. Springer, 1990.

[DS93] M. Droste and R.M. Shortt. Petri nets and automata with concurrency relations – an adjunction. In M. Droste et al., editors, *Semantics of Programming Languages and Model Theory*. Gordon and Breach, 69–87. 1993.

[GV87] R.J. v. Glabbeek and F. Vaandrager. Petri net models for algebraic theories of concurrency. In J.W. de Bakker et al., editors, *PARLE Vol. II*, Lect. Notes Comp. Sci. 259, 224–242. Springer, 1987.

[GV97] R.J. v. Glabbeek and F. Vaandrager. The difference between splitting in n and n+1. *Information and Computation*, 136:109–142, 1997.

[GW89] R.J. v. Glabbeek and W.P. Weijland. Branching time and abstraction in bisimulation semantics. In G.X. Ritter, editor, *Information Processing 89*, 613–618. North-Holland, 1989.

[HKT92] P. Hoogers, H. Kleijn, and P.S. Thiagarajan. A trace semantics for Petri nets. In W. Kuich, editor, *Automata, Languages and Programming, ICALP 1992*, Lect. Notes Comp. Sci. 623, 595–604. Springer, 1992.

[NRT92] M. Nielsen, G. Rozenberg, and P.S. Thiagarajan. Elementary transition systems. *Theor. Comput. Sci.*, 96:3–33, 1992.

[NW95] M. Nielsen and G. Winskel. Models for concurrency. In S. Abramsky, D. Gabbay, and T. Maibaum, editors, *Handbook of Logic in Computer Science Vol. 4*, pages 1–148. Oxford Univ. Press, 1995.

[Pet81] J.L. Peterson. *Petri Net Theory.* Prentice-Hall, 1981.

[PKY95] M. Pietkiewicz-Koutny and A. Yakovlev. Non-pure nets and their transition systems. Technical Report TR 528, Dept. Comp. Sci., Univ. of Newcastle upon Tyne, 1995.

[Rei85] W. Reisig. *Petri Nets.* EATCS Monographs on Theoretical Computer Science 4. Springer, 1985.

[Sun98] K. Sunesen. *Reasoning about Reactive Systems.* PhD thesis, Univ. Aarhus, 1998.

[Vog92] W. Vogler. *Modular Construction and Partial Order Semantics of Petri Nets.* Lect. Notes Comp. Sci. 625. Springer, 1992.

[Vog95] W. Vogler. Timed testing of concurrent systems. *Information and Computation*, 121:149–171, 1995.

[Yak98] A. Yakovlev. Designing control logic for counterflow pipeline processor using Petri nets. *Formal Methods in System Design*, 12:39–71, 1998.