Bull. Nov. Comp. Center, Comp. Science, 7 (1997), 57–81

Equivalences for behavioural analysis of multilevel systems *

Igor V. Tarasyuk

Abstract. The paper is devoted to the investigation of behavioural equivalences of concurrent systems modelled by Petri nets. Back-forth and place bisimulation equivalences known from the literature are supplemented by new ones, and their relationship with basic behavioural equivalence relations is examined for the whole class of Petri nets as well as for their subclass of sequential nets. In addition, the preservation of all the equivalence notions by refinements is examined.

1. Introduction

The notion of equivalence is central in any theory of systems. It allows us to compare systems taking into account particular aspects of their behaviour.

Petri nets [21] became a popular formal model for design of concurrent and distributed systems. One of the main advantages of Petri nets is their ability for structural characterization of three fundamental features of concurrent computations: causality, nondeterminism and concurrency.

In recent years, a wide range of semantic equivalences was proposed in concurrency theory. Some of them were either directly defined or transferred from other formal models to Petri nets. The following basic notions of equivalences for Petri nets are known from the literature (some of them were introduced by the author in [26, 27, 28] to obtain the complete set of relations in interleaving/true concurrency and linear time/branching time semantics).

- *Trace equivalences* (respect only protocols of nets functioning): interleaving ($\equiv_i$) [13], step ($\equiv_s$) [22], partial word ($\equiv_{pw}$) [12], pomset ($\equiv_{pom}$) [24] and process

  ($\equiv_{pr}$) [26].
- *Usual bisimulation equivalences* (respect branching structure of nets functioning): interleaving ($\underline{\leftrightarrow}_i$) [20], step ($\underline{\leftrightarrow}_s$) [16], partial word ($\underline{\leftrightarrow}_{pw}$) [29], pomset ($\underline{\leftrightarrow}_{pom}$) [7] and process ($\underline{\leftrightarrow}_{pr}$) [4].
- *ST-bisimulation equivalences* (respect the duration of transition occurrences in nets functioning): interleaving ($\underline{\leftrightarrow}_{iST}$) [11], partial word ($\underline{\leftrightarrow}_{pwST}$) [29], pomset ($\underline{\leftrightarrow}_{pomST}$) [29] and process ($\underline{\leftrightarrow}_{prST}$) [26].
- *History preserving bisimulation equivalences* (respect the "past" or "history" of nets functioning): pomset ($\underline{\leftrightarrow}_{pomh}$) [25] and process ($\underline{\leftrightarrow}_{prh}$) [26].

*The work is supported by Volkswagen Fund, grant I/70 564 and INTAS-RFBR, grant 95-0378

- *Conflict preserving equivalences* (completely respect conflicts in nets): multi event structure ($\equiv_{mes}$) [27] and occurrence ($\equiv_{occ}$) [15].
- *Isomorphism* ($\simeq$) [21] (i.e. coincidence of nets up to renaming of places and transitions).

Recently, two important groups of equivalence relations were introduced: back-forth and place bisimulation equivalences. Back-forth bisimulation equivalences are based on the idea that bisimulation relation do not only require systems to simulate each other behaviour in the forward direction (as usually) but also when going back in history. They are closely connected with equivalences of logics with past modalities.

These equivalence notions were initially introduced in [14] in the framework of transition systems. It was shown that back-forth variant ($\underline{\leftrightarrow}_{ibif}$) of interleaving bisimulation equivalence coincide with ordinary $\underline{\leftrightarrow}_i$.

In [8, 9, 10] the new variants of step ($\underline{\leftrightarrow}_{sbsf}$), partial word ($\underline{\leftrightarrow}_{pwbpwf}$) and pomset ($\underline{\leftrightarrow}_{pombpomf}$) back-forth bisimulation equivalences were defined in the framework of prime event structures and compared with usual, ST- and history preserving bisimulation equivalences. It was demonstrated that among all back-forth bisimulation equivalences only $\underline{\leftrightarrow}_{pombpomf}$ is preserved by refinements (it coincides with $\underline{\leftrightarrow}_{pomh}$ which has such a property).

In [23] the new idea of differentiating the kinds of back and forth simulations appeared (following this idea, it is possible, for example, to define step back pomset forth bisimulation equivalence ($\underline{\leftrightarrow}_{sbpomf}$)). The set of all possible back-forth equivalence notions was proposed in interleaving, step, partial word and pomset semantics. Two new notions which do not coincide with known ones were proposed: step back partial word forth ($\underline{\leftrightarrow}_{sbpwf}$) and step back pomset forth ($\underline{\leftrightarrow}_{sbpomf}$) bisimulation equivalences. It was proved that the former is not preserved by refinements, and the question was addressed about the latter.

Place bisimulation equivalences were initially introduced in [1] on the basis of definition from [17, 18, 19]. Place bisimulations are relations over places instead of markings or processes. The relation on markings is obtained using the *"lifting"* of relation on places. The main application of place bisimulation equivalences is effective behaviour preserving reduction technique for Petri nets based on them.

In [1, 2] interleaving place bisimulation equivalence ($\sim_i$) was proposed. In these papers also strict interleaving place bisimulation equivalence ($\approx_i$) was defined, by imposing the additional requirement stating that corresponding transitions of nets must be related by bisimulation. The question about possibility of introducing history preserving place bisimulation eqivalence was addressed.

In [4, 5] step ($\sim_s$), partial word ($\sim_{pw}$), pomset ($\sim_{pom}$), process ($\sim_{pr}$) place bisimulation equivalences and their strict analogs ($\approx_s$, $\approx_{pw}$, $\approx_{pom}$, $\approx_{pr}$) were proposed. The coincidence of $\sim_i$, $\sim_s$ and $\sim_{pw}$ was established. Also it was shown that all strict bisimulation equivalences coincide with $\sim_{pr}$. Therefore, we have only three different equivalences: $\sim_i$, $\sim_{pom}$ and $\sim_{pr}$. In addition, in these papers the polinomial algorithm of net reduction was proposed which preserves the behaviour of a net (i.e. the initial and reduced nets are bisimulation equivalent).

To choose most appropriate behavioural viewpoint on systems to be modelled, having a complete set of equivalence notions in all semantics and understanding their interrelations is very important. This branch of research is usually called *comparative concurrency semantics*. To clarify the nature of equivalences and evaluate how they respect a concurrency, it is actual to consider also correlation of these notions on concurrency-free (sequential) nets. Treating equivalences for preservation by refinements allows one to decide which of them may be used for top-down design.

The main contributions of this paper are the following.

Working in the framework of Petri nets, we extend the set of back-forth equivalences from [23] by that of induced by process semantics and obtain two new notions which cannot be reduced to the known ones: step back process forth ($\underline{\leftrightarrow}_{sbprf}$) and pomset back process forth ($\underline{\leftrightarrow}_{pombprf}$) bisimulation equivalences.

We compare all back-forth and place equivalences with the set of basic behavioural notions from [26, 27, 28] giving rise to the better understanding of the nature of new (and old) notions and complete the results of [10, 23, 4, 5]. In particular, we prove that $\sim_{pr}$ implies $\underline{\leftrightarrow}_{prh}$ and answer the question from [1]: $\sim_{pr}$ is strict enough to preserve the "histories" of a net functioning. Hence, it is no sence to define history preserving place bisimulation eqivalence. Moreover, since ST- and history preserving bisimulation equivalences are consequences of $\sim_{pr}$, the algorithm of net reduction from [4, 5], based on this equivalence, preserves the *timed traces* [11] of the initial net (since ST-bisimulation equivalences are *real time consistent* [11]) and "histories" of its functionings (since history preserving bisimulation equivalences respect the "past" of processes).

In [6], SM-refinement operator for Petri nets was proposed, which "replaces" their transitions by SM-nets, a special subclass of state machine nets. We treat all the considered equivalence notions for preservation by SM-refinements and establish that among back-forth relations *only* $\underline{\leftrightarrow}_{pombpomf}$ and $\underline{\leftrightarrow}_{prbprf}$ are preserved by SM-refinements (they coincide with corresponding history presrving ones for which this result holds). So, we obtained the negative answer to the question from [23]: neither $\underline{\leftrightarrow}_{sbpomf}$ nor even $\underline{\leftrightarrow}_{pombprf}$ is preserved by refinements. We prove that $\sim_{pr}$ is the only place bisimulation equivalence which is preserved by SM-refinements.

In addition, we investigate the interrelations of all the equivalence notions on sequential nets (subclass of Petri nets corresponding to transition systems where neither transitions can be fired concurrently). The merging of most of the equivalence relations in interleaving – pomset semantics is demonstrated. We prove that on sequential nets back-forth equivalences coincide with usual forth ones.

The rest of paper is organized as follows. Basic definitions are introduced in Section 2. In Section 3 back-forth bisimulation equivalences are proposed and compared with basic equivalence relations. In Section 4 place bisimulation equivalences are defined and their interrelations with equivalence notions considered before are investigated. In Section 5 we establish which equivalence relations are preserved by SM-refinements. Section 6 is devoted to comparison of equivalences on sequential nets. Concluding Section 7 contains a review of the main results obtained and some directions of further research.

2. Basic definitions

In this section we give some basic definitions used further.

2.1. Multisets

Multisets are extension of sets by letting them to contain several equal elements.

Definition 1. Let X be some set. A finite *multiset* M over X is a mapping $M : X \to \mathbf{N}$ ($\mathbf{N}$ is a set of natural numbers) s.t $|\{x \in X \mid M(x) > 0\}| < \infty$.

$\mathcal{M}(X)$ denotes the *set of all finite multisets* over X. When $\forall x \in X$ $M(x) \leq 1$, M is a proper set. *Cardinality* of multiset M is defined in such a way: $|M| = \sum_{x \in X} M(x)$. We write $x \in M$ if $M(x) > 0$ and $M \subseteq M'$, if $\forall x \in X$ $M(x) \leq M'(x)$. We define $(M + M')(x) = M(x) + M'(x)$ and $(M - M')(x) = \max\{0, M(x) - M'(x)\}$.

2.2. Labelled nets

Labelled nets are Petri nets with transitions labelled by action names.

Let $Act = \{a, b, \ldots\}$ be a set of *action names* or *labels*.

Definition 2. A *labelled net* is a quadruple $N = \langle P_N, T_N, F_N, l_N \rangle$, where:

- $P_N = \{p, q, \ldots\}$ is a set of *places*;
- $T_N = \{t, u, \ldots\}$ is a set of *transitions*;
- $F_N : (P_N \times T_N) \cup (T_N \times P_N) \to \mathbf{N}$ is the *flow relation* with weights ($\mathbf{N}$ denotes a set of natural numbers);
- $l_N : T_N \to Act$ is a *labelling* of transitions with action names.

Given labelled nets $N = \langle P_N, T_N, F_N, l_N \rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'} \rangle$. A mapping $\beta : N \to N'$ is an *isomorphism* between N and N', denoted by $\beta : N \simeq N'$, if:

1. β is a bijection such that $\beta(P_N) = P_{N'}$ and $\beta(T_N) = T_{N'}$;
2. $\forall p \in P_N\ \forall t \in T_N\ F_N(p, t) = F_{N'}(\beta(p), \beta(t))$ and $F_N(t, p) = F_{N'}(\beta(t), \beta(p))$;
3. $\forall t \in T_N\ l_N(t) = l_{N'}(\beta(t))$.

Two labelled nets N and N' are *isomorphic*, denoted by $N \simeq N'$, if $\exists \beta : N \simeq N'$.

Given a labelled net N and some transition $t \in T_N$, the *precondition* and *postcondition* t, denoted by ${}^\bullet t$ and $t^\bullet$ respectively, are the multisets defined in such a way: $({}^\bullet t)(p) = F_N(p, t)$ and $(t^\bullet)(p) = F_N(t, p)$. Analogous definitions are introduced for places: $({}^\bullet p)(t) = F_N(t, p)$ and $(p^\bullet)(t) = F_N(p, t)$. Let ${}^\circ N = \{p \in P_N \mid {}^\bullet p = \emptyset\}$ is a set of *initial (input)* places of N and $N^\circ = \{p \in P_N \mid p^\bullet = \emptyset\}$ is a set of *final (output)* places of N.

A labelled net N is *acyclic*, if there exist no transitions $t_0, \ldots, t_n \in T_N$ such that $t_{i-1}^\bullet \cap {}^\bullet t_i \neq \emptyset$ $(1 \leq i \leq n)$ and $t_0 = t_n$. A labelled net N is *ordinary* if $\forall p \in P_N$ ${}^\bullet p$ and $p^\bullet$ are proper sets (not multisets).

Let $N = \langle P_N, T_N, F_N, l_N \rangle$ be acyclic ordinary labelled net and $x, y \in P_N \cup T_N$. Let us introduce the following notions.

- $x \prec_N y \Leftrightarrow xF_N^+ y$, where F_N^+ is a transitive closure of F_N (*strict causal dependence* relation);
- $\downarrow_N x = \{y \in P_N \cup T_N \mid y \prec_N x\}$ (the set of *strict predecessors* of x);

A set $T \subseteq T_N$ is *left-closed* in N, if $\forall t \in T$ $(\downarrow_N t) \cap T_N \subseteq T$.

2.3. Marked nets

Marked nets are labelled nets having some "tokens" in their places, and these places are considered to be "marked" ones. We can consider a behaviour of a marked net, moving these tokens in accordance to the rools of a special "token game".

A *marking* of a labelled net N is a multiset $M \in \mathcal{M}(P_N)$.

Definition 3. A *marked net (net)* is a tuple $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$, where $\langle P_N, T_N, F_N, l_N\rangle$ is a labelled net and $M_N \in \mathcal{M}(P_N)$ is the *initial* marking.

Given marked nets $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$. A mapping $\beta : N \to N'$ is an *isomorphism* between N and N', denoted by $\beta : N \simeq N'$, if:

1. $\beta : \langle P_N, T_N, F_N, l_N\rangle \simeq \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}\rangle$;
2. $\forall p \in M_N$ $M_N(p) = M_{N'}(\beta(p))$.

Two marked nets N and N' are *isomorphic*, denoted by $N \simeq N'$, if $\exists \beta : N \simeq N'$.

Let $M \in \mathcal{M}(P_N)$ be a marking of a net N. A transition $t \in T_N$ is *fireable* in M, if ${}^\bullet t \subseteq M$. If t is fireable in M, firing it yields a new marking $\widetilde{M} = M - {}^\bullet t + t^\bullet$, denoted by $M \xrightarrow{t} \widetilde{M}$. A marking M of a net N is *reachable*, if $M = M_N$ or there exists a reachable marking $\widehat{M}$ of N s.t. $\widehat{M} \xrightarrow{t} M$ for some $t \in T_N$. $Mark(N)$ denotes a *set of all reachable* markings of a net N.

2.4. Partially ordered sets

Partially ordered sets [24] are important formalism, often used as a semantical domain for concurrent systems. These are clearly represent causality and concurrency which is interpreted as a causal independence.

Definition 4. A *labelled partially ordered set (lposet)* is a triple $\rho = \langle X, \prec, l\rangle$, where:

- $X = \{x, y, \ldots\}$ is some set;
- $\prec \subseteq X \times X$ is a strict partial order (irreflexive transitive relation) over X;
- $l : X \to Act$ is a *labelling* function.

Let $\rho = \langle X, \prec, l\rangle$ and $\rho' = \langle X', \prec', l'\rangle$ be lposets.

A mapping $\beta : X \to X'$ is a *label-preserving bijection* between ρ and ρ', denoted by $\beta : \rho \approx \rho'$, if:

1. β is a bijection;
2. $\forall x \in X$ $l(x) = l'(\beta(x))$.

We write $\rho \approx \rho'$, if $\exists \beta : \rho \approx \rho'$.

A mapping $\beta : X \to X'$ is a *homomorphism* between ρ and ρ', denoted by $\beta : \rho \sqsubseteq \rho'$, if:

1. $\beta : \rho \approx \rho'$;
2. $\forall x, y \in X\ x \prec y \ \Rightarrow\ \beta(x) \prec' \beta(y)$.

We write $\rho \sqsubseteq \rho'$, if $\exists \beta : \rho \sqsubseteq \rho'$.

A mapping $\beta : X \to X'$ is an *isomorphism* between ρ and ρ', denoted by $\beta : \rho \simeq \rho'$, if $\beta : \rho \sqsubseteq \rho'$ and $\beta^{-1} : \rho' \sqsubseteq \rho$. Two lposets ρ and ρ' are *isomorphic*, denoted by $\rho \simeq \rho'$, if $\exists \beta : \rho \simeq \rho'$.

Definition 5. *Partially ordered multiset (pomset)* is an isomorphism class of lposets.

2.5. C-processes

C-processes [6] represent runs of concurrent systems and contain the information about causal dependencies of events in such runs.

Definition 6. A *causal net* is an acyclic ordinary labelled net $C = \langle P_C, T_C, F_C, l_C \rangle$, s.t:

1. $\forall r \in P_C\ |{}^\bullet r| \leq 1$ and $|r^\bullet| \leq 1$, i.e. places are unbranched;
2. $|\downarrow_C x| < \infty$, i.e. a set of causes is finite.

Let us note that on the basis of any causal net $C = \langle P_C, T_C, F_C, l_C \rangle$ one can define lposet $\rho_C = \langle T_C, \prec_N \cap (T_C \times T_C), l_C \rangle$.

The fundamental property of causal nets is [4]: if C is a causal net, then there exists an occurrence sequence ${}^\circ C = L_0 \stackrel{v_1}{\to} \cdots \stackrel{v_n}{\to} L_n = C^\circ$ such that $L_i \subseteq P_C$ $(0 \leq i \leq n)$, $P_C = \cup_{i=0}^n L_i$ and $T_C = \{v_1, \ldots, v_n\}$. Such a sequence is called a *full execution* of C.

Definition 7. Given a net N and a causal net C. A mapping $\varphi : P_C \cup T_C \to P_N \cup T_N$ is an *embedding* C into N, denoted by $\varphi : C \to N$, if:

1. $\varphi(P_C) \in \mathcal{M}(P_N)$ and $\varphi(T_C) \in \mathcal{M}(T_N)$, i.e. sorts are preserved;
2. $\forall v \in T_C\ {}^\bullet\varphi(v) = \varphi({}^\bullet v)$ and $\varphi(v)^\bullet = \varphi(v^\bullet)$, i.e. flow relation is respected;
3. $\forall v \in T_C\ l_C(v) = l_N(\varphi(v))$, i.e. labelling is preserved.

Since embeddings respect the flow relation, if ${}^\circ C \stackrel{v_1}{\to} \cdots \stackrel{v_n}{\to} C^\circ$ is a full execution of C, then $M = \varphi({}^\circ C) \stackrel{\varphi(v_1)}{\longrightarrow} \cdots \stackrel{\varphi(v_n)}{\longrightarrow} \varphi(C^\circ) = M'$ is an occurrence sequence in N.

Definition 8. A *fireable in marking M C-process (process)* of a net N is a pair $\pi = (C, \varphi)$, where C is a causal net and $\varphi : C \to N$ is an embedding such that $M = \varphi({}^\circ C)$. A fireable in M_N process is a *process* of N.

We write $\Pi(N, M)$ for a *set of all fireable* in marking M processes of a net N and $\Pi(N)$ for the *set of all processes* of a net N. The *initial* process of a net N is $\pi_N = (C_N, \varphi_N) \in \Pi(N)$, such that $T_{C_N} = \emptyset$. If $\pi \in \Pi(N, M)$, then firing of this process transforms a marking M into $M' = M - \varphi({}^\circ C) + \varphi(C^\circ) = \varphi(C^\circ)$, denoted by $M \xrightarrow{\pi} M'$.

Let $\pi = (C, \varphi),\ \tilde{\pi} = (\widetilde{C}, \tilde{\varphi}) \in \Pi(N)$ and $\hat{\pi} = (\widehat{C}, \hat{\varphi}) \in \Pi(N, \varphi(C^\circ))$.

A process $\tilde{\pi}$ is an *extension* of π *by process* $\hat{\pi}$, denoted by $\pi \xrightarrow{\hat{\pi}} \tilde{\pi}$, if $T_C \subseteq T_{\widetilde{C}}$ is a left-closed set in $\widetilde{C}$ and $T_{\widehat{C}} = T_{\widetilde{C}} \setminus T_C$.

A process $\tilde{\pi}$ is an extension of a process π *by one transition* $v \in T_{\widetilde{C}}$, denoted by $\pi \xrightarrow{v} \tilde{\pi}$, if $\pi \xrightarrow{\hat{\pi}} \tilde{\pi}$ and $T_{\widehat{C}} = \{v\}$.

A process $\tilde{\pi}$ is an extension of a process π *by sequence of transitions* $\sigma = v_1 \cdots v_n \in T^*_{\widetilde{C}}$, denoted by $\pi \xrightarrow{\sigma} \tilde{\pi}$, if $\exists \pi_i \in \Pi(N)\ (1 \leq i \leq n)\ \pi \xrightarrow{v_1} \pi_1 \xrightarrow{v_2} \ldots \xrightarrow{v_n} \pi_n = \tilde{\pi}$.

3. Back-forth bisimulation equivalences

In this section, in the framework of Petri nets, we supplement the definitions of back-forth bisimulation equivalences [23] by the new notions induced by process semantics and compare them with basic ones.

3.1. Definitions of back-forth bisimulation equivalences

The definitions of back-forth bisimulation equivalences are based on the following notion of sequential run.

Definition 9. A *sequential run* of a net N is a pair (π, σ), where:

- a process $\pi \in \Pi(N)$ contains the information about causal dependencies of transitions which brought to this state;
- a sequence $\sigma \in T^*_C$ such that $\pi_N \xrightarrow{\sigma} \pi$, contains the information about the order in which the transitions occur which brought to this state.

Let us denote the *set of all sequential runs* of a net N by $Runs(N)$.

The *initial* sequential run of a net N is a pair (π_N, ε), where ε is an empty sequence.

Let $(\pi, \sigma),\ (\tilde{\pi}, \tilde{\sigma}) \in Runs(N)$. We write $(\pi, \sigma) \xrightarrow{\hat{\pi}} (\tilde{\pi}, \tilde{\sigma})$, if $\pi \xrightarrow{\hat{\pi}} \tilde{\pi}$, $\exists \hat{\sigma} \in T^*_{\widehat{C}}\ \pi \xrightarrow{\hat{\sigma}} \tilde{\pi}$ and $\tilde{\sigma} = \sigma\hat{\sigma}$.

Definition 10. Let N and N' be some nets. A relation $\mathcal{R} \subseteq Runs(N) \times Runs(N')$ is a *$\star$-back $\star\star$-forth bisimulation* between N and N', $\star, \star\star \in \{$*interleaving, step, partial word, pomset, process*$\}$, denoted by
$\mathcal{R} : N \underline{\leftrightarrow}_{\star b \star\star f} N'$, $\star, \star\star \in \{i, s, pw, pom, pr\}$, if:

1. $((\pi_N, \varepsilon), (\pi_{N'}, \varepsilon)) \in \mathcal{R}$.
2. $((\pi, \sigma), (\pi', \sigma')) \in \mathcal{R}$

- (back)

  $(\tilde{\pi}, \tilde{\sigma}) \stackrel{\hat{\pi}}{\rightarrow} (\pi, \sigma)$,

  (a) $|T_{\widehat{C}}| = 1$, if $\star = i$;
  (b) $\prec_{\widehat{C}} = \emptyset$, if $\star = s$;

  $\Rightarrow\ \exists(\tilde{\pi}', \tilde{\sigma}') :\ (\tilde{\pi}', \tilde{\sigma}') \stackrel{\hat{\pi}'}{\rightarrow} (\pi', \sigma'),\ ((\tilde{\pi}, \tilde{\sigma}), (\tilde{\pi}', \tilde{\sigma}')) \in \mathcal{R}$ and

  (a) $\rho_{\widehat{C}'} \sqsubseteq \rho_{\widehat{C}}$, if $\star = pw$;
  (b) $\rho_{\widehat{C}} \simeq \rho_{\widehat{C}'}$, if $\star \in \{i, s, pom\}$;
  (c) $\widehat{C} \simeq \widehat{C}'$, if $\star = pr$;

- (forth)

  $(\pi, \sigma) \stackrel{\hat{\pi}}{\rightarrow} (\tilde{\pi}, \tilde{\sigma})$,

  (a) $|T_{\widehat{C}}| = 1$, if $\star\star = i$;
  (b) $\prec_{\widehat{C}} = \emptyset$, if $\star\star = s$;

  $\Rightarrow\ \exists(\tilde{\pi}', \tilde{\sigma}') :\ (\pi', \sigma') \stackrel{\hat{\pi}'}{\rightarrow} (\tilde{\pi}', \tilde{\sigma}'),\ ((\tilde{\pi}, \tilde{\sigma}), (\tilde{\pi}', \tilde{\sigma}')) \in \mathcal{R}$ and

  (a) $\rho_{\widehat{C}'} \sqsubseteq \rho_{\widehat{C}}$, if $\star\star = pw$;
  (b) $\rho_{\widehat{C}} \simeq \rho_{\widehat{C}'}$, if $\star\star \in \{i, s, pom\}$;
  (c) $\widehat{C} \simeq \widehat{C}'$, if $\star\star = pr$.

3. As item 2, but the roles of N and N' are reversed.

Two nets N and N' *$\star$-back $\star\star$-forth bisimulation equivalent*, $\star, \star\star \in$ {*interleaving, step, partial word, pomset, process*}, denoted by $N \underline{\leftrightarrow}_{\star b \star\star f} N'$, if $\exists \mathcal{R} : N \underline{\leftrightarrow}_{\star b \star\star f} N',\ \star, \star\star \in \{i, s, pw, pom, pr\}$.

3.2. Interrelations of back-forth bisimulation equivalences

In back-forth bisimulations, moveing back from a state is possible only along the history which brought to the state. Such a determinism implies merging of some equivalences.

Proposition 1. *Let $\star \in \{i, s, pw, pom, pr\}$. For nets N and N' the following holds:*

1. $N \underline{\leftrightarrow}_{pwb\star f} N' \ \Leftrightarrow\ N \underline{\leftrightarrow}_{pomb\star f} N'$;
2. $N \underline{\leftrightarrow}_{\star bif} N' \ \Leftrightarrow\ N \underline{\leftrightarrow}_{\star b\star f} N'$.

In Figure 1 dashed lines embrace coinciding back-forth bisimulation equivalences.

Hence, interrelations of the remaining back-forth equivalences may be represented by the graph in Figure 2.

3.3. Interrelations of back-forth bisimulation and basic equivalences

Let us consider how back-forth equivalences are connected with basic ones.

$$\begin{tikzcd}
\underline{\leftrightarrow}_{prbif} \arrow[d] & \underline{\leftrightarrow}_{prbsf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{prbpwf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{prbpomf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{prbprf} \arrow[l] \arrow[d] \\
\underline{\leftrightarrow}_{pombif} \arrow[d] & \underline{\leftrightarrow}_{pombsf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{pombpwf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{pombpomf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{pombprf} \arrow[l] \arrow[d] \\
\underline{\leftrightarrow}_{pwbif} \arrow[d] & \underline{\leftrightarrow}_{pwbsf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{pwbpwf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{pwbpomf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{pwbprf} \arrow[l] \arrow[d] \\
\underline{\leftrightarrow}_{sbif} \arrow[d] & \underline{\leftrightarrow}_{sbsf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{sbpwf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{sbpomf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{sbprf} \arrow[l] \arrow[d] \\
\underline{\leftrightarrow}_{ibif} & \underline{\leftrightarrow}_{ibsf} \arrow[l] & \underline{\leftrightarrow}_{ibpwf} \arrow[l] & \underline{\leftrightarrow}_{ibpomf} \arrow[l] & \underline{\leftrightarrow}_{ibprf} \arrow[l]
\end{tikzcd}$$

Figure 1. Merging of back-forth bisimulation equivalences

$$\begin{tikzcd}
& & & & \underline{\leftrightarrow}_{prbprf} \arrow[d] \\
& & & \underline{\leftrightarrow}_{pombpomf} \arrow[d] & \underline{\leftrightarrow}_{pombprf} \arrow[l] \arrow[d] \\
& \underline{\leftrightarrow}_{sbsf} \arrow[d] & \underline{\leftrightarrow}_{sbpwf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{sbpomf} \arrow[l] \arrow[d] & \underline{\leftrightarrow}_{sbprf} \arrow[l] \arrow[d] \\
\underline{\leftrightarrow}_{ibif} & \underline{\leftrightarrow}_{ibsf} \arrow[l] & \underline{\leftrightarrow}_{ibpwf} \arrow[l] & \underline{\leftrightarrow}_{ibpomf} \arrow[l] & \underline{\leftrightarrow}_{ibprf} \arrow[l]
\end{tikzcd}$$

Figure 2. Interrelations of back-forth bisimulation equivalences

$$
\begin{tikzcd}
 & & & & & & & & \simeq \arrow[d] & \\
 & & & & & & \equiv_{mes} \arrow[d] & & \equiv_{occ} \arrow[ll] \arrow[d] & \\
 & & & & & & \underline{\leftrightarrow}_{pomh} \arrow[dd] & & \underline{\leftrightarrow}_{prh} \arrow[dr] \arrow[dd] & \\
 & & & & & & & & & \underline{\leftrightarrow}_{pombprf} \arrow[lllu] \arrow[dd] \\
 & & \underline{\leftrightarrow}_{iST} \arrow[dd] & & \underline{\leftrightarrow}_{pwST} \arrow[ll] \arrow[dd] & & \underline{\leftrightarrow}_{pomST} \arrow[ll] \arrow[dr] \arrow[dd] & & \underline{\leftrightarrow}_{prST} \arrow[ll] \arrow[dr] \arrow[dd] & \\
 & & & \underline{\leftrightarrow}_{sbsf} \arrow[dl] & & \underline{\leftrightarrow}_{sbpwf} \arrow[ll] \arrow[dl] & & \underline{\leftrightarrow}_{sbpomf} \arrow[ll] \arrow[dl] & & \underline{\leftrightarrow}_{sbprf} \arrow[ll] \arrow[dl] \\
\underline{\leftrightarrow}_{i} \arrow[d] & & \underline{\leftrightarrow}_{s} \arrow[ll] \arrow[d] & & \underline{\leftrightarrow}_{pw} \arrow[ll] \arrow[d] & & \underline{\leftrightarrow}_{pom} \arrow[ll] \arrow[d] & & \underline{\leftrightarrow}_{pr} \arrow[ll] \arrow[d] & \\
\equiv_{i} & & \equiv_{s} \arrow[ll] & & \equiv_{pw} \arrow[ll] & & \equiv_{pom} \arrow[ll] & & \equiv_{pr} \arrow[ll] &
\end{tikzcd}
$$

Figure 3. Interrelations of back-forth bisimulation and basic equivalences

Proposition 2. *Let $\star \in \{i, s, pw, pom, pr\}$, $\star\star \in \{pom, pr\}$. For nets N and N' the following holds:*

1. $N \underline{\leftrightarrow}_{ib\star f} N' \;\Leftrightarrow\; N \underline{\leftrightarrow}_{\star} N'$;
2. $N \underline{\leftrightarrow}_{\star\star b \star\star f} N' \;\Leftrightarrow\; N \underline{\leftrightarrow}_{\star\star h} N'$;
3. $N \underline{\leftrightarrow}_{\star\star ST} N' \;\Rightarrow\; N \underline{\leftrightarrow}_{sb\star\star f} N'$.

In the following, the symbol '_' will denote the empty altermative.

Theorem 1. *Let $\leftrightarrow, \twoheadleftarrow\!\!\!\twoheadrightarrow \in \{\equiv, \underline{\leftrightarrow}, \simeq\}$ and $\star, \star\star \in \{\_, i, s, pw, pom, pr, iST, pwST, pomST, prST, pomh, prh, mes, occ, sbsf, sbpwf, sbpomf, sbprf, pombprf\}$. For nets N and N' the following holds: $N \leftrightarrow_{\star} N' \;\Rightarrow\; N \twoheadleftarrow\!\!\!\twoheadrightarrow_{\star\star} N'$ iff in graph in Figure 3 there exists a directed path from $\leftrightarrow_{\star}$ to $\twoheadleftarrow\!\!\!\twoheadrightarrow_{\star\star}$.*

Proof. ($\Leftarrow$) By definitions of the equivalences.

($\Rightarrow$) An absence of additional nontrivial arrows in the graph in Figure 3 is proved by the following examples.

- In Figure 4(a) $N \underline{\leftrightarrow}_{i} N'$, but $N \not\equiv_{s} N'$, since only in the net N' actions a and b cannot happen concurrently.
- In Figure 4(c) $N \underline{\leftrightarrow}_{iST} N'$, but $N \not\equiv_{pw} N'$, since for the pomset corresponding to the net N there is no even less sequential pomset in N'.
- In Figure 4(b) $N \underline{\leftrightarrow}_{pwh} N'$, but $N \not\equiv_{pom} N'$, since only in the net N' action b can depend on action a.
- In Figure 4(d) $N \equiv_{mes} N'$, but $N \not\equiv_{pr} N'$, since N' is a causal net which is not isomorphic to the causal net N (because of additional output place).

- In Figure 4(e) $N \equiv_{pr} N'$, but $N \not\underline{\leftrightarrow}_i N'$, since only in the net N' action a can happen so that action b can not happen afterwards.
- In Figure 5(a) $N \underline{\leftrightarrow}_{pr} N'$, but $N \not\underline{\leftrightarrow}_{iST} N'$, since only in the net N' action a can start so that no action b can begin working until finishing of a.
- In Figure 5(b) $N \underline{\leftrightarrow}_{prST} N'$, but $N \not\underline{\leftrightarrow}_{pomh} N'$, since only in the net N' after action a action b can happen so that action c must depend on a.
- In Figure 5(c) $N \underline{\leftrightarrow}_{prh} N'$, but $N \not\equiv_{mes} N'$, since only the MES corresponding to the net N' has two conflict actions a.
- In Figure 5(d) $N \equiv_{occ} N'$, but $N \not\simeq N'$, since unfireable transitions of the nets N and N' are labelled by different actions (a and b).
- In Figure 4(c) $N \underline{\leftrightarrow}_{sbsf} N'$, but $N \not\equiv_{pw} N'$.
- In Figure 6(a) $N \underline{\leftrightarrow}_{sbpwf} N'$, but $N \not\equiv_{pom} N'$, since only in the net N' action c can depend on actions a and b.
- In Figure 6(b) $N \underline{\leftrightarrow}_{sbprf} N'$, but $N \not\underline{\leftrightarrow}_{iST} N'$, since only in the net N' action a can start so that:

  1. until a finishes the sequence of actions bc cannot happen, and
  2. immediately after finishing of a action c cannot happen.

- In Figure 6(c) $N \underline{\leftrightarrow}_{pombprf} N'$, but $N \not\underline{\leftrightarrow}_{prST} N'$, since only in the net N' the process with action a can start so that it can be extended by process with action b in the only way (i.e. so that extended process be unique).
- In Figure 4(b) $N \underline{\leftrightarrow}_{pwST} N'$, but $N \not\underline{\leftrightarrow}_{sbsf} N'$, since only in the net N' the sequence of actions ab can happen so that b must depend on a.
- In Figure 5(a) $N \underline{\leftrightarrow}_{pr} N'$, but $N \not\underline{\leftrightarrow}_{sbsf} N'$, since only in the net N' action a can happen so that action b must depend on a. $\square$

4. Place bisimulation equivalences

In this section place bisimulation equivalences from [4] are compared with back-forth bisimulation and basic equivalences.

4.1. Definitions of place bisimulation equivalences

Usual bisimulations may be defined on the basis of markings (instead of processes) by replacing in processes by corresponding markings in the definitions.

Definition 11. Let N and N' be some nets. A relation $\mathcal{R} \subseteq Mark(N) \times Mark(N')$ is a $\star$-*bisimulation* between N and N', $\star \in$ {*interleaving, step, partial word, pomset, process*}, denoted by $\mathcal{R} : N \underline{\leftrightarrow}_\star N'$, $\star \in \{i, s, pw, pom,$
$pr\}$, if:

1. $(M_N, M_{N'}) \in \mathcal{R}$.
2. $(M, M') \in \mathcal{R}$, $M \stackrel{\hat{\pi}}{\to} \widetilde{M}$,

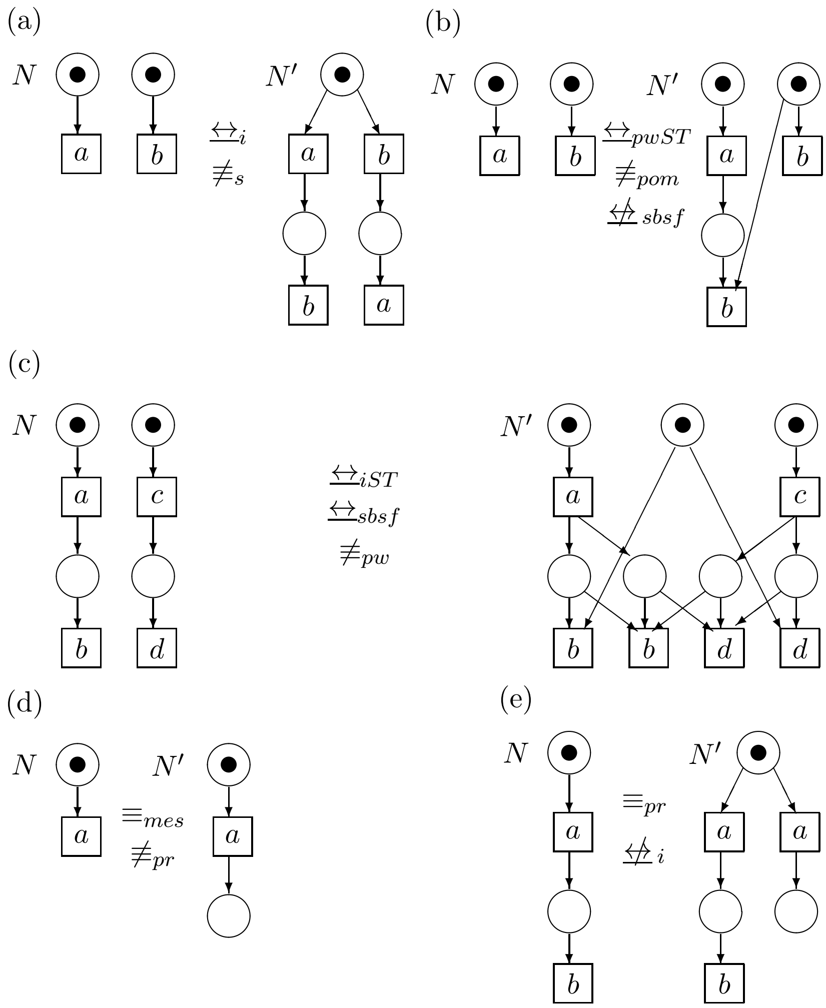

Figure 4. Examples of basic equivalences

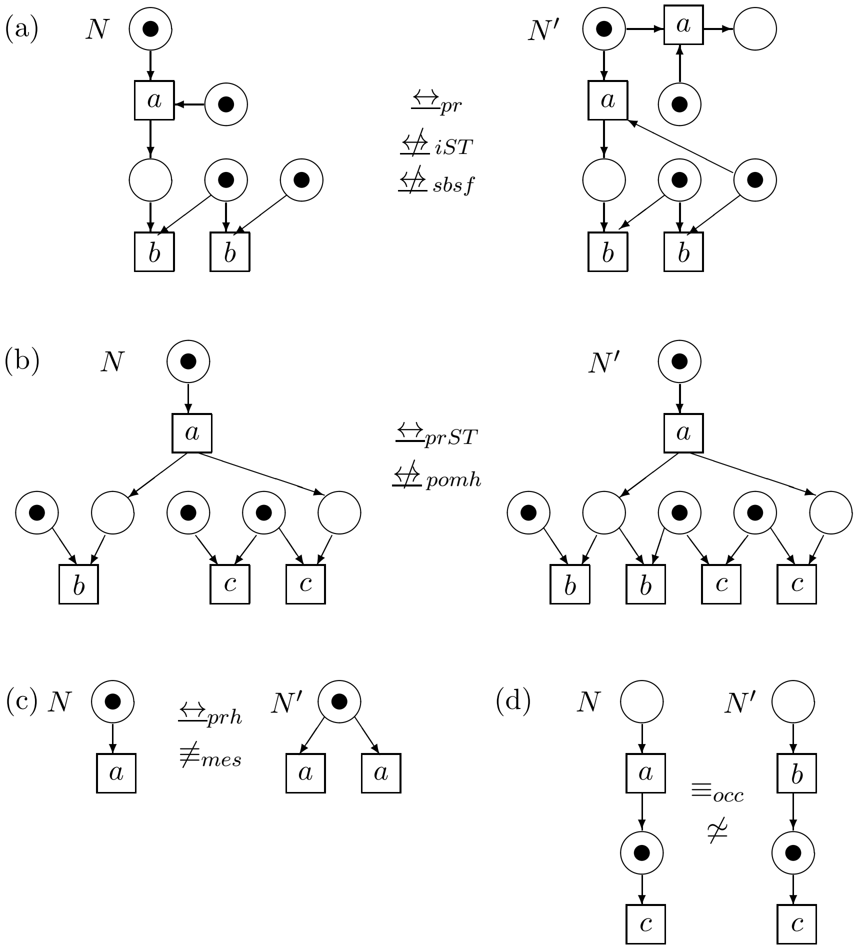

Figure 5. Examples of basic equivalences (continued)

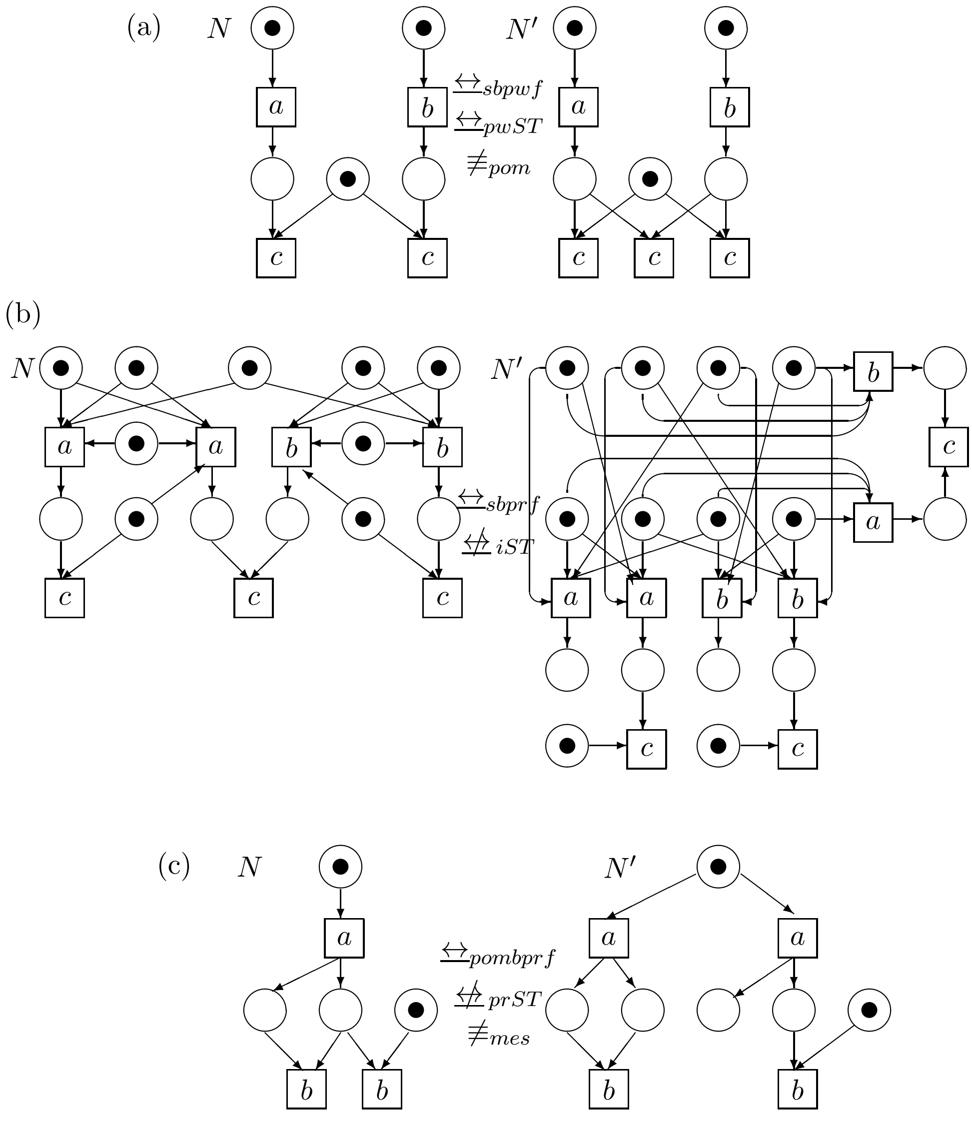

Figure 6. Examples of back-forth bisimulation equivalences

(a) $|T_{\widehat{C}}| = 1$, if $\star = i$;
(b) $\prec_{\widehat{C}} = \emptyset$, if $\star = s$;

$\Rightarrow \ \exists \widetilde{M'} : \ M' \stackrel{\hat{\pi}'}{\to} \widetilde{M'}, \ (\widetilde{M}, \widetilde{M'}) \in \mathcal{R}$ and

(a) $\rho_{\widehat{C}'} \sqsubseteq \rho_{\widehat{C}}$, if $\star = pw$;
(b) $\rho_{\widehat{C}} \simeq \rho_{\widehat{C}'}$, if $\star \in \{i, s, pom\}$;
(c) $\widehat{C} \simeq \widehat{C}'$, if $\star = pr$.

3. As item 2, but the roles of N and N' are reversed.

Two nets N and N' are $\star$-*bisimulation equivalent*, $\star \in$ {*interleaving, step, partial word, pomset, process*}, denoted by $N \underline{\leftrightarrow}_\star N'$, if $\exists \mathcal{R} : N \underline{\leftrightarrow}_\star N', \ \star \in \{i, s, pw, pom, pr\}$.

Place bisimulations are relations between places instead of markings. A relation on markings is obtained with use of *"lifting"* of bisimulation relation on places.

Let us note that in the definitions of bisimulations based on markings any markings may be used, not reachable only. As mentioned [4, 5], this does not change bisimulation equivalences.

Definition 12. Let for nets N and N' $\mathcal{R} \subseteq P_N \times P_{N'}$ be a relation between their places. A *lifting* of $\mathcal{R}$ is a relation $\overline{\mathcal{R}} \subseteq \mathcal{M}(P_N) \times \mathcal{M}(P_{N'})$, defined as follows:

$$(M, M') \in \overline{\mathcal{R}} \ \Leftrightarrow \ \left\{ \begin{array}{l} \exists \{(p_1, p'_1), \ldots, (p_n, p'_n)\} \in \mathcal{M}(\mathcal{R}) : \\ M = \{p_1, \ldots p_n\}, \ M' = \{p'_1, \ldots p'_n\} \end{array} \right.$$

Definition 13. Let N and N' be some nets. A relation $\mathcal{R} \subseteq P_N \times P_{N'}$ is a $\star$-*place bisimulation* between N and N', $\star \in$ {*interleaving, step, partial word, pomset, process*}, denoted by $\mathcal{R} : N \sim_\star N'$, if $\overline{\mathcal{R}} : N \underline{\leftrightarrow}_\star N', \ \star \in \{i, s, pw, pom, pr\}$.

Two nets N and N' are $\star$-*place bisimulation equivalent*, $\star \in$ {*interleaving, step, partial word, pomset, process*}, denoted by $N \sim_\star N'$, if $\exists \mathcal{R} : N \sim_\star N', \ \star \in \{i, s, pw, pom, pr\}$.

Strict place bisimulation equivalences are defined using the additional requirement stating that corresponding tansitions of nets must (as well as makings) be related by $\overline{\mathcal{R}}$. This relation is defined on transitions as follows.

Definition 14. Let for nets N and N' $t \in T_N, \ t' \in T_{N'}$. Then

$$(t, t') \in \overline{\mathcal{R}} \ \Leftrightarrow \ \left\{ \begin{array}{l} ({}^\bullet t, {}^\bullet t') \in \overline{\mathcal{R}} \ \wedge \\ (t^\bullet, t'^\bullet) \in \overline{\mathcal{R}} \ \wedge \\ l_N(t) = l_{N'}(t') \end{array} \right.$$

$$\begin{tikzcd}
\approx_i \arrow[d] & \approx_s \arrow[l] \arrow[d] & \approx_{pw} \arrow[l] \arrow[d] & \approx_{pom} \arrow[l] \arrow[d] & \approx_{pr} \arrow[l] \arrow[d] \\
\sim_i & \sim_s \arrow[l] & \sim_{pw} \arrow[l] & \sim_{pom} \arrow[l] & \sim_{pr} \arrow[l]
\end{tikzcd}$$

Figure 7. Merging of place bisimulation equivalences

$$\begin{tikzcd}
\sim_i & \sim_{pom} \arrow[l] & \sim_{pr} \arrow[l]
\end{tikzcd}$$

Figure 8. Interrelations of place bisimulation equivalences

Definition 15. Let N and N' be some nets. A relation $\mathcal{R} \subseteq P_N \times P_{N'}$ is a *strict $\star$-place bisimulation* between N and N', $\star \in$\{*interleaving, step, partial word, pomset, process*\}, denoted by $\mathcal{R} : N \approx_\star N'$, $\star \in \{i, s, pw, pom, pr\}$, if:

1. $\overline{\mathcal{R}} : N \underline{\leftrightarrow}_\star N'$.
2. In the definition of $\star$-bisimulation in item 2 (and in item 3 symmetrically) the new requirement is added: $\forall v \in T_{\widehat{C}}\ (\hat{\varphi}(v), \hat{\varphi}'(\beta(v))) \in \overline{\mathcal{R}}$, where:

   (a) $\beta : \rho_{\widehat{C}'} \sqsubseteq \rho_{\widehat{C}}$, if $\star = pw$;

   (b) $\beta : \rho_{\widehat{C}} \simeq \rho_{\widehat{C}'}$, if $\star \in \{i, s, pom\}$;

   (c) $\beta : \widehat{C} \simeq \widehat{C}'$, if $\star = pr$.

Two nets N and N' are *strict $\star$-place bisimulation equivalent*, $\star \in$ \{*interleaving, step, partial word, pomset, process*\}, denoted by $N \approx_\star N'$, if $\exists \mathcal{R} : N \approx_\star N'$, $\star \in \{i, s, pw, pom, pr\}$.

An important property of place bisimulations is *additivity.* Let for nets N and N' $\mathcal{R} : N \sim_\star N'$. Then $(M_1, M_1') \in \overline{\mathcal{R}}$ and $(M_2, M_2') \in \overline{\mathcal{R}}$ implies $((M_1 + M_2), (M_1' + M_2')) \in \overline{\mathcal{R}}$. In particular, if we add n tokens in each of the places $p \in P_N$ and $p' \in P_{N'}$ s.t. $(p, p') \in \mathcal{R}$, then the nets obtained as a result of such a changing of the initial markings, must be also place bisimulation equivalent.

4.2. Interrelations of place bisimulation equivalences

Let us consider interrelations of place bisimulation equivalences.

Proposition 3. *[4, 5] For nets N and N' the following holds:*

1. $N \sim_i N' \;\Leftrightarrow\; N \sim_{pw} N'$;
2. $N \sim_{pr} N' \;\Leftrightarrow\; N \approx_i N' \;\Leftrightarrow\; N \approx_{pr} N'$.

In Figure 7 dashed lines embrace coinciding place bisimulation equivalences.

Hence, interrelations of place bisimulation equivalences may be represented by graph in Figure 8.

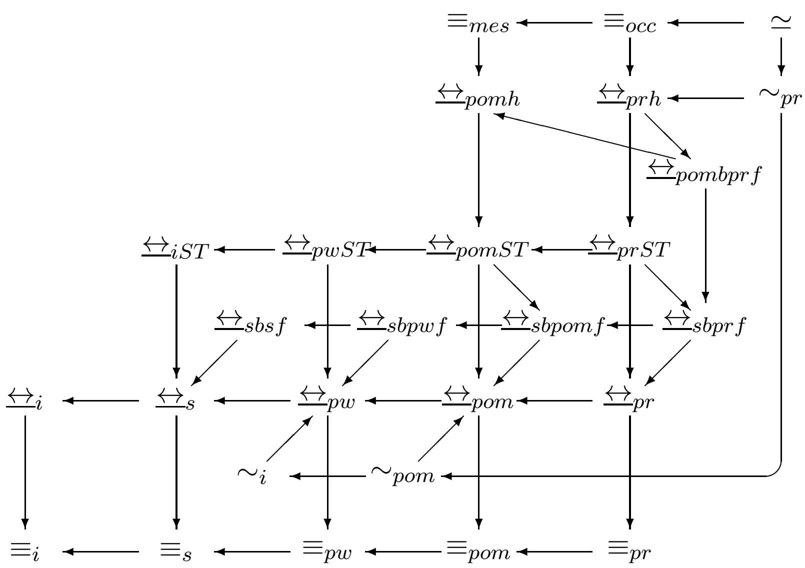

Figure 9. Interrelations of place bisimulation equivalences with basic equivalences and back-forth bisimulation equivalences

4.3. Interrelations of place bisimulation equivalences with basic equivalences and back-forth bisimulation equivalences

Let us consider interrelations of place bisimulation equivalences with basic equivalences and back-forth bisimulation equivalences.

Proposition 4. *For nets N and N' the following holds: $N \sim_{pr} N' \;\Rightarrow\; N\underline{\leftrightarrow}_{prh}N'$.*

Theorem 2. *Let $\leftrightarrow, \twoheadleftarrow\!\!\!\twoheadrightarrow \in \{\equiv, \underline{\leftrightarrow}, \sim, \simeq\}$, $\star, \star\star \in \{\_, i, s, pw, pom, pr, iST,$ $pwST, pomST, prST, pomh, prh, mes, occ, sbsf, sbpwf, sbpomf, sbprf, pombprf\}$. For nets N and N' the following holds: $N \leftrightarrow_\star N' \;\Rightarrow\; N \twoheadleftarrow\!\!\!\twoheadrightarrow_{\star\star} N'$ iff in the graph in Figure 9 there exists a directed path from $\leftrightarrow_\star$ to $\twoheadleftarrow\!\!\!\twoheadrightarrow_{\star\star}$.*

Proof. ($\Leftarrow$) By definitions of the equivalences.

($\Rightarrow$) An absence of additional nontrivial arrows in the graph in Figure 9 is proved by Theorem 1 and the following examples. Let us note that dashed lines in Figure 10 connect places related by place bisimulation.

- In Figure 10(a) $N \sim_i N'$, but $N \not\equiv_{pom} N'$, since only in the net N' action b can depend on a.
- In Figure 10(b) $N \sim_{pom} N'$, but $N \not\equiv_{pr} N'$, since only in the net N' the transition with label a has two input (and two output) places.
- In Figure 10(c) $N \equiv_{occ} N'$, but $N \not\sim_i N'$, since any place bisimulation must relate input places of the nets N and N'. But if we put an additional token in each of these places, then the action c can happen only in N'.

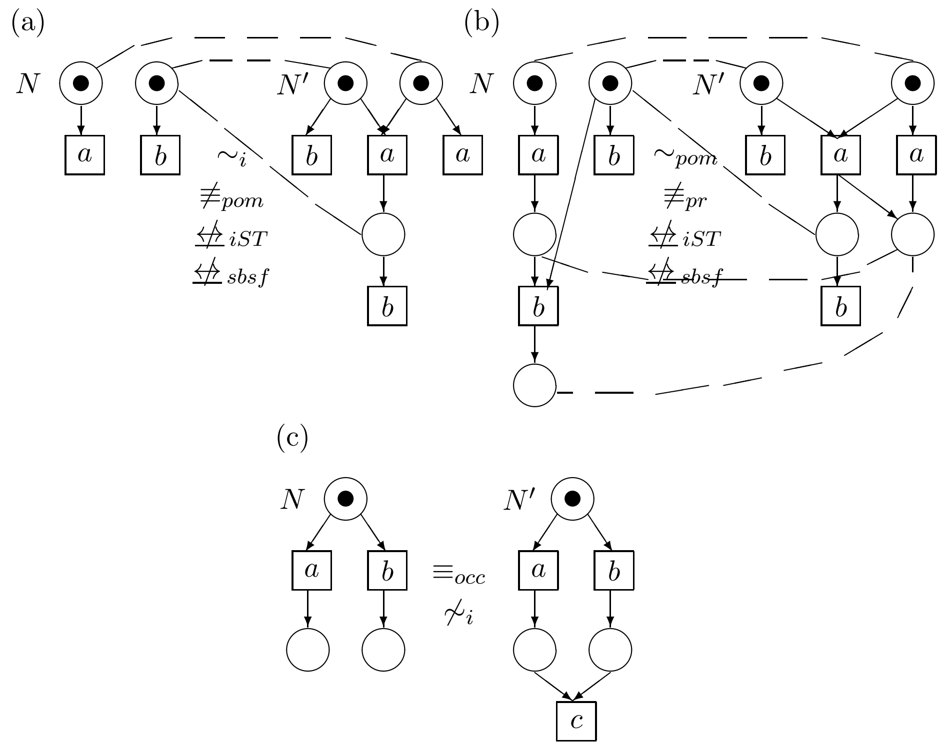

Figure 10. Examples of place bisimulation equivalences

- In Figure 10(b) $N \sim_{pom} N'$, but $N \underline{\not\leftrightarrow}_{iST} N'$, since only in the net N' action a can start so that no b can begin working until finishing of a.
- In Figure 5(c) $N \sim_{pr} N'$, but $N \not\equiv_{mes} N'$, only the MES corresponding to the net N', has two conflict actions a.
- In Figure 10(b) $N \sim_{pom} N'$, but $N \underline{\not\leftrightarrow}_{sbsf} N'$, since only in the net N' action a can happen so that b must depend on a. □

5. Preservation of the equivalences by refinements

Let us consider which equivalences may be used for top-down design.

Definition 16. An *SM-net* is a net $D = \langle P_D, T_D, F_D, l_D, M_D \rangle$ such that:

1. $\exists p_{in},\ p_{out} \in P_D$ such that $p_{in} \neq p_{out}$ and ${}^\circ D = \{p_{in}\},\ D^\circ = \{p_{out}\}$, i.e. net D has unique input and unique output place.
2. $M_D = \{p_{in}\}$ and $\forall M \in Mark(D)\ (p_{out} \in M \ \Rightarrow\ M = \{p_{out}\})$, i.e. at the beginning there is unique token in p_{in}, and at the end there is unique token in p_{out};

3. $p_{in}^{\bullet}$ and ${}^{\bullet}p_{out}$ are proper sets (not multisets), i.e. p_{in} (respectively p_{out}) represents a set of all tokens consumed (respectively produced) for any refined transition.
4. $\forall t \in T_D\ |{}^{\bullet}t| = |t^{\bullet}| = 1$, i.e. each transition has exactly one input and one output place.

SM-refinement operator "replaces" all transitions with particular label of a net by SM-net.

Definition 17. Let $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ be some net, $a \in l_N(T_N)$ and $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$ be SM-net. An *SM-refinement*, denoted by $ref(N, a, D)$, is (up to isomorphism) a net $\overline{N} = \langle P_{\overline{N}}, T_{\overline{N}}, F_{\overline{N}}, l_{\overline{N}}, M_{\overline{N}}\rangle$, where:

- $P_{\overline{N}} = P_N \cup \{\langle p, u\rangle \mid p \in P_D \setminus \{p_{in}, p_{out}\},\ u \in l_N^{-1}(a)\}$;
- $T_{\overline{N}} = (T_N \setminus l_N^{-1}(a)) \cup \{\langle t, u\rangle \mid t \in T_D,\ u \in l_N^{-1}(a)\}$;
- $F_{\overline{N}}(\bar{x}, \bar{y}) = \begin{cases} F_N(\bar{x}, \bar{y}), & \bar{x}, \bar{y} \in P_N \cup (T_N \setminus l_N^{-1}(a)); \\ F_D(x, y), & \bar{x} = \langle x, u\rangle,\ \bar{y} = \langle y, u\rangle,\ u \in l_N^{-1}(a); \\ F_N(\bar{x}, u), & \bar{y} = \langle y, u\rangle,\ \bar{x} \in {}^{\bullet}u,\ u \in l_N^{-1}(a),\ y \in p_{in}^{\bullet}; \\ F_N(u, \bar{y}), & \bar{x} = \langle x, u\rangle,\ \bar{y} \in {}^{\bullet}u,\ u \in l_N^{-1}(a),\ x \in {}^{\bullet}p_{out}; \\ 0, & \text{otherwise;} \end{cases}$
- $l_{\overline{N}}(\bar{u}) = \begin{cases} l_N(\bar{u}), & \bar{u} \in T_N \setminus l_N^{-1}(a); \\ l_D(t), & \bar{u} = \langle t, u\rangle,\ t \in T_D,\ u \in l_N^{-1}(a); \end{cases}$
- $M_{\overline{N}}(p) = \begin{cases} M_N(p), & p \in P_N; \\ 0, & \text{otherwise.} \end{cases}$

Some equivalence on nets is *preserved by refinements*, if equivalent nets remain equivalent after applying any refinement operator to them accordingly.

Theorem 3. *Let* $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}, \sim, \simeq\}$ *and* $\star \in \{\_, i, s, pw, pom, pr, iST, pwST, pomST, prST, pomh, prh, mes, occ, sbsf, sbpwf, sbpomf, sbprf, pombprf\}$*. For nets* $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$*,* $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ *such that* $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ *and SM-net* $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$ *the following holds:* $N \leftrightarrow_\star N' \ \Rightarrow\ ref(N, a, D) \leftrightarrow_\star ref(N', a, D)$ *iff equivalence* $\leftrightarrow_\star$ *is in oval in Figure 11.*

6. Investigation of the equivalences on sequential nets

Let us consider the influence of concurrency on interrelations of the equivalences.

Definition 18. A net $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ is *sequential*, if $\forall M \in Mark(N)$ $\neg\exists t, u \in T_N :\ {}^{\bullet}t + {}^{\bullet}u \subseteq M$, i.e. neither transitions are conurrently enabled in any reachable marking.

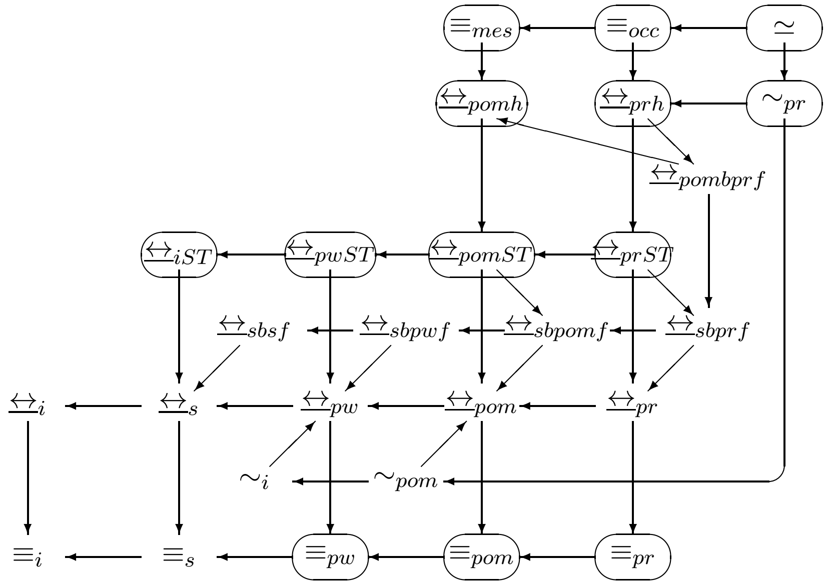

Figure 11. Preservation of the equivalences by SM-refinements

Proposition 5. *For sequential nets N and N' the following holds:*

1. $N \equiv_i N' \;\Leftrightarrow\; N \equiv_{pom} N'$;
2. $N \underline{\leftrightarrow}_i N' \;\Leftrightarrow\; N \underline{\leftrightarrow}_{pomh} N'$;
3. $N \underline{\leftrightarrow}_{pr} N' \;\Leftrightarrow\; N \underline{\leftrightarrow}_{pombprf} N'$;
4. $N \sim_i N' \;\Leftrightarrow\; N \sim_{pom} N'$.

In Figure 12 dashed lines embrace the equivalences coinciding on sequential nets.

Theorem 4. *Let $\leftrightarrow, \twoheadleftarrow\!\!\!\twoheadrightarrow \in \{\equiv, \underline{\leftrightarrow}, \sim, \simeq\}$, $\star, \star\star \in \{\_, i, pr, prST, prh, mes, occ\}$. For sequential nets N and N' the following holds: $N \leftrightarrow_\star N' \;\Rightarrow\; N \twoheadleftarrow\!\!\!\twoheadrightarrow_{\star\star} N'$ iff in the graph in Figure 13 there exists a directed path from $\leftrightarrow_\star$ to $\twoheadleftarrow\!\!\!\twoheadrightarrow_{\star\star}$.*

Proof. ($\Leftarrow$) By Theorem 2.

($\Rightarrow$) An absence of additional nontrivial arrows in the graph in Figure 13 is proved by the following examples on sequential nets.

- In Figure 4(d) $N \equiv_{mes} N'$, but $N \not\equiv_{pr} N'$.
- In Figure 4(e) $N \equiv_{pr} N'$, but $N \underline{\not\leftrightarrow}_i N'$.
- In Figure 6(c) $N \underline{\leftrightarrow}_{pr} N'$, but $N \underline{\not\leftrightarrow}_{prST} N'$, since only in the net N' the process with action a can start so that it can be extended by action b in the only way (i.e. so that extended process be unique).

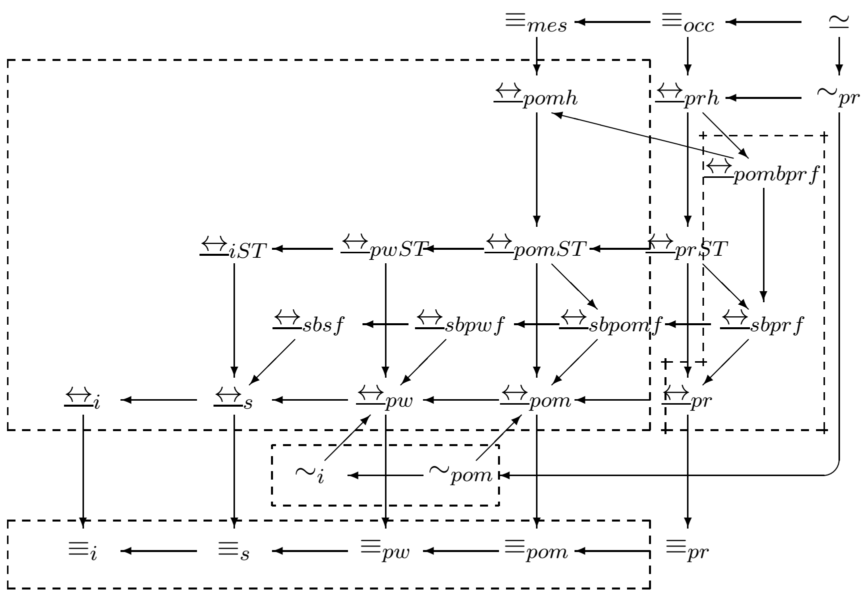

Figure 12. Merging of the equivalences on sequential nets

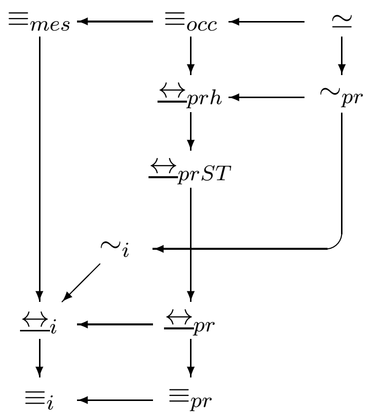

Figure 13. Interrelations of the equivalences on sequential nets

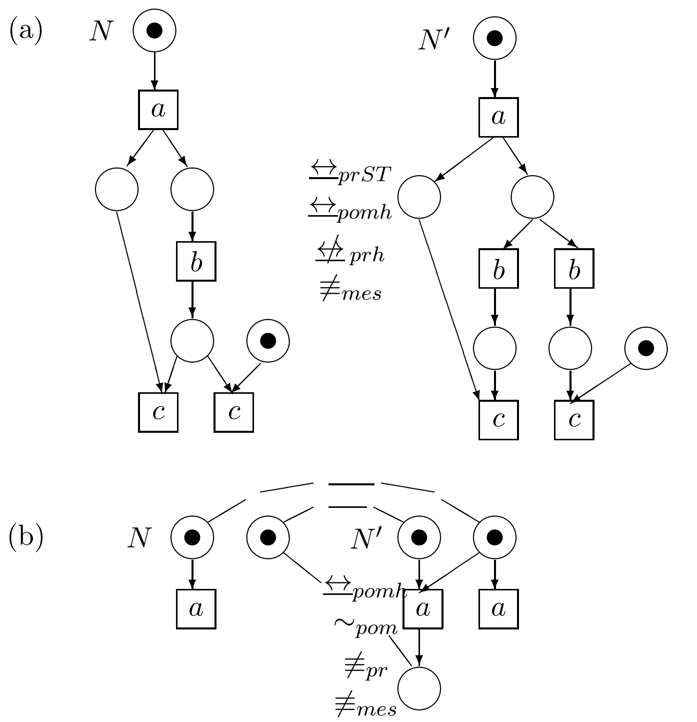

Figure 14. Examples of the equivalences on sequential nets

- In Figure 14(a) $N\underline{\leftrightarrow}_{prST}N'$, but $N\underline{\not\leftrightarrow}_{prh}N'$, since only in the net N' there is process with actions a and b s.t. it can be extended by process with action c in the only way. (i.e. so that connection of causal net with action c and a-containing subnet of causal net with actions a and b be unique).
- In Figure 5(c) $N\underline{\leftrightarrow}_{prh}N'$, but $N \not\equiv_{mes} N'$.
- In Figure 5(d) $N \equiv_{occ} N'$, but $N \not\simeq N'$.
- In Figure 14(b) $N \sim_i N'$, but $N \not\equiv_{pr} N'$, since the transition with label a has two input places only in the net N'.
- In Figure 10(c) $N \equiv_{occ} N'$, but $N \not\sim_i N'$.
- In Figure 5(c) $N \sim_{pr} N'$, but $N \not\equiv_{mes} N'$. □

7. Conclusion

In this paper, we examined and supplemented by new ones a group of back-forth and place bisimulation equivalences. We compared them with basic ones on the whole class of Petri nets as well as on their subclass of sequential nets. All the considered equivalences were treated for preservation by SM-refinements to establish which of them may be used for top-down design of concurrent systems.

Further research may consist in the investigation of analogs of the considered equivalences on Petri nets with τ-actions (τ-equivalences). τ-actions are used to abstract from internal, invisible to external observer behaviour of systems to be modelled. In the framework of Petri nets with τ-actions interrelations of equivalences are drastically changed.

For example, let us try to define τ-equivalences in process semantics. We abstract from τ-labelled transitions of C-nets by removing these transitions and multiplication of their input and output places. Then all causal dependencies of transitions with visible labels are preserved, and process τ-equivalences will imply corresponding pomset ones. But while such an abstraction the quantity of input and output places of some transitions with visible labels may be changed. The consequence is, in particular, that history preserving τ-bisimulation equivalences do not imply usual τ-bisimulation ones.

Therefore, it is no sence to introduce process τ-equivalences. By similar reasons, it is no sence to define strict place τ-bisimulation equivalences. In addition, multi event structure τ-equivalence does not imply even usual τ-bisimulation relations, but only τ-trace ones.

In the literature, a number of τ-equivalences were defined.

Some basic τ-equivalences were considered on Petri nets and event structures in [6, 22, 29]. It was shown the independence of ST- and history preserving τ-bisimulation equivalences.

In [14] interleaving back – interleaving forth τ-bisimuation equivalence was defined on transition systems. Its coincidence with interleaving branching τ-bisimuation equivalence was proved. Similar result was obtained in [23], where pomset back – pomset forth history preserving τ-bisimulation equivalence was introduced, and its merging with new notion of branching pomset history preserving τ-bisimulation equivalence was established.

In [5, 3] interleaving place τ-bisimulation and τp-bisimulation equivalences were introduced.

In future, we plan to define τ-analogs of all the equivalence relations considered in this paper and exam them following the same pattern.

Acknowledgements. I would like to thank Dr. Irina B. Virbitskaite for many helpful discussions. I am also grateful to Prof. Dr. Eike Best, head of the Institute of Informatics, University of Hildesheim, where this paper was written.

References

[1] C. Autant, Z. Belmesk, Ph. Schnoebelen: *Strong bisimularity on nets revisited.* Research Report 847-I, LIFIA-IMAG, Grenoble, France, 28 p., March 1991.

[2] C. Autant, Z. Belmesk, Ph. Schnoebelen: *Strong bisimularity on nets revisited. Extended abstract.* LNCS 506, p. 295–312, June 1991.

[3] C. Autant, W. Pfister, Ph. Schnoebelen: *Place bisimulations for the reduction of labelled Petri nets with silent moves.* Proceedings of International Conference on Computing and Information, 1994.

[4] C. Autant, Ph. Schnoebelen: *Place bisimulations in Petri nets.* LNCS 616, p. 45–61, June 1992.

[5] C. Autant: *Petri nets for the semantics and the implementation of parallel processes.* Ph.D. thesis, Institut National Polytechnique de Grenoble, May 1993 (in French).

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

[7] G. Boudol, I. Castellani: *On the semantics of concurrency: partial orders and transition systems.* LNCS 249, p. 123–137, 1987.

[8] F. Cherief: *Back and forth bisimulations on prime event structures.* LNCS 605, p.843–858, June 1992.

[9] F. Cherief: *Contributions à la sémantique du parallélisme: bisimulations pour le raffinement et le vrai parallélisme.* Ph.D. thesis, Institut National Politechnique de Grenoble, France, October 1992 (in French).

[10] F. Cherief: *Investigations of back and forth bisimulations on prime event structures.* Computers and Artificial Intelligence 11(5), p. 481–496, 1992.

[11] R.J. van Glabbeek, F.W. Vaandrager: *Petri net models for algebraic theories of concurrency.* LNCS 259, p. 224–242, 1987.

[12] J. Grabowski: *On partial languages.* Fundamenta Informaticae IV(2), p. 428–498, 1981.

[13] C.A.R. Hoare: *Communicating sequential processes, on the construction of programs.* (McKeag R.M., Macnaghten A.M., eds.) Cambridge University Press, p. 229–254, 1980.

[14] R. De Nicola, U. Montanari, F.W. Vaandrager: *Back and forth bisimulations.* LNCS 458, p. 152–165, 1990.

[15] M. Nielsen, G. Plotkin, G. Winskel: *Petri nets, event structures and domains.* TCS 13, p. 85–108, 1981.

[16] M. Nielsen, P.S. Thiagarajan: *Degrees of non-determinizm and concurrency: A Petri net view.* LNCS 181, p. 89–117, December 1984.

[17] E.-R. Olderog: *Nets, terms and formulas, three views of concurrent processes and their relationship.* Habilitationsschrift, Christian-Albrechts Universität, Kiel, July 1989.

[18] E.-R. Olderog: *Strong bisimularity on nets: a new concept for comparsing net semantics.* LNCS 354, p. 549–573, 1989.

[19] E.-R. Olderog: *Nets, terms and formulas.* Cambridge Tracts in Theoretical Computer Science 23, Cambridge University Press, 1991.

[20] D.M.R. Park: *Concurrency and automata on infinite sequences.* LNCS 104, p. 167–183, March 1981.

[21] C.A. Petri: *Kommunikation mit Automaten.* Ph.D. thesis, Universität Bonn, Schriften des Instituts für Instrumentelle Mathematik, 1962 (in German).

[22] L. Pomello: *Some equivalence notions for concurrent systems. An overview.* LNCS 222, p. 381–400, 1986.

[23] S. Pinchinat: *Bisimulations for the semantics of reactive systems.* Ph.D. thesis, Institut National Politechnique de Grenoble, January 1993 (in French).

[24] V.R. Pratt: *On the composition of processes.* Proceedings of 9^{th} POPL, p. 213–223, 1982.

[25] A. Rabinovitch, B.A. Trakhtenbrot: *Behaviour structures and nets.* Fundamenta Informaticae XI, p. 357–404, 1988.

[26] I.V. Tarasyuk: *An investigation of equivalence notions on some subclasses of Petri nets.* Bulletin of the Novosibirsk Computing Center (Series Computer Science) 3, p. 89–101, Computing Center, Novosibirsk, 1995.

[27] I.V. Tarasyuk: *Equivalence notions for design of concurrent systems using Petri nets.* Hildesheimer Informatik-Bericht 4/96, part 1, 19 p., Institut für Informatik, Universität Hildesheim, Hildesheim, Germany, January 1996.

[28] I.V. Tarasyuk: *Petri net equivalences for design of concurrent systems.* Proceedings of Workshop "Concurrency, Specification and Programming - 96", Informatik-Bericht 69, p. 190–204, Institut für Informatik, Humboldt-Universität zu Berlin, Berlin, Germany, 1996.

[29] W. Vogler: *Bisimulation and action refinement.* LNCS 480, p. 309–321, 1991.