Programming and Computer Software, Vol. 24, No. 4, 1998, pp. 162–175.

Equivalence Notions Applied to Designing Concurrent Systems with the Use of Petri Nets

I. V. Tarasyuk

Institute of Information Systems, Siberian Division, Russian Academy of Sciences, pr. Akademika Lavrent'eva 6, Novosibirsk, 630090 Russia

e-mail: itar@iis.nsk.ru

Received June 19, 1996

Abstract—This paper is dedicated to the study of behavioral equivalences of concurrent systems modeled by Petri nets. The main notions of equivalence known from the literature are complemented by new ones and analyzed on the whole class of Petri nets and on the subclass of sequential nets (nets without concurrency). A complete description of relationships between equivalences considered is obtained. Whether or not equivalence notions are preserved under the refinement operation, which makes it possible to consider the behavior of nets at a lower abstraction level, is also analyzed.

1. INTRODUCTION

A Petri net is a well-known formal model used for designing concurrent and distributed systems. As is well known, one of the main advantages of Petri nets is the possibility to structurally characterize basic aspects of concurrent computations: causal dependence, nondeterminism, and concurrency.

In recent years, a variety of semantic equivalences have been introduced in the concurrency theory. Many of them were either defined directly or were carried over from other models to Petri nets. The following basic equivalence notions for Petri nets are known from the literature:

(1) *Trace equivalences* (which take into account only protocols of net operation): interleaving equivalence [8], step equivalence [11], and equivalence of partially ordered multisets [7].

(2) *Ordinary bisimulation equivalences* (which take into account the branching structure of net operation): interleaving equivalence [10], step equivalence [9], equivalence of partial words [15], equivalence of partially ordered multisets [5], and process equivalence [2].

(3) *ST-bisimulation equivalences* (which take into account the duration of transition action in the net): interleaving equivalence [7], equivalence of partial words [15], and equivalence of partially ordered multisets [15].

(4) *History-preserving bisimulation equivalences* (which take into account "the past" (history) of net operation): the equivalence of partially ordered multisets was introduced [12].

(5) *Conflict preserving equivalences* (which take full account of conflicts in the net): the O-process equivalence was considered in [7].

(6) *Isomorphism*, i.e., the identity of nets up to renaming places and transitions.

When designing concurrent systems by the top-down method, the *refinement operator* is used that endows some of the net's elementary components with an internal structure; this makes it possible to consider such systems at a lower level of abstraction. In [4], the *SM-refinement* operator for Petri nets was suggested that changes their transitions for *SM*-nets, which constitute a specialized subclass of automata nets.

In this paper, we introduce a number of new notions in addition to known ones with the aim of obtaining a complete set of equivalences for Petri nets. These new notions are trace equivalences of partial words and processes, ST-bisimulation equivalence, history preserving process bisimulation equivalence, and the equivalence on multistructures of events. Relationships are established between new and known equivalence notions both on the whole class of Petri nets and on the subclass of sequential nets, where the firing of concurrent transition actions are not allowed. In addition, for all behavioral equivalences considered, it is verified whether they are preserved under SM-refinement.

The paper is organized as follows. In Section 2, basic definitions are given. In Section 3, behavioral equivalence notions are introduced. Section 4 is dedicated to the analysis of equivalences on the whole class of Petri nets, and Section 5 considers equivalences on the subclass of sequential nets. The invariance of equivalences under refinement is analyzed in Section 6. The final section, Section 7, contains a brief review of the results obtained and an outline of lines of further investigations.

0361-7688/98/2404-0162$20.00 © 1998 МАИК Наука / Interperiodica Publishing

2. BASIC DEFINITIONS

In this section, we give basic definitions used in the paper.

2.1. Multisets

Definition 2.1. *Let X be a set. A mapping* $M: X \longrightarrow \mathbf{N}$ *(where* $\mathbf{N}$ *is the set of non-negative integers) such that* $|\{x \in X | M(x) > 0\}| < \infty$ *is called a finite multiset M over X.*

Denote by $\mathcal{M}(X)$ *the set of all finite multisets over X*. If $\forall x \in X\ M(x) \le 1$, M is an ordinary set. The *cardinality* of a multiset M is defined as $|M| = \sum_{x \in X} M(x)$. We will write $x \in M$ if $M(x) > 0$ and $M_1 \subseteq M_2$ if $\forall x \in X\ M_1(x) \le M_2(x)$. Introduce the following definitions: $(M_1 + M_2)(x) = M_1(x) + M_2(x)$ and $(M_1 - M_2)(x) = \max\{0, M_1(x) - M_2(x)\}$.

2.2. Labeled nets

Definition 2.2. *Let* $Act = \{a, b, \ldots\}$ *be a set of actions or labels. A labeled net is a quadruple* $N = \langle P_N, T_N, F_N, l_N\rangle$*, where*

(1) $P_N = \{p, q, \ldots\}$ *is the set of places;*

(2) $T_N = \{u, v, \ldots\}$ *is the set of transitions;*

(3) $F_N: (P_N \times T_N) \cup (T_N \times P_N) \longrightarrow \mathbf{N}$ *is the incidence relation with weights (*$\mathbf{N}$ *denotes the set of non-negative integers); and*

(4) $l_N: T_N \longrightarrow Act$ *is a label of transition by actions.*

Let $N = \langle P_N, T_N, F_N, l_N\rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}\rangle$ be labeled nets. A mapping $\beta: N \longrightarrow N'$ is called *isomorphism* between N and N' (denoted by $\beta: N \simeq N'$) if β is a bijection such that $\beta(P_N) = P_{N'}$, $\beta(T_N) = T_{N'}$, $\forall p \in P_N\ \forall t \in T_N\ F_N(p, t) = F_{N'}(\beta(p), \beta(t))$, $F_N(t, p) = F_{N'}(\beta(t), \beta(p))$, and $\forall t \in T_N\ l_N(t) = l_{N'}(\beta(t))$. Labeled nets N and N' are called *isomorphic* (denoted by $N \simeq N'$) if $\exists \beta: N \simeq N'$.

For a labeled net N and its transition $t \in T_N$, define the *precondition* and *postcondition* of t (denoted by ${}^\bullet t$ and $t^\bullet$, respectively) as multisets $({}^\bullet t)(p) = F_N(p, t)$ and $(t^\bullet)(p) = F_N(t, p)$. Similar definitions are introduced for places: $({}^\bullet p)(t) = F_N(t, p)$ and $(p^\bullet)(t) = F_N(p, t)$. Denote by ${}^\circ N = \{p \in P_N | {}^\bullet p = \varnothing\}$ the set of *input* places of N and by $N^\circ = \{p \in P_N | p^\bullet = \varnothing\}$, the set of *output* places of N.

A labeled net N is called *acyclic* if there is no sequence $t_0, \ldots, t_n \in T_N$ such that $t_{i-1}^\bullet \cap {}^\bullet t_i \ne \varnothing$ $(1 \le i \le n)$ and $t_0 = t_n$. A labeled net N is called *ordinary* if $\forall p \in P_N$, ${}^\bullet p$ and $p^\bullet$ are ordinary sets (not multisets).

Let $N = \langle P_N, T_N, F_N, l_N\rangle$ be an acyclic ordinary labeled net, and $x, y \in P_N \cup T_N$. Introduce the following notions:

(1) $x \prec_N y \Leftrightarrow x F_N^+ y$, where F_N^+ is the transitive closure of F_N (the relation of *strict causal dependence*);

(2) $x \preceq_N y \Leftrightarrow (x \prec_N y) \vee (x = y)$ (the relation of *causal dependence*);

(3) $x \#_N y \Leftrightarrow \exists t, u \in T_N\ (t \ne u, {}^\bullet t \cap {}^\bullet u \ne \varnothing, t \preceq_N x, u \preceq_N y)$ (the relation of *conflict*);

(4) $\downarrow_N x = \{y \in P_N \cup T_N | y \prec_N x\}$ (the set of *strict predecessors* of x).

A set $T \subseteq T_N$ is *closed to the left* in N, if $\forall t \in T\ (\downarrow_N t) \cap T_N \subseteq T$. A set T is *conflict-free* in N if $\forall t, u \in T\ \neg(t \#_N u)$. A set T is a *configuration* in N if it is finite, closed to the left, and conflict-free in N.

2.3. Marked Nets

Definition 2.3. *A multiset* $M \in \mathcal{M}(P_N)$ *is called a marking of a labeled net N. A marked net (net) is a quintuple,* $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$*, where* $\langle P_N, T_N, F_N, l_N\rangle$ *is a labeled net and* $M_N \in \mathcal{M}(P_N)$ *is an initial marking.*

Let $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ be marked nets. A mapping $\beta: N \longrightarrow N'$ is called an *isomorphism* between N and N' (denoted by $\beta: N \simeq N'$) if $\beta: \langle P_N, T_N, F_N, l_N\rangle \simeq \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}\rangle$, and $\forall p \in M_N\ M_N(p) = M_{N'}(\beta(p))$. Nets N and N' are called *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 termed *feasible* in M if ${}^\bullet t \subseteq M$. If t is feasible in M, its action results in a new marking $\tilde{M} = M - {}^\bullet t + t^\bullet$, denoted by $M \xrightarrow{t} \tilde{M}$. A marking M of a net N is called *attainable* if $M = M_N$ or an attainable marking $\hat{M}$ of N exists such that $\hat{M} \xrightarrow{t} M$ for some $t \in T_N$. $Mark(N)$ denotes the set of *all attainable markings* of N.

2.4. Partially Ordered Sets

Definition 2.4. *A marked partially ordered set is a triple* $\rho = \langle X, \prec, l\rangle$*, where*

(1) $X = \{x, y, \ldots\}$ *is a set;*

(2) $\prec \subseteq X \times X$ *is a strict partial order (irreflexive transitive relation) on X;*

(3) $l: X \longrightarrow Act$ *is a marking function.*

Let $\rho = \langle X, \prec, l\rangle$ be a partially ordered set and $x \in X$. Then, $\downarrow x = \{y \in X | y \prec x\}$ is called the set of *strict predecessors* of x.

Let $\rho = \langle X, \prec, l\rangle$ and $\rho' = \langle X', \prec', l'\rangle$ be labeled partially ordered sets.

A mapping $\beta: X \longrightarrow X'$ is called a *label-preserving bijection* between ρ and ρ' (denoted by $\beta: \rho \approx \rho'$ if β is a bijection such that $\forall x \in X\ l(x) = l'(\beta(x))$. We will write $\rho \approx \rho'$ if $\exists \beta: \rho \approx \rho'$.

A mapping $\beta: X \longrightarrow X'$ is called a *homomorphism* between ρ and ρ' (denoted by $\beta: \rho \sqsubseteq \rho'$) if $\beta: \rho \approx \rho'$ and $\forall x, y \in X\ x \prec y \Rightarrow \beta(x) \prec' \beta(y)$. We will write $\rho \sqsubseteq \rho'$ if $\exists \beta: \rho \sqsubseteq \rho'$.

A mapping β: $X \longrightarrow X'$ is called an *isomorphism* between ρ and ρ' (denoted by β: $\rho \approx \rho'$ if β: $\rho \sqsubseteq \rho'$ and β^{-1}: $\rho \sqsubseteq \rho'$. Labeled partially ordered sets are called *isomorphic* (denoted by $\rho \approx \rho'$) if $\exists\beta$: $\rho \simeq \rho'$.

Definition 2.5. *A class of isomorphism of labeled partially ordered sets is called a partially ordered multiset.*

2.5. Structures of Events

Definition 2.6. *A labeled structure of events is a quadruple* $\xi = \langle X, \prec, \#, l\rangle$, *where*

(1) $X = \{x, y, \ldots\}$ *is a set of events;*

(2) $\prec \subseteq X \times X$ *is a strict partial order—a relation of causal dependence satisfying the principle of the finiteness of causes:* $\forall x \in X\ |{\downarrow}x| < \infty$*;*

(3) $\# \subseteq X \times X$ *is an irreflexive symmetric conflict relation satisfying the principle of the conflict inheritance:* $\forall x, y, z \in X\ x\#y \prec z \Rightarrow x\#z$*;*

(4) l: $X \longrightarrow Act$ *is a marking function.*

Let $\xi = \langle X, \prec, \#, l\rangle$ and $\xi' = \langle X', \prec', \#', l'\rangle$ be labeled structures of events. A mapping β: $X \longrightarrow X'$ is called an *isomorphism* between ξ and ξ' (denoted by β: $\xi \simeq \xi'$) if β: $\langle X, \prec, l\rangle \simeq \langle X', \prec', l'\rangle$ and $\forall x, y \in X\ x\#y \Leftrightarrow \beta(x)\#'\beta(y)$. Labeled structures of events ξ and ξ' are called *isomorphic* (denoted by $\xi \simeq \xi'$) if $\exists\beta$: $\xi \simeq \xi'$.

Definition 2.7. *A class of isomorphism of labeled structures of events is called a multistructure of events*.

2.5.1. C-processes. A C-process is a process based on a C-net [3].

Definition 2.8. *A* C-*net is an acyclic ordinary labeled net* $C = \langle P_C, T_C, F_C, l_C\rangle$ *such that*

1. $\forall r \in P_C\ |{}^\bullet r| \le 1$ and $|r^\bullet| \le 1$, *i.e., places do not branch;*

2. $|{\downarrow}_C x| < \infty$, *i.e., the set of causes is finite.*

Note that there is a labeled partially ordered set $\rho_C = \langle T_C, \prec_N \cap (T_C \times T_C), l_C\rangle$ corresponding to a C-net, $C = \langle P_C, T_C, F_C, l_C\rangle$. The following property is fundamental for C-nets [2]: if C is a C-net, then a sequence of transitions, $^\circ C = L_0 \xrightarrow{v_1} \ldots \xrightarrow{v_n} L_n = C^\circ$ exists such that $L_i \subseteq P_C$ $(0 \le i < n)$ $P_C = U_{i=0}^{n} L_i$, and $T_C = \{v_1, \ldots, v_n\}$. This sequence is called the *complete execution* of C.

Definition 2.9. *Let a net N and a* C-*net C be given. A mapping* φ: $P_C \cup T_C \longrightarrow P_N \cup T_N$ *is called an embedding of C into N (denoted by* φ: $C \longrightarrow N$*) if*

1. $\varphi(P_C) \in \mathcal{M}(P_N)$ *and* $\varphi(T_C) \in \mathcal{M}(T_N)$, *i.e., types of net elements are preserved;*

2. $\forall v \in T_C\ {}^\bullet\varphi(v) = \varphi({}^\bullet v)$ *and* $\varphi(v)^\bullet = \varphi(v^\bullet)$, *i.e., the incidence relation is taken into account; and*

3. $\forall v \in T_C\ l_C(v) = l_N(\varphi(v))$, *i.e., the label is preserved.*

In view of the fact that embedding takes account of the incidence relation, we find that, if $^\circ C \xrightarrow{v_1} \ldots \xrightarrow{v_n} C^\circ$ is a complete execution of C, then $M = \varphi(^\circ C) \xrightarrow{\varphi(v_1)} \ldots \xrightarrow{\varphi(v_n)} \varphi(C^\circ) = \tilde{M}$ is a sequence of transitions in N, which is denoted by $M \xrightarrow{C, \varphi} \tilde{M}$.

Definition 2.10. *A pair* $\pi = (C, \varphi)$, *where C is a* C-*net and* φ: $C \longrightarrow N$ *is an embedding such that* $M = \varphi(^\circ C)$, *is called the C-process (process) feasible in the marking M. A process feasible in* M_N *is called the process of N.*

Denote by $\Pi(N, M)$ the set of all processes feasible in the marking M of a set N and by $\Pi(N)$, the set of all processes of N. The *initial* process of N is a process $\pi_N = (C_N, \varphi_N) \in \Pi(N)$ such that $T_{C_N} = \varnothing$. If $\pi \in \Pi(N, M)$, then the execution of this process transforms the marking M into $\tilde{M} = M - \varphi(^\circ C) + \varphi(C^\circ) = \varphi(C^\circ)$ (denoted by $M \xrightarrow{\pi} \tilde{M}$).

Let $\pi = (C, \varphi)$, $\tilde{\pi} = (\tilde{C}, \tilde{\varphi}) \in \Pi(N)$, and $\hat{\pi} = (\hat{C}, \hat{\varphi}) \in \Pi(N, \varphi(C^\circ))$. The process π is called a *prefix* of the process $\tilde{\pi}$ if $T_C \subseteq T_{\tilde{C}}$ is a set in $\tilde{C}$ closed to the left. The process $\hat{\pi}$ is called a *suffix* of the process $\tilde{\pi}$ if $T_{\hat{C}} = T_{\tilde{C}} \backslash T_C$. Then, $\tilde{\pi}$ is an *extension of* π *to the process* $\hat{\pi}$, and $\hat{\pi}$ is an *extending* process for π, which is denoted by $\pi \xrightarrow{\hat{\pi}} \tilde{\pi}$. We will write $\pi \longrightarrow \tilde{\pi}$ if $\hat{\pi}$ such that $\exists\hat{\pi}\ \pi \xrightarrow{\hat{\pi}} \tilde{\pi}$.

A process $\tilde{\pi}$ is called an extension of π *for one operation* if $\pi \xrightarrow{v} \tilde{\pi}$, $T_{\hat{C}} = \{v\}$, and $l_{\hat{C}}(v) = a$. This is denoted by $\pi \xrightarrow{V} \tilde{\pi}$ or $\pi \xrightarrow{A} \tilde{\pi}$.

A process $\tilde{\pi}$ is called an extension of π *for a multiset of operations* or a *step* if $\pi \xrightarrow{A} \tilde{\pi}$, $\prec_{\hat{C}} = \varnothing$, $T_{\hat{C}} = V$, and $l_{\hat{C}}(V) = A$. This is denoted by $\pi \xrightarrow{V} \tilde{\pi}$ or $\pi \xrightarrow{A} \tilde{\pi}$.

2.5.2. O-processes.

An O-process is a process based on an O-net (branchy process in the terminology of [6]).

Definition 2.11. *An* O-*net is an acyclic ordinary labeled net* $O = \langle P_O, T_O, F_O, l_O\rangle$ *such that*

1. $\forall r \in P_O\ |{}^\bullet r| \le 1$, *i.e., there is no direct conflict;*

2. $\forall x \in P_O \cup T_O\ \neg(x\#_O x)$, *i.e., the conflict relation is irreflexive; and*

3. $\forall x \in P_O \cup T_O\ |{\downarrow}_O x| < \infty$, *i.e., the set of causes is finite.*

Note that, to any O-net, $O = \langle P_O, T_O, F_O, l_O\rangle$, the labeled structure of events $\xi_O = \langle T_O, \prec_O \cap (T_O \times T_O), \#_O \cap (T_O \times T_O), l_O\rangle$ can be assigned.

Definition 2.12. *Let* $O = \langle P_O, T_O, F_O, l_O\rangle$ *be an* O-*net and* $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ *be a net. A mapping* ψ: $P_O \cup T_O \longrightarrow P_N \cup T_N$ *is called an embedding of O into N (denoted by* ψ: $O \longrightarrow N$*) if*

1. $\psi(P_O) \in \mathcal{M}(P_N)$ *and* $\psi(T_O) \in \mathcal{M}(T_N)$, *i.e., types of nets elements are preserved;*

2. $\forall v \in T_O\ l_O(v) = l_N(\psi(v))$, *i.e., the label is preserved;*

3. $\forall v \in T_O\ {}^\bullet\psi(v) = \psi({}^\bullet v)$ *and* $\psi(v)^\bullet = \psi(v^\bullet)$, *i.e., the incidence relation is taken into account; and*

4. $\forall v, w \in T_O\ ({}^\bullet v = {}^\bullet w) \wedge (\psi(v) = \psi(w)) \Rightarrow v = w$, *i.e., there are no "extra" conflicts.*

Definition 2.13. *A pair* $\varpi = (O, \psi)$, *where O is an* O-*net and* ψ: $O \longrightarrow N$ *is an embedding such that* $M_N = \psi(°O)$, *is called an* O-*process of the net N.*

Denote the set of all O-*processes* of N by $\wp(N)$. The initial O-process of N is identical to the C-initial process; i.e., $\varpi_N = \pi_N$.

Let $\varpi = (O, \psi)$, $\tilde{\varpi} = (\tilde{O}, \tilde{\psi}) \in \wp(N)$. The O-process ϖ is called the *prefix* of the O-process $\tilde{\varpi}$ if $T_O \subseteq T_{\tilde{O}}$ is a set closed to the left in $\tilde{O}$. Then, $\tilde{\varpi}$ is called an *extension* of the O-process ϖ, which is denoted by $\varpi \longrightarrow \tilde{\varpi}$. Let $\pi = (C, \varphi) \in \Pi(N)$ and $\varpi = (O, \psi) \in \wp(N)$. A C-process ϖ is called an *evaluation* of the O-process $\tilde{\varpi}$ if $T_C \subseteq T_O$ is a configuration in O.

An O-process ϖ of a net N is called *maximal* if $\forall \tilde{\varpi} = (\tilde{O}, \tilde{\psi}) \in \wp(N)$ such that $\varpi \longrightarrow \tilde{\varpi}$, $T_{\tilde{O}} \backslash T_O = \varnothing$. Denote by $\wp_{\max}(N)$ the set of all *maximal* O-*processes* of N. Note that $\wp_{\max}(N)$ consists of the single O-process of the form $\varpi_{\max} = (O_{\max}, \psi_{\max})$. In this case, the class of isomorphism of the O-net, $O_{\max}$, is called a *development* of the net N, which is denoted by $\mathcal{U}(N)$. A multistructure of events, $\mathcal{E}(N) = \xi_{\mathcal{U}(N)}$, which is the class of isomorphism of the labeled structure of events ξ_O for $O \in \mathcal{U}(N)$, can be assigned to the development $\mathcal{U}(N)$.

3. EQUIVALENCE NOTIONS

In this section, we consider both equivalence notions for Petri nets that are known from the literature and new equivalence notions.

3.1. Trace Equivalences

Trace equivalences take into account only the protocols of the net operation and do not take into account the nondeterministic choice between several extensions of the process. For this reason, they are called equivalences of *linear time*.

Definition 3.1. *An interleaving trace of a net N is a sequence* $a_1 \ldots a_n \in Act^*$ *such that* $\pi_N \xrightarrow{a_1} \pi_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} \pi_n$, *where* $\pi_i \in \Pi(N)$ $(1 \le i \le n)$. *IntTraces(N) denotes the set of all interleaving traces of N. Nets N and N' are interleaving–trace equivalent (denoted by* $N \equiv_i N'$*) if IntTraces(N) = IntTraces(N').*

Definition 3.2. *A step trace of a net N is a sequence* $A_1 \ldots A_n \in (\mathcal{M}(Act))^*$ *such that* $\pi_N \xrightarrow{A_1} \pi_1 \xrightarrow{A_2} \ldots \xrightarrow{A_n} \pi_n$, *where* $\pi_i \in \Pi(N)$ $(0 \le i \le n)$. *StepTraces(N) denotes the set of all step traces of N. Nets N and N' are step–trace equivalent (denoted by* $N \equiv_s N'$*) if StepTraces(N) = StepTraces(N').*

Definition 3.3. *A partially ordered multiset trace of a net N is the partially ordered multiset* ρ *that is the class of isomorphism of the partially ordered multiset* ρ_C *for* $\pi = (C, \varphi) \in \Pi(N)$. *We write* $\rho \sqsubseteq \rho'$ *if* $\rho_C \sqsubseteq \rho_{C'}$, *where* $\rho_C \in \rho$ *and* $\rho_{C'} \in \rho'$. *In this case, the partially ordered multiset* ρ *is more parallel than* ρ'. *Pomsets(N) denotes the set of all partially ordered multiset traces of the net N. Nets N and N' are called trace equivalent on partial words (denoted by* $N \equiv_{pw} N'$ *if Pomsets*$(N) \sqsubseteq$ *Pomsets*(N') *and Pomsets*$(N') \sqsubseteq$ *Pomsets*(N); *i.e., for each* $\rho' \in$ *Pomsets*(N'), *a* $\rho \in$ *Pomsets*(N) *exists such that* $\rho \sqsubseteq \rho'$, *and vice versa.*

Definition 3.4. *Nets N and N' are called partially ordered multiset trace equivalent if Pomsets(N) = Pomsets(N'); this is denoted by* $N \equiv_{pom} N'$.

Definition 3.5. *The class of isomorphism of a* C-*net C for* $\pi = (C, \varphi) \in \Pi(N)$ *is called the process trace of N. ProcessNets(N) denotes the set of all process traces of N. Nets N and N' are called process–trace equivalent if ProcessNets(N) = ProcessNets(N'); this is denoted by* $N \equiv_{pr} N'$.

3.2. Ordinary Bisimulation Equivalences

Bisimulation equivalences take into account the moment of nondeterministic choice between several extensions of a process (branching). For this reason, they are called *branchy time* equivalences.

Definition 3.6. *A relation* $\mathcal{R} \subseteq \Pi(N) \times \Pi(N')$ *is called* $\star$-*bisimulation between N and N',* $\star \in$ *{interleaving, step, on partial words, on partially ordered multisets, process} (this fact is denoted by* $\mathcal{R}$: $N \underline{\leftrightarrow}_\star N'$, $\star \in \{i, s, pw, pom, pr\}$*) if*

1. $((\pi_N, \pi_{N'}) \in \mathcal{R}$;

2. $(\pi, \pi') \in \mathcal{R}$, $\pi \xrightarrow{\hat{\pi}} \tilde{\pi}$,

(a) $|T_{\hat{C}}| = 1$, *if* $\star = i$;

(b) $\prec_{\hat{C}} = \varnothing$, *if* $\star = s$;

$\Rightarrow \exists \tilde{\pi}' : \pi' \xrightarrow{\hat{\pi}'} \tilde{\pi}'$, $(\tilde{\pi}, \tilde{\pi}') \in \mathcal{R}$ *and*

(a) $\rho_{\hat{C}'} \sqsubseteq \rho_{\hat{C}}$, *if* $\star = pw$;

(b) $\rho_{\hat{C}} \simeq \rho_{\hat{C}'}$, *if* $\star = \{i, s, pom\}$;

(c) $\hat{C} \simeq \hat{C}'$, *if* $= pr$;

3. The same as item 2, but the roles of N and N' are interchanged.

Nets N and N' are $\star$-*bisimulation equivalent,* $\star \in$ *{interleaving, step, on partial words, on partially ordered multisets, process} if* $\exists\mathcal{R}$: $N \underline{\leftrightarrow}_\star N'$, $\star \in \{i, s, pw, pom, pr\}$; *this fact is denoted by* $N \underline{\leftrightarrow}_\star N'$.

3.3. ST-bisimulation Equivalences

To define ST-bisimulation equivalences, we introduce the notion of an ST-process that represents states of the net with nonzero time of transition actions.

Definition 3.7. *An ST-process of a net N is a pair* (π_E, π_P) *such that* $\pi_E, \pi_P \in \Pi(N)$, $\pi_P \xrightarrow{\pi_W} \pi_E$, *and* $\forall v, w \in T_{C_E}\ v \prec_{C_E} w \Rightarrow v \in T_{C_P}$.

In this case, π_E is the process that has started executing, i.e., all actions of π_E started executing. The process π_P corresponds to the part of π_E that has finished executing, and π_W corresponds to the part being executed. $ST - \Pi(N)$ denotes the set of all *ST-processes* of *N,* and (π_N, π_N) denotes the *initial ST-process* of *N*. Let $(\pi_E, \pi_P), (\tilde{\pi}_E, \tilde{\pi}_P) \in ST - \Pi(N)$. Then, we will write $(\pi_E, \pi_P) \longrightarrow (\tilde{\pi}_E, \tilde{\pi}_P)$ if $\pi_E \longrightarrow \tilde{\pi}_E$ and $\pi_P \longrightarrow \tilde{\pi}_P$.

ST-bisimulation equivalences take into account the duration of transition actions assuming that transitions that are executing at the moment take dibs from the input places but do not yet put them into the output places [7].

Definition 3.8. *A relation* $\mathcal{R} \subseteq ST - \Pi(N) \times ST - \Pi(N') \times \mathcal{B}$, *where* $\mathcal{B} = \{\beta \mid \beta\colon T_C \longrightarrow T_{C'}, \pi = (C, \varphi) \in \Pi(N), \pi' = (C', \varphi') \in \Pi(N')\}$, *is called an* ★*-ST-bisimulation between N and N' (*★ ∈ *{interleaving, on partial words, on partially ordered multisets, process}) (this is denoted by* $\mathcal{R}\colon N \underline{\leftrightarrow}_{\star ST} N'$, $\star \in \{i, pw, pom, pr\}$*) if*

1. $((\pi_N, \pi_N), (\pi_{N'}, \pi_{N'}), \emptyset) \in \mathcal{R}$;

2. $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R} \Rightarrow \beta\colon \rho_{C_E} \approx \rho_{C'_E}$ *and* $\beta(T_{C_P}) = T_{C'_P}$;

3. $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R}$, $(\pi_E, \pi_P) \longrightarrow (\tilde{\pi}_E, \tilde{\pi}_P) \Rightarrow \exists \tilde{\beta}, (\tilde{\pi}'_E, \tilde{\pi}'_P)\colon (\pi'_E, \pi'_P) \longrightarrow (\tilde{\pi}'_E, \tilde{\pi}'_P)$, $\tilde{\beta}|_{T_{C_E}} = \beta$, $((\tilde{\pi}_E, \tilde{\pi}_P), (\tilde{\pi}'_E, \tilde{\pi}'_P), \tilde{\beta}) \in \mathcal{R}$, and if $\pi_P \xrightarrow{\pi} \tilde{\pi}_E$, $\pi'_P \xrightarrow{\pi'} \tilde{\pi}'_E$, $\gamma = \tilde{\beta}|_{T_C}$, then

(a) $\gamma^{-1}\colon \rho_{C'} \sqsubseteq \rho_C$, *if* $\star = pw$;

(b) $\gamma\colon \rho_C \simeq \rho_{C'}$, *if* $\star \in \{pom, pr\}$;

(c) $C \simeq C'$, *if* $\star = pr$;

4. The same as item 3, but with the roles of N and N' being interchanged.

Nets N and N' are called ★*-ST-bisimulation equivalent,* ★ ∈ *{interleaving, on partial words, on partially ordered multisets, process}, if* $\exists \mathcal{R}\colon N \underline{\leftrightarrow}_{\star ST} N'$, $\star \in \{i, pw, pom, pr\}$*; this is denoted by* $N \underline{\leftrightarrow}_{\star ST} N'$.

3.4. History-Preserving Bisimulation Equivalences

History-preserving bisimulation equivalences take into account the past (history) of the net operation; i.e., modeling takes account of the part of the process whose execution results in the current state.

Definition 3.9. *A relation* $\mathcal{R} \subseteq \Pi(N) \times \Pi(N') \times \mathcal{B}$, *where* $\mathcal{B} = \{\beta \mid \beta\colon T_C \longrightarrow T_{C'}, \pi = (C, \varphi) \in \Pi(N), \pi' = (C', \varphi') \in \Pi(N')\}$, *is called a* ★*-history preserving bisimulation between N and N' (*★ ∈ *{on partially ordered multisets, process}) (this is denoted by* $\mathcal{R}\colon N \underline{\leftrightarrow}_{\star h} N'$, $\star \in \{pom, pr\}$*) if*

1. $(\pi_N, \pi_{N'}, \emptyset) \in \mathcal{R}$;

2. $(\pi, \pi', \beta) \in \mathcal{R} \Rightarrow$

(a) $\beta\colon \rho_C \simeq \rho_{C'}$ *if* $\star \in \{pom, pr\}$;

(b) $C \simeq C'$, *if* $\star = pr$;

3. $(\pi, \pi', \beta) \in \mathcal{R}$, $\pi \longrightarrow \tilde{\pi} \Rightarrow \exists \tilde{\beta}, \tilde{\pi}'\colon \pi' \longrightarrow \tilde{\pi}'$, $\tilde{\beta}|_{T_C} = \beta$, $(\tilde{\pi}, \tilde{\pi}', \tilde{\beta}) \in \mathcal{R}$;

4. The same as item 3, but with the roles of N and N' being interchanged.

Nets N and N' are called ★*-history preserving equivalent,* ★ ∈ *{on partially ordered multisets, process}, if* $\exists \mathcal{R}\colon N \underline{\leftrightarrow}_{\star h} N'$, $\star \in \{pom, pr\}$*; this is denoted by* $N \underline{\leftrightarrow}_{\star h} N'$.

3.4.1. Conflict preserving equivalences. Conflict preserving equivalences take full account of conflicts in nets.

Definition 3.10. *Nets N and N' are called multistructure of events equivalent if* $\mathcal{E}(N) = \mathcal{E}(N')$. *This is denoted by* $N \equiv_{mes} N'$.

Definition 3.11. *Nets N and N' are called* O*-process equivalent if* $\mathcal{U}(N) = \mathcal{U}(N')$. *This is denoted by* $N \equiv_{occ} N'$.

4. COMPARING EQUIVALENCES

In this section, we analyze relationships between equivalence notions on the whole class of Petri nets.

Theorem 4.1. *Let* $\leftrightarrow, \longleftrightarrow \in \{\equiv, \underline{\leftrightarrow}, \simeq\}$ *and* $\star, \star\star \in \{i, s, pw, pom, pr, iST, pwST, pomST, prST, pomh, prh, mes, occ\}$. *For nets N and N',* $N \leftrightarrow_{\star} N' \Rightarrow N \longleftrightarrow_{\star\star} N'$ *if and only if a directed route from* $\leftrightarrow_{\star}$ *to* $\longleftrightarrow_{\star\star}$ *in the graph depicted in Fig. 1 exists.*

Proof. ⇐ Let us verify that all implications in Fig. 1 are valid.

(1) Relationships between trace and interleaving equivalences are consequences of the fact that isomorphism of labeled partially ordered sets with the empty precedence relation is isomorphism of one-element labeled partially ordered sets.

(2) Relationships between partial words and step equivalences are consequences of the fact that homomorphism of labeled partially ordered sets is isomorphism of labeled partially ordered sets with the empty precedence relation.

(3) The relationship $\underline{\leftrightarrow}_{pwST} \longrightarrow \underline{\leftrightarrow}_{iST}$ is the consequence of the fact that homomorphism of labeled partially ordered sets is a label-preserving bijection.

(4) Relationships between equivalences on partially ordered multisets and on partial words are conse-

quences of the fact that isomorphism of labeled partially ordered sets is a homomorphism.

(5) Relationships between process equivalences and equivalences on partial words are consequences of the fact that labeled partially ordered sets based on C-isomorphic nets are also isomorphic.

(6) The relationship $\equiv_{occ} \longrightarrow \equiv_{mes}$ is valid because multistructures of events of isomorphic O-nets are identical.

(7) The relationship $\underline{\leftrightarrow}_i \longrightarrow \equiv_i$ can be verified as follows. Let $\mathcal{R}$: $N \underline{\leftrightarrow}_i N'$. If $\pi_N \xrightarrow{a_1} \pi_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} \pi_n$, then a sequence $(\pi_N, \pi_{N'}), (\pi_1, \pi_1'), \dots, (\pi_n, \pi_n') \in \mathcal{R}$ exists such that $\pi_{N'} \xrightarrow{a_1} \pi_1' \xrightarrow{a_2} \dots \xrightarrow{a_n} \pi_n'$, and vice versa, by virtue of the symmetry of bisimulation.

(8) The relationship $\underline{\leftrightarrow}_s \longrightarrow \equiv_s$ can be validated similarly to the previous item by using $A_1, \dots, A_n \in \mathcal{M}(Act)$ instead of $a_1, \dots, a_n \in Act$.

(9) The relationship $\underline{\leftrightarrow}_{pw} \longrightarrow \equiv_{pw}$can be proved as follows. Let $\mathcal{R}$: $N \underline{\leftrightarrow}_{pw} N'$ and ρ be the class of isomorphism ρ_C for $\pi = (C, \varphi) \in \Pi(N)$. Since $\pi_N \xrightarrow{\pi} \pi$, a pair $(\pi, \pi') \in \mathcal{R}$ exists such that $\pi' = (C', \varphi')$ and $\rho_{C'} \sqsubseteq \rho_C$. If ρ' is the class of isomorphism $\rho_{C'}$, then $\rho' \sqsubseteq \rho$. Hence, $Pomsets(N') \sqsubseteq Pomsets(N)$. The inclusion $Pomsets(N) \sqsubseteq Pomsets(N')$ can be proved similarly by using the symmetry of bisimulation.

(10) The relationship $\underline{\leftrightarrow}_{pom} \longrightarrow \equiv_{pom}$ can be proved as in the previous item by using isomorphism of labeled partially ordered sets instead of homomorphism.

(11) The relationship $\underline{\leftrightarrow}_{pr} \longrightarrow \equiv_{pr}$ can be proved as in the previous item by using process traces instead of partially ordered multiset traces and isomorphism of C-nets instead of the isomomorphism of their labeled partially ordered sets.

(12) The relationship $\underline{\leftrightarrow}_{\star ST} \longrightarrow \underline{\leftrightarrow}_{\star}$, $\star \in \{pw, pom, pr\}$ can be proved by constructing on the basis of the relation $\mathcal{R}$: $N \underline{\leftrightarrow}_{\star ST} N'$ a relation S: $N \underline{\leftrightarrow}_{\star} N'$ defined as $S = \{(\pi, \pi') | \exists\beta\ ((\pi, \pi), (\pi', \pi'), \beta) \in \mathcal{R}\}$.

(13) The relationship $\underline{\leftrightarrow}_{iST} \longrightarrow \underline{\leftrightarrow}_s$ can be validated in the same way as in the previous item taking into account the fact that the sequence of ST-processes $(\pi_0, \pi_0), \dots, (\pi_n, \pi_0), \dots, (\pi_n, \pi_n)$ such that $\pi = \pi_0 \xrightarrow{a_1} \dots \xrightarrow{a_n} \pi_n = \tilde{\pi}$ corresponds to the step $\pi \xrightarrow{A} \tilde{\pi}$, where $A = \{a_1, \dots, a_n\} \in \mathcal{M}(Act)$.

(14) The relationships $\underline{\leftrightarrow}_{\star h} \longrightarrow \underline{\leftrightarrow}_{\star ST}$, $\star \in \{pom, pr\}$ can be proved by constructing, on the basis of the relation $\mathcal{R}$: $N \underline{\leftrightarrow}_{\star h} N'$, a relation S: $N \underline{\leftrightarrow}_{\star ST} N'$ defined as $S = \{((\pi_E, \pi_P), (\pi_E', \pi_P'), \beta) | (\pi_E, \pi_E', \beta) \in \mathcal{R}, (\pi_E, \pi_P) \in ST - \Pi(N), (\pi_E', \pi_P') \in ST - \Pi(N'), \beta(T_{C_P}) = T_{C_P'}\}$.

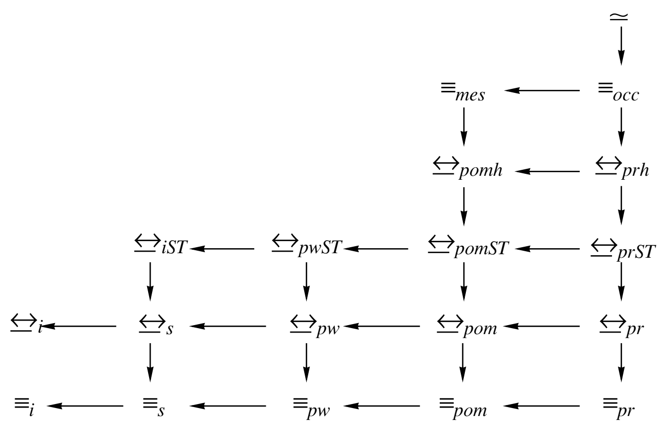

Fig. 1. Relationships between equivalences.

(15) The relationship $\equiv_{mes} \longrightarrow \underline{\leftrightarrow}_{pomh}$ is proved as follows. Let $\varpi = (O, \psi) \in \wp_{\max}(N)$, $\varpi' = (O', \psi') \in \wp_{\max}(N')$, γ: $\xi_O \simeq \xi_{O'}$. Then, $\mathcal{R}$: $N \underline{\leftrightarrow}_{pomh} N'$, where the relation $\mathcal{R}$ is defined as follows: $\mathcal{R} = \{(\pi, \pi', \beta) | \pi$ is a calculation of ϖ and π' is a calculation of ϖ' such that $\gamma|_{T_C}: \rho_C \simeq \rho_{C'}, \beta = \gamma|_{T_C}\}$.

(16) The relationship $\equiv_{occ} \longrightarrow \underline{\leftrightarrow}_{prh}$ is proved as follows. Let $\varpi = (O, \psi) \in \wp_{\max}(N)$, $\varpi' = (O', \psi') \in \wp_{\max}(N')$, γ: $O \simeq O'$. Then, $\mathcal{R}$: $N \underline{\leftrightarrow}_{prh} N'$, where the relation $\mathcal{R}$ is defined as follows: $\mathcal{R} = \{(\pi, \pi', \beta) | \pi$ is a calculation of ϖ and π' is a calculation of ϖ' such that $\gamma|_{(P_C \cup T_C)}: C \simeq C', \beta = \gamma|_{T_C}\}$.

(17) The relationship $\simeq \longrightarrow \equiv_{occ}$ is a consequence of the fact that isomorphic nets have identical developments.

$\Rightarrow$ The absence of additional nontrivial relationships in Fig. 1 can be proved by the following examples.

(1) In Fig. 2a, $N \underline{\leftrightarrow}_i N'$, because only in $N \not\equiv_s N'$ actions a and b cannot execute concurrently.

(2) In Fig. 2c, $N \underline{\leftrightarrow}_{iST} N'$, but $N \not\equiv_{pw} N'$, because a partially ordered multiset corresponds to N such that even the more parallel partially ordered multiset cannot execute in N'.

(3) In Fig. 2b, $N \underline{\leftrightarrow}_{pwST} N'$, but $N \not\equiv_{pom} N'$, because b can depend on a in N'.

(4) In Fig. 2d, $N \equiv_{mes} N'$, but $N \not\equiv_{pr} N'$, because the C-net N' is not isomorphic to the C-net N (due to the additional output place).

(5) In Fig. 2e, $N \equiv_{pr} N'$, but $N \not\underline{\leftrightarrow}_i N'$, because, only in N', can the action a be executed in such a way that prevents the action b from executing.

(6) In Fig. 3a, $N \underline{\leftrightarrow}_{pr} N'$, but $N \not\underline{\leftrightarrow}_{iST} N'$, because, in N', the action a can start executing in such a way that no action b can start until a terminates.

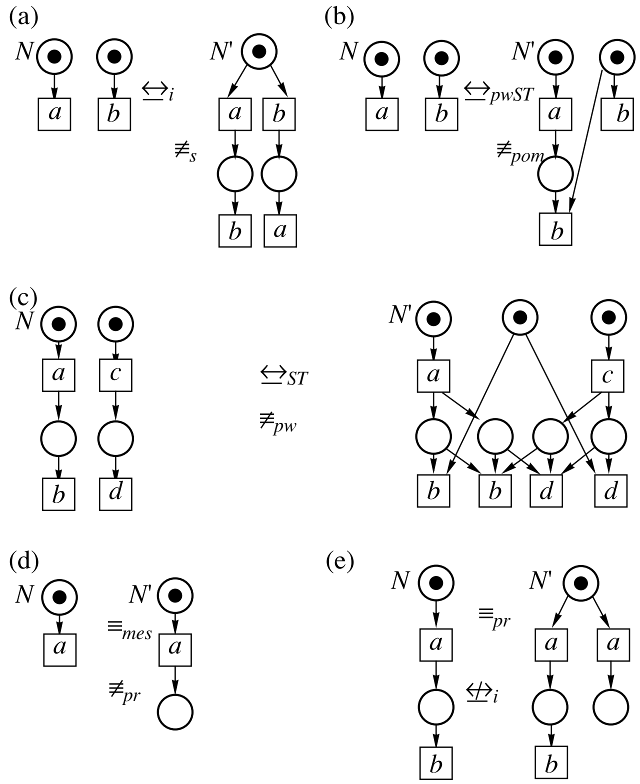

Fig. 2. Examples of equivalences.

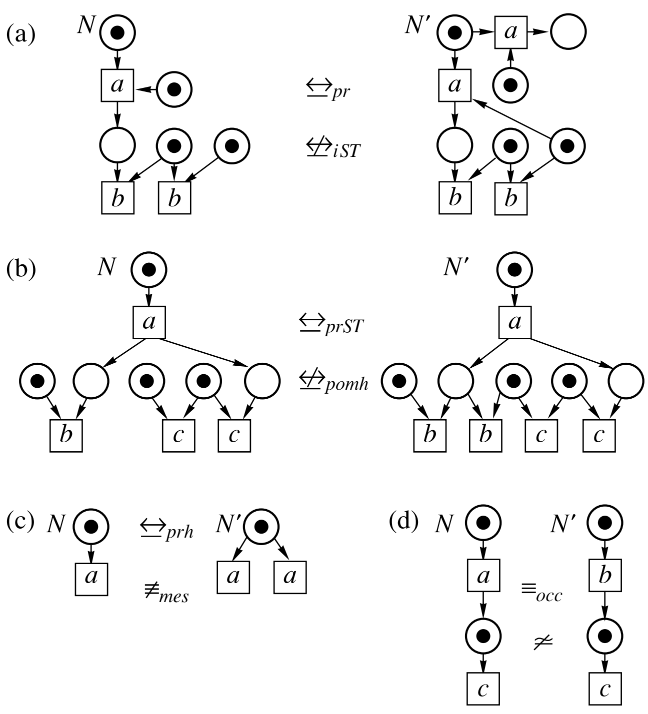

Fig. 3. Examples of equivalences (continuation).

(7) In Fig. 3b, $N \underline{\longleftrightarrow}_{prST} N'$, but $N \underline{\not\longleftrightarrow}_{pomh} N'$, because only in N', a and b can be executed in such a way that the next action c necessarily depends on a.

(8) In Fig. 3c, $N \underline{\longleftrightarrow}_{prh} N'$, but $N \not\equiv_{mes} N'$, because a labeled structure of events with two conflicting actions a corresponds to N'.

(9) In Fig. 3d, $N \equiv_{occ} N'$, but $N \not\simeq N'$, because never acting transitions of the nets N and N' are labeled by different actions (a and b).

□

5. COMPARING EQUIVALENCES ON SEQUENTIAL NETS

In this section, relationships between the equivalences introduced are analyzed on sequential nets, where concurrent transitions cannot act concurrently.

Definition 5.1. *A net* $N = \langle P_N, T_N, F_N, l_N, M_N \rangle$ *is called sequential if* $\forall M \in Mark(N)\ \neg\exists t, u \in T_N\colon {}^{\bullet}t + {}^{\bullet}u \subseteq M$*; i.e., no two transitions can be feasible together in any attainable marking*.

Proposition 5.1. For sequential nets N and N', the following holds:

1. $N \equiv_i N' \Leftrightarrow N \equiv_{pom} N''$; and
2. $N \underline{\longleftrightarrow}_i N' \Leftrightarrow N \underline{\longleftrightarrow}_{pomh} N'$.

Proof. 1. ⇐ By Theorem 4.1.

⇒ Let $N \equiv_i N'$; then, $IntTraces(N) = IntTraces(N')$. To prove that $N \equiv_{pom} N'$, it suffices to prove the equality $Pomsets(N) = Pomsets(N')$. It is obvious, since $Pomsets(N)$ and $Pomsets(N')$ are linearly ordered multisets (chains) and a one-to-one correspondence between $IntTraces(N)$ and $Pomsets(N)$ ($IntTraces(N')$ and $Pomsets(N')$, respectively, exists.

2. Can be proved by using Proposition 5.4 from [4].

Theorem 5.1. *Let* $\longleftrightarrow, \longleftrightarrow \in \{\equiv, \underline{\longleftrightarrow}, \simeq\}$ *and* $\star, \star\star \in \{i, pr, prST, prh, mes, occ\}$. *For sequential nets N and* $N \longleftrightarrow_{\star} N' \Rightarrow N \longleftrightarrow_{\star\star} N'$ *if and only if a directed route from* $\longleftrightarrow_{\star}$ *to* $\longleftrightarrow_{\star\star}$ ** *in the graph depicted in Fig. 4 exists.*

Proof. ⇐ By Theorem 4.1.

⇒ The absence of nontrivial relationships in Fig. 4 can be proved by the following examples on sequential nets.

(1) In Fig. 2d, $N \equiv_{mes} N'$, but $N \not\equiv_{pr} N'$.

(2) In Fig. 2e, $N \equiv_{pr} N'$, but $N \underline{\not\longleftrightarrow}_i N'$.

(3) In Fig. 5a, $N \underline{\longleftrightarrow}_{pr} N'$, but $N \underline{\not\longleftrightarrow}_{prST} N'$, because in N', a process c with the action a can start executing in such a way that it can be extended to the process with action b in only one way (i.e., the extended process is unique).

(4) In Fig. 5b, $N \underline{\longleftrightarrow}_{prST} N'$, but $N \underline{\not\longleftrightarrow}_{prh} N'$, because only in N', a process with actions a and b exists such that it can be extended to a process with action c in only one way (i.e., in such a way that there is only one kind of link of the C-net containing the action c with the subnet of the C-net with actions a and b that contains action a).

(5) In Fig. 3c, $N \underline{\longleftrightarrow}_{prh} N'$, but $N \not\equiv_{mes} N'$.

(6) In Fig. 3d, $N \equiv_{occ} N'$, but $N \neq N'$. □

6. PRESERVING EQUIVALENCES UNDER REFINEMENT

In this section, we verify whether the equivalence notions are preserved under the refinement operation, i.e., when going to a lower level of abstraction.

Definition 6.1. *A net $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$ is called an SM-net if*

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., D has a single input place and a single output place;

2. $M_D = \{p_{in}\}$ and $\forall M \in Mark(D)$ $(p_{out} \in M \Rightarrow M = \{p_{out}\})$; i.e., there is the single dib p_{in} at the beginning and the single dib p_{out} at the end;

3. $p_{in}^\bullet$ and $^\bullet p_{out}$ are ordinary sets (not multisets); i.e., p_{in} (respectively, p_{out}) represents the set of all dibs consumed (respectively, created) for any refined transition;

4. $\forall t \in T_D$ $|^\bullet t| = |t^\bullet| = 1$, i.e.; there is exactly one input and one output place.

Definition 6.2. *Let $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ be a net, $a \in l_N(T_N)$, and $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$ be a* SM-*net.* SM-*refinement is (up to isomorphism) a net, $\overline{N} = \langle P_{\overline{N}}, T_{\overline{N}}, F_{\overline{N}}, l_{\overline{N}}, M_{\overline{N}}\rangle$, denoted by ref(N, a, D), such that*

$$1.\ P_{\overline{N}} = P_N \cup \{\langle p, u\rangle | p \in P_D \backslash \{p_{in}, p_{out}\}, u \in l_N^{-1}(a)\};$$

$$2.\ T_{\overline{N}} = (T_N \backslash l_N^{-1}(a)) \cup \{\langle t, u\rangle | t \in T_D, u \in l_N^{-1}(a)\};$$

$$3.\ F_{\overline{N}}(\bar{x}, \bar{y})$$

$$= \begin{cases} F_N(\bar{x}, \bar{y}), & \bar{x}, \bar{y} \in P_N \cup (T_N \backslash l_N^{-1}(a)) \\ F_D(x, y), & \bar{x} = \langle x, u\rangle, \bar{y} = \langle y, u\rangle, u \in l_N^{-1}(a) \\ F_N(\bar{x}, u), & \bar{y} = \langle y, u\rangle, \bar{x} \in {}^\bullet u, u \in l_N^{-1}(a), y \in p_{in}^\bullet \\ F_N(u, \bar{y}), & \bar{x} = \langle x, u\rangle, \bar{y} \in {}^\bullet u, u \in l_N^{-1}(a), x \in {}^\bullet p_{out} \\ 0, & otherwise; \end{cases}$$

$$4.\ l_{\overline{N}}(\bar{u}) = \begin{cases} l_N(\bar{u}), & \bar{u} \in T_N \backslash l_N^{-1}(a) \\ l_D(t), & \bar{u} = \langle t, u\rangle, t \in T_D, u \in l_N^{-1}(a); \end{cases}$$

$$5.\ M_{\overline{N}}(p) = \begin{cases} M_N(p), & p \in P_N \\ 0, & otherwise. \end{cases}$$

A net equivalence is said to be *preserved under refinements* if equivalent nets remain equivalent after any refinement operator is simultaneously applied to them.

Proposition 6.1. *The equivalences $\equiv_\star$, $\star \in \{i, s\}$ and $\underleftrightarrow{\quad}_{\star\star}$, $\star\star \in \{i, s, pw, pom, pr\}$ are not preserved under* SM-*refinements.*

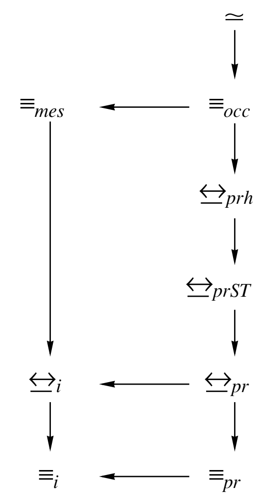

Fig. 4. Relationships between equivalences on sequential nets.

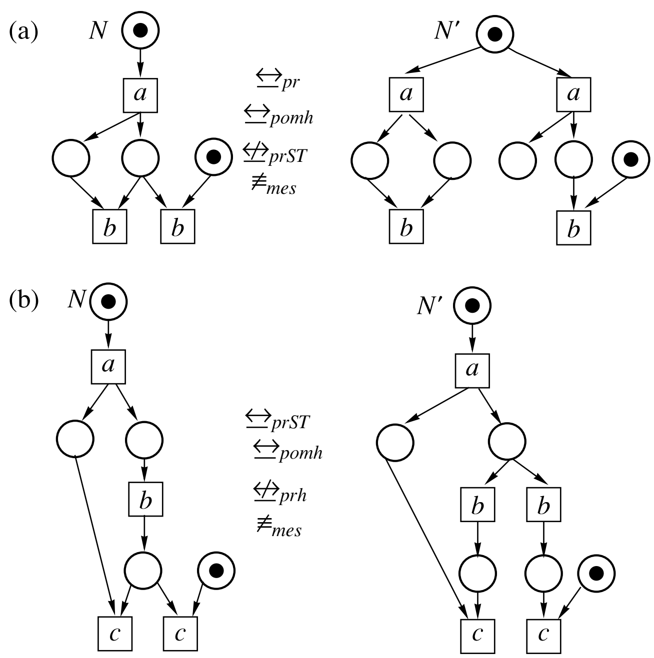

Fig. 5. Examples of equivalences on sequential nets.

Proof.

(1) In Fig. 6, $N \underleftrightarrow{\quad}_s N'$ but $ref(N, c, D) \not\equiv_i ref(N', c, D)$, because the set of actions c_1abc_2 can be executed only in $ref(N', c, D)$. Hence, no equivalences between $\equiv_i$ and $\underleftrightarrow{\quad}_s$ can be preserved under SM-refinements.

(2) In Fig. 7, $N \underleftrightarrow{\quad}_{pr} N'$ but $ref(N, a, D) \not\underleftrightarrow{\quad}_i ref(N', a, D)$, because only in $ref(N', a, D)$, the action b cannot be executed after a_1 has executed. Hence, no equivalences between $\underleftrightarrow{\quad}_i$ and $\underleftrightarrow{\quad}_{pr}$ can be preserved under SM-refinements.

Let us find which net equivalences are preserved under SM-refinements.

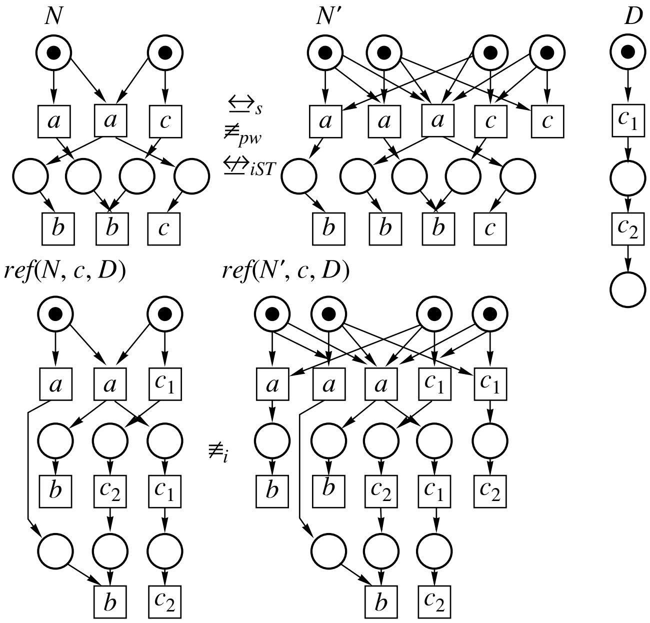

Fig. 6. Equivalences between $\equiv_i$ and $\underleftrightarrow{\quad}_s$ are not preserved under SM-refinements.

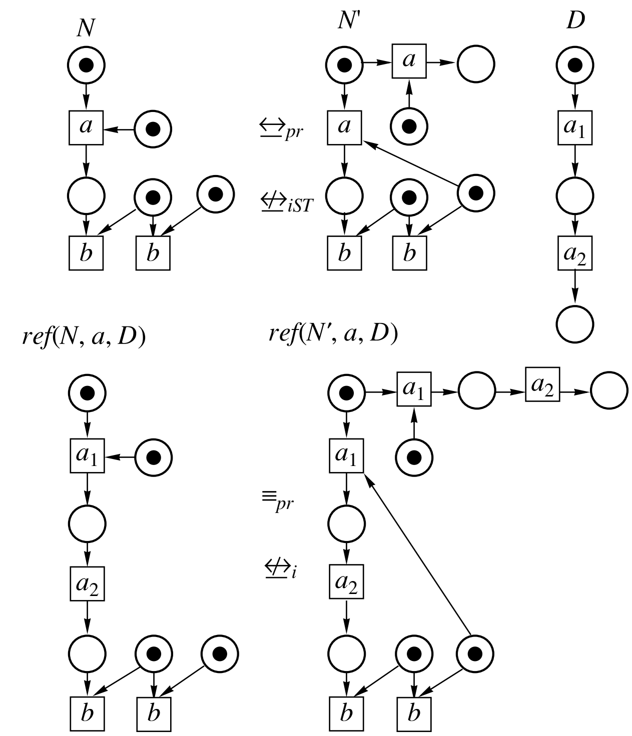

Fig. 7. Equivalences between $\underleftrightarrow{\quad}_i$ and $\underleftrightarrow{\quad}_{pr}$ are not preserved under SM-refinements.

Proposition 6.2. *Let* $\star \in \{pw, pom, pr\}$. *For the nets* $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ *and* $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ *such that* $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ *and for the* SM-*net* $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$ *the following holds:* $N \equiv_\star N' \Rightarrow ref(N, a, D) \equiv_\star ref(N', a, D)$.

Proof. Let $\overline{N} = ref(N, a, D)$ and $\overline{N}' = ref(N', a, D)$. Note that C-nets of the processes of SM-nets are simple chains, i.e., nets where each node has exactly one predecessor (except for the unique input place) and exactly one successor (except for the unique output place).

Construction (*)

1. Let $\bar{\pi} = (\overline{C}, \bar{\varphi}) \in \Pi(\overline{N})$. Then, any node $\overline{C}$ that is not mapped into the set $P_N \cup T_N$ by the embedding function possesses the following properties:

(1) it has the form $\langle e, f\rangle$, ($e \in P_{C_D} \cup T_{C_D}$, $\pi_D = (C_D, \varphi_D) \in \Pi(D)$ and $f \in T_C$, $\pi = (C, \varphi) \in \Pi(N)$), and is mapped by the embedding function to the element of the form $\langle x, u\rangle$, $x \in T_D \cup (P_D \backslash \{p_{in}, p_{out}\})$, $u \in l_N^{-1}(a)$;

(2) it has the single predecessor $\langle e_{\min}, f\rangle$ mapped by the embedding function to $\langle t_{\min}, u\rangle$, $t_{\min} \in p_{in}^\bullet$;

(3) it belongs to the single maximal chain ϑ (which corresponds to the net C_D) beginning at $\langle e_{\min}, f\rangle$, whose all nodes are mapped to elements of the form $\langle y, u\rangle$, $y \in T_D \cup (P_D \backslash \{p_{in}, p_{out}\})$, and whose only links with the process environment are as follows:

- through input places $\langle e_{\min}, f\rangle$ (always);
- (a) through output places of the maximal node of the chain $\langle e_{\max}, f\rangle$, which is the transition mapped to $\langle t_{\max}, u\rangle$ by the embedding function $t_{\max} \in {}^\bullet p_{out}$;

(b) or the chain terminates earlier at the maximal place.

Hence, any such chain ϑ included into net $\overline{C}$ can be changed for:

(a) the transition f mapped to u by the embedding function, because they have identical inputs and outputs;

(b) the transition f, mapped to u by the embedding function, with new output places corresponding to u, because they have identical inputs and there is nothing after f (in this case, f is the maximal transition).

As a result, we obtain the process $\pi = (C, \varphi) \in \Pi(N)$.

2. Since $N \equiv_\star N'$, $\star \in \{pw, pom, pr\}$, we can choose $\pi' = (C', \varphi') \in \Pi(N')$ and β such that

(1) β^{-1}: $\rho_{C'} \sqsubseteq \rho_C$, if $\star = pw$;

(2) β: $\rho_C \simeq \rho_{C'}$, if $\star = pom$;

(3) β: $C \simeq C'$, if $\star = pr$.

3. For any chain ϑ constructed in this way, change in C' the transition $\beta(f)$ embedded into u' for the copy ϑ' of ϑ, where all nodes of the form $\langle e, f\rangle$ are changed for $\langle e, \beta(f)\rangle$. The following two cases are possible:

(a) if the chain is complete, then $\beta(f)$ and ϑ' have identical outputs (from u');

(b) if the chain is incomplete (terminates at a place), then we discard all output places of $\beta(f)$.

In both cases, $\beta(f)$ and ϑ' have identical inputs (in u').

It is clear that the object constructed is the process $\bar{\pi}' = (\bar{C}', \bar{\varphi}') \in \Pi(\bar{N}')$.

4. Let $g \in T_{\bar{C}}$. Define the mapping $\bar{\beta}$ as follows:

$$\bar{\beta}(g) = \begin{cases} \beta(g), & g \text{ is included in none of the chains;} \\ \langle e, \beta(f)\rangle, & g = \langle e, f\rangle \text{ is included in chain } \vartheta. \end{cases}$$

□ (Construction (*))

It remains to prove the following assertions:

(1) $\bar{\beta}^{-1}: \rho_{\bar{C}'} \sqsubseteq \rho_{\bar{C}}$, if $\star = pw$;

(2) $\bar{\beta}: \rho_{\bar{C}} \simeq \rho_{\bar{C}'}$, if $\star = pom$;

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

Consider the case $\star = pw$ (the cases $\star = pom$ and $\star = pr$ are simpler). Let $g, h \in T_{\bar{C}}$. The following five cases are possible:

1. g and h are included in none of the chains;

2. g is included in a chain ϑ, and h is included in none of the chains;

3. g is included in none of the chains, and h is included in a chain ϑ;

4. g and h are included in one and the same chain ϑ;

5. g is included in a chain ϑ_1, h is included in a chain ϑ_2, and $\vartheta_1 \neq \vartheta_2$.

Consider case 5, since cases 1–4 are simpler. In this case, $g = \langle e_1, f_1\rangle$ and $h = \langle e_2, f_2\rangle$ where $e_1 \in T_{C_{D1}}$ and $e_2 \in T_{C_{D2}}$ for $\pi_{D1} = (C_{D1}, \varphi_{D1})$, $\pi_{D2} = (C_{D2}, \varphi_{D2}) \in \Pi(D)$, $f_1, f_2 \in T_C$ for $\pi = (C, \varphi) \in \Pi(N)$, f_1 and f_2 are refined in C into different chains ϑ_1 and ϑ_2, respectively. We have $\bar{\beta}(g) \prec_{\bar{C}'} \bar{\beta}(h) \Rightarrow \bar{\beta}(\langle e_1, f_1\rangle) \prec_{\bar{C}'} \bar{\beta}(\langle e_2, f_2\rangle) \Rightarrow$ (by definition of $\bar{\beta}$) $\langle e_1, \beta(f_1)\rangle \prec_{\bar{C}'} \langle e_2, \beta(f_2)\rangle \Rightarrow$ (because the chains are linked to the process environment only through their minimal and maximal transitions) $\langle e_{\max 1}, \beta(f_1)\rangle \prec_{\bar{C}'} \langle e_{\min 2}, \beta(f_2)\rangle \Rightarrow$ (by the construction (*)) $\beta(f_1) \prec_{C'} \beta(f_2) \Rightarrow$ (since $\beta^{-1}: \rho_{C'} \sqsubseteq \rho_C$) $f_1 \prec_C f_2 \Rightarrow$ (by the construction (*)) $\langle e_{\max 1}, f_1\rangle \prec_{\bar{C}} \langle e_{\min 2}, f_2\rangle \Rightarrow \langle e_1, f_1\rangle \prec_{\bar{C}} \langle e_2, f_2\rangle \Rightarrow g \prec_{\bar{C}} h$. □

Proposition 6.3. *Let $\star \in \{i, pw, pom, pr\}$. For nets $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ such that $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ and for* SM-*net $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$, the following holds: $N \underleftrightarrow{\quad}_{\star ST} N' \Rightarrow ref(N, a, D) \underleftrightarrow{\quad}_{\star ST} ref(N', a, D)$.*

Proof. Let $\bar{N} = ref(N, a, D)$, $\bar{N}' = ref(N', a, D)$ and $\mathcal{R}: N \underleftrightarrow{\quad}_{\star ST} N'$, $\star \in \{i, pw, pom, pr\}$.

Construction (**):

1. Let $(\bar{\pi}_E, \bar{\pi}_P) \in ST - \Pi(\bar{N})$ and $\pi_E, \pi_P \in \Pi(N)$ be obtained from $\bar{\pi}_E$ and $\bar{\pi}_P$ by using part 1 of the construction (*) from Proposition 6.2, respectively.

Lemma 1. $(\pi_E, \pi_P) \in ST - \Pi(N)$.

Proof of Lemma 1. Let $g, h \in T_{C_E}$ and $g \prec_{C_E} h$. The following four cases are possible:

(a) $l_{C_E}(g) \neq a \neq l_{C_E}(h)$;

(b) $l_{C_E}(g) = a \neq l_{C_E}(h)$;

(c) $l_{C_E}(g) \neq a = l_{C_E}(h)$;

(d) $l_{C_E}(g) = a = l_{C_E}(h)$.

Consider case (d), since cases (a)–(c) are simpler. In this case, g and h are refined in $\bar{C}_E$ into different chains ϑ_1 and ϑ_2 with nodes of the form $\langle e_1, g\rangle$ and $\langle e_2, h\rangle$, respectively, where $e_1 \in T_{C_{D1}}$ and $e_2 \in T_{C_{D2}}$ for $\pi_{D1} = (C_{D1}, \varphi_{D1})$, $\pi_{D2} = (C_{D2}, \varphi_{D2}) \in \Pi(D)$. We have $g \prec_{\bar{C}_E} h \Rightarrow$ (by the construction (*)) $\langle e_{\max 1}, g\rangle \prec_{\bar{C}_E} \langle e_{\min 2}, h\rangle \Rightarrow$ (since $(\bar{\pi}_E, \bar{\pi}_P) \in ST - \Pi(\bar{N})$, and $\langle e_{\min 2}, h\rangle \in T_{\bar{C}_E}$) $\langle e_{\max 1}, g\rangle \in T_{\bar{C}_P} \Rightarrow$ (by the construction (*)) $g \in T_{\bar{C}_P}$. □ (Lemma 1).

2. Choose $(\pi'_E, \pi'_P) \in ST - \Pi(N')$ and β such that $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R}$.

3. Obtain $\bar{\pi}'_E, \bar{\pi}'_P \in \Pi(\bar{N}'$ from $\bar{\pi}'_E$ and $\bar{\pi}'_P$, respectively, by using part 3 of the construction (*) from Proposition 6.2.

Lemma 2. $(\bar{\pi}'_E, \bar{\pi}'_P) \in ST - \Pi(\bar{N}')$.

Proof of Lemma 2. Let $g', h' \in T_{\bar{C}'_E}$ and $g' \prec_{\bar{C}'_E} h'$. The following five cases are possible:

(a) g' and h' are included in none of the chains;

(b) g' is included in a chain ϑ', and h' is included in none of the chains;

(c) g' is included in none of the chains, and h' is included in a chain ϑ';

(d) g' and h' are included in one and the same chain ϑ';

(e) g' is included in a chain ϑ'_1, h' is included in a chain ϑ'_2, and $\vartheta'_1 \neq \vartheta'_2$.

Consider case (e), since cases (a)–(d) are simpler. In this case, $g' = \langle e_1, f'_1\rangle$ and $h' = \langle e_2, f'_2\rangle$, where $e_1 \in T_{C_{D1}}$ and $e_2 \in T_{C_{D2}}$ for $\pi_{D1} = (C_{D1}, \varphi_{D1})$, $\pi_{D2} = (C_{D2}, \varphi_{D2}) \in \Pi(D)$, $f'_1, f'_2 \in T_{C'}$ for $\pi' = (C', \varphi') \in \Pi(N')$, f'_1 and f'_2 are refined in $\bar{C}''_E$ into different chains ϑ'_1 and ϑ'_2, respectively. We have $g' \prec_{\bar{C}'_E} h' \Rightarrow \langle e_1, f'_1\rangle \prec_{\bar{C}'_E} \langle e_2,$

$f'_2\rangle \Rightarrow$ (because the chains are linked to the process environment only through their minimal and maximal transitions) $\langle e_{\max 1}, f'_1\rangle \prec_{\bar{C}'_E} \langle e_{\min 2}, f'_2\rangle \Rightarrow$ (by the construction (*)) $f'_1 \prec_{C'_E} f'_2 \Rightarrow$ (since $(\pi'_E, \pi'_P) \in ST - \Pi(N')$ $f'_1 \in T_{C'_P} \Rightarrow$ (by the construction (*)) $g' = \langle e_1, f'_1\rangle \in T_{\bar{C}'_P}$. □ (Lemma 2).

4. Let $g \in T_{\bar{C}_E}$. Define the mapping $\bar{\beta}$ as follows:

$$\bar{\beta}(g) = \begin{cases} \beta(g), & \text{if } g \text{ included in none of the chains;} \\ \langle e, \beta(f)\rangle, & g = \langle e, f\rangle \text{ is included in a chain } \vartheta. \end{cases}$$

□ (Construction (**))

Let S consist of elements of the form $((\bar{\pi}_E, \bar{\pi}_P), (\bar{\pi}'_E, \bar{\pi}'_P), \bar{\beta})$, $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R}$. We prove that $S\colon \bar{N} \underset{\star ST}{\longleftrightarrow} \bar{N}'$.

1. Evidently, $((\pi_{\bar{N}}, \pi_{\bar{N}}), \pi_{\bar{N}'}, \pi_{\bar{N}'}, \varnothing) \in S$.

2. Let $((\bar{\pi}_E, \bar{\pi}_P), (\bar{\pi}'_E, \bar{\pi}'_P), \bar{\beta}) \in S$. It is apparent that by the construction (**) $\bar{\beta}\colon \rho_{\bar{C}_E} \simeq \rho_{\bar{C}'_E}$ and $\bar{\beta}(T_{\bar{C}_P}) = T_{\bar{C}'_P}$ because $\beta(T_{C_P}) = T_{C'_P}$.

3. Let $((\bar{\pi}_E, \bar{\pi}_P), (\bar{\pi}'_E, \bar{\pi}'_P), \bar{\beta}) \in S$ and $(\bar{\pi}_E, \bar{\pi}_P) \longrightarrow (\tilde{\bar{\pi}}_E, \tilde{\bar{\pi}}_P)$.

The element $((\bar{\pi}_E, \bar{\pi}_P), (\bar{\pi}'_E, \bar{\pi}'_P), \bar{\beta})$ is obtained by construction (**) from the element $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R}$. By part 1 of construction (**), we obtain $(\tilde{\pi}_E, \tilde{\pi}_P) \in ST - \Pi(N)$ from $(\tilde{\bar{\pi}}_E, \tilde{\bar{\pi}}_P)$. Evidently, $(\pi_E, \pi_P) \longrightarrow (\tilde{\pi}_E, \tilde{\pi}_P)$. Since $\mathcal{R}\colon N \underset{\star ST}{\longleftrightarrow} N'$, $\star \in \{i, pw, pom, pr\}$, we have that $\exists\tilde{\beta}$, $(\tilde{\pi}'_E, \tilde{\pi}'_P)$ such that $\tilde{\beta}|_{T_{C_E}} = \beta$, $(\pi'_E, \pi'_P) \longrightarrow (\tilde{\pi}'_E, \tilde{\pi}'_P)$, and $((\tilde{\pi}_E, \tilde{\pi}_P), (\tilde{\pi}'_E, \tilde{\pi}'_P), \tilde{\beta}) \in \mathcal{R}$. According to parts 3 and 4 of the construction (**), we obtain $(\tilde{\bar{\pi}}'_E, \tilde{\bar{\pi}}'_P) \in ST - \Pi(\bar{N})$ and $\tilde{\bar{\beta}}$ from $(\tilde{\pi}'_E, \tilde{\pi}'_P)$ and $\tilde{\beta}$, respectively.

Lemma 3. $((\tilde{\bar{\pi}}_E, \tilde{\bar{\pi}}_P), (\tilde{\bar{\pi}}'_E, \tilde{\bar{\pi}}'_P), \tilde{\bar{\beta}}) \in S$.

The Proof of Lemma 3 is evident from construction (**). □ (Lemma 3).

Lemma 4. $\tilde{\bar{\beta}}|_{T_{\bar{C}_E}} = \bar{\beta}$. □ (Lemma 4).

Proof of Lemma 4. Let $g \in T_{\bar{C}_E}$. Two cases are possible:

(a) g is included in none of the chains;

(b) g is included in a chain ϑ.

Consider case (b) (case (a) is trivial). In this case, $g = \langle e, f\rangle$, where $e \in T_{C_D}$ for $\pi_D = (C_D, \varphi_D)$, $f \in T_C$ for $\pi = (C, \varphi) \in \Pi(N)$, and f are refined into the chain ϑ in $\bar{C}_E$. We have $\tilde{\bar{\beta}}(\langle e, f\rangle) = \langle e, \tilde{\beta}(f)\rangle =$ (since $f \in T_{C_E}$ and $\tilde{\beta}|_{T_{C_E}} = \beta$, $\langle e, \beta(f)\rangle =$ (by definition of $\bar{\beta}$) $\bar{\beta}(\langle e, f\rangle)$. □ (Lemma 4).

Lemma 5. $(\bar{\pi}'_E, \bar{\pi}'_P) \longrightarrow (\tilde{\bar{\pi}}'_E, \tilde{\bar{\pi}}'_P)$.

Proof of Lemma 5. The proof follows from the fact that $(\pi'_E, \pi'_P) \longrightarrow (\tilde{\pi}_E', \tilde{\pi}_P)'$ and the construction (**). □ (Lemma 5).

Remark 1. Since $\tilde{\bar{\beta}}|_{\bar{C}_E} = \bar{\beta}$ by Lemma 4, and it follows from $(\bar{\pi}_E, \bar{\pi}_P) \in ST - \Pi(\bar{N})$ that $\bar{\beta}(T_{\bar{C}_P}) = T_{\bar{C}'_P}$, we have $\tilde{\bar{\beta}}(T_{\tilde{\bar{C}}_E} \backslash T_{\bar{C}_P}) = T_{\tilde{\bar{C}}'_E} \backslash T_{\tilde{\bar{C}}'_P}$. Hence, $\tilde{\bar{\beta}}(T_{\bar{C}}) = T_{\bar{C}'}$. □ (Remark 1).

Remark 2. Since it follows from $f \in T_{C_P}$ that $\langle e, f\rangle \in T_{\bar{C}_P}$, then $\langle e, f\rangle \notin T_{\bar{C}_P}$ implies $f \notin T_{C_P}$. Hence, $\langle e, f\rangle \in T_{\tilde{\bar{C}}_E} \backslash T_{\bar{C}_P} = T_{\bar{C}}$ implies $f \in T_{\tilde{C}_E} \backslash T_{C_P} = T_C$. □ (Remark 2).

It remains to prove the following propositions:

(1) $\tilde{\bar{\beta}}^{-1}\colon \rho_{\bar{C}'} \sqsubseteq \rho_{\bar{C}}$, if $\star = pw$;

(2) $\tilde{\bar{\beta}}\colon \rho_{\bar{C}} \simeq \rho_{\bar{C}'}$, if $\star = \{i, pom, pr\}$;

(3) $\bar{C} \simeq \bar{C}'$, if $\star = pr$.

Consider the case $\star = pr$, since the case $\star = pw$ can be considered as in Proposition 6.2, and the case $\star = pom$ is simpler. First, we should prove that $\tilde{\bar{\beta}}\colon \rho_{\bar{C}} \simeq \rho_{\bar{C}'}$. This can be done similarly to the proof of the case $\star = pw$ of Proposition 6.2 with all implications changed for symbols "equivalent."

Now we can prove that $\bar{C} \simeq \bar{C}'$. Since $\mathcal{R}\colon N \underset{prST}{\longleftrightarrow} N'$, we have $\exists\alpha\colon C \simeq C'$. Then, the mapping

$\bar{\alpha}: \bar{C} \simeq \bar{C}'$ can be obtained from α as follows. Let $g \in P_{\bar{C}} \cup T_{\bar{C}}$. Then,

$$\bar{\alpha}(g) = \begin{cases} \alpha(g), & \text{if } g \text{ is included in none of the chains} \\ \langle e, \alpha(f)\rangle, & g = \langle e, f\rangle \text{ is included in a chain } \vartheta. \end{cases}$$

4. The same as item 3, but the roles of $\bar{N}$ and $\bar{N}'$ are interchanged.

Proposition 6.4. [4] *For nets* $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ *and* $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ *such that* $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ *and for the* SM-*net* $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$, *the following is valid:* $N \underset{pomh}{\longleftrightarrow} N' \Rightarrow ref(N, a, D) \underset{pomh}{\longleftrightarrow} ref(N', a, D)$.

Proposition 6.5. *For nets* $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ *and* $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ *such that* $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ *and for the* SM-*net* $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$, *the following is valid:* $N \underset{prh}{\longleftrightarrow} N' \Rightarrow ref(N, a, D) \underset{prh}{\longleftrightarrow} ref(N', a, D)$.

Proof. It is sufficient to observe that the construction that transforms the bisimulation relation on original nets into the relation on refined nets (this construction was used in the proof of Proposition 8.5 in [4]) preserves isomorphism of C-nets of processes. □

Proposition 6.6. *Let* $\star \in \{mes, occ\}$. *For nets* $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ *and* $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ *such that* $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ *and for the* SM-*net* $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$, *the following is valid:* $N \equiv_\star N' \Rightarrow ref(N, a, D) \equiv_\star ref(N', a, D)$.

Proof. Let $\bar{N} = ref(N, a, D)$ and $\bar{N}' = ref(N', a, D)$. Note that O-nets of O-processes of SM-nets are trees, i.e., such nets where any node has exactly one predecessor (except for the unique input place).

Construction (***)

1. Let $\bar{\varpi} = (\bar{O}, \bar{\psi}) \in \wp_{\max}(\bar{N})$. Then, each node $\bar{O}$ that is not mapped by the embedding function into the set $P_N \cup T_N$ possesses the following properties:

(1) has the form $\langle e, f\rangle$ ($e \in P_{O_D} \cup T_{O_D}$, $\varpi_D = (O_D, \psi_D) \in \wp_{\max}(D)$, and $f \in T_O$, $\varpi = (O, \psi) \in \wp_{\max}(N)$) and is mapped by the embedding function to the element of the form $\langle x, u\rangle$, $x \in T_D \cup (P_D\backslash\{p_{in}, p_{out}\})$, $u \in l_N^{-1}(a)$;

(2) has the unique predecessor $\langle e^i_{\min}, f\rangle$ $(1 \le i \le n)$ that is mapped by the embedding function to $\langle t^i_{\min}, u\rangle$, $t^i_{\min} \in p^\bullet_{in}$;

(3) belongs to the unique maximal tree ϑ^i that is a member of the set of trees $\vartheta = \bigcup_{i=1}^n \vartheta^i$ corresponding to the net O_D, begins at $\langle e^i_{\min}, f\rangle$, all nodes of which are mapped by the embedding function to elements of the form $\langle y, u\rangle$, $y \in T_D \cup (P_D\backslash\{p_{in}, p_{out}\})$, and whose only links with the process environment are as follows:

(1) through input places $\langle e^i_{\max}, f\rangle$ (always);

(2) through output places of the maximal nodes of the tree $\langle e^{ij}_{\max}, f\rangle$ $(1 \le j \le m)$, that are the transitions mapped by the embedding function to $\langle t^{ij}_{\max}, u\rangle$, $t^{ij}_{\max} \in {}^\bullet p_{out}$.

Note that all $e^i_{\min}$ $(1 \le i \le n)$ have identical links with the process environment (as all $e^{ij}_{\min}$ $(1 \le i \le n, 1 \le j \le m)$). Hence, any such set ϑ included in the net $\bar{O}$ can be changed for the transition f that is mapped to u by the embedding function, because they have identical inputs and outputs. As a result, we obtain the O-process $\varpi = (O, \psi) \in \wp_{\max}(N)$.

2. Since $N \equiv_\star N'$, $\star \in \{mes, occ\}$, we can always choose $\varpi' = (O', \psi') \in \wp_{\max}(N')$ and β such that

(1) $\beta: \xi_O \simeq \xi_{O'}$, if $\star = mes$;

(2) $\beta: O \simeq O'$, if $\star = occ$.

3. For the ϑ constructed by the method described above, change in O' the transition $\beta(f)$ that can be embedded into u' for a copy ϑ' of the set of trees ϑ, where all nodes of the form $\langle e, f\rangle$ are changed for $\langle e, \beta(f)\rangle$. Then, $\beta(f)$ and ϑ' have the same outputs (from u') and inputs (to u').

It is clear that the constructed object is the O-process $\bar{\varpi}' = (\bar{O}', \bar{\psi}') \in \wp_{\max}(\bar{N}')$.

4. Let $g \in T_{\bar{O}}$. Define the mapping $\bar{\beta}$ as follows:

$$\bar{\beta}(g) = \begin{cases} \beta(g), & \text{if } g \text{ is included in none of the set of trees} \\ \langle e, \beta(f)\rangle, & g = \langle e, f\rangle \text{ is included in the set of trees } \vartheta. \end{cases}$$

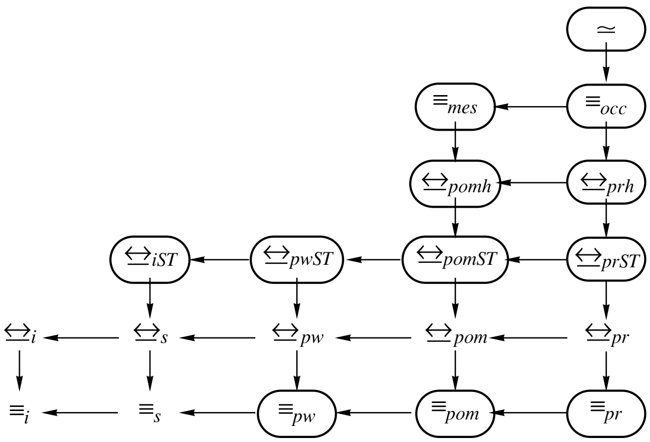

Fig. 8. Equivalence preserving under SM-refinements.

□ (Construction (***))

It remains to prove the following propositions:

(i) $\bar{\beta}: \xi_{\bar{O}} \simeq \xi_{\bar{O}'}$, if ★ = *mes*;

(ii) $\bar{\beta}: \bar{O} \simeq \bar{O}'$, if ★ = *occ*.

Consider the case ★ = *mes*, since the case ★ = *occ* is simpler. Let $g, h \in T_{\bar{O}}$. The following five cases are possible:

1. g and h are included in none of the sets of trees;

2. g is included in a set of trees ϑ, and h is included in none of the sets of trees;

3. g is included in none of the sets of trees, and h is included in a set of trees ϑ;

4. g and h are included in one and the same set of trees ϑ;

5. g is included in a set of trees ϑ_1, h is included in a set of trees ϑ_2, and $\vartheta_1 \neq \vartheta_2$.

Consider case 5, since cases 1–4 are simpler. In this case, $g = \langle e_1, f_1\rangle$ and $h = \langle e_2, f_2\rangle$, where $e_1 \in T_{O_{D1}}$ and $e_2 \in T_{O_{D2}}$ for $\varpi_{D1} = (O_{D1}, \psi_{D1})$, $\varpi_{D2} = (O_{D2}, \psi_{D2}) \in \wp(D)$, $f_1, f_2 \in T_O$ for $\varpi = (O, \psi) \in \wp(N)$, f_1 and f_2 are refined in O into different sets of trees ϑ_1 and ϑ_2, respectively. We prove that the precedence and conflict relations are preserved.

(1) $g \prec_{\bar{O}} h \Leftrightarrow \langle e_1, f_1\rangle \prec_{\bar{O}} \langle e_2, f_2\rangle \Leftrightarrow$ (because ϑ_1 and ϑ_2 are linked to the process environment only through their minimal and maximal transitions, and all minimal (maximal) transitions have identical links to this environment) $\forall i, j, k\ \langle e^{ij}_{\max 1}, f_1\rangle \prec_{\bar{O}} \langle e^{k}_{\min 2}, f_2\rangle \Leftrightarrow$ (by construction (***)) $f_1 \prec_O f_2 \Leftrightarrow$ (because β: $\xi_O \simeq \xi_{O'}$) $\beta(f_1) \prec_{O'} \beta(f_2) \Leftrightarrow$ (by construction (***)) $\forall i, j, k$ $\langle e^{ij}_{\max 1}, \beta(f_1)\rangle \prec_{\bar{O}'} \langle e^{k}_{\min 2}, \beta(f_2)\rangle \Leftrightarrow \langle e_1, \beta(f_1)\rangle \prec_{\bar{O}'} \langle e_2, \beta(f_2)\rangle \Leftrightarrow$ (by definition of $\bar{\beta}$ ($\bar{\beta}\langle e_1, f_1\rangle \prec_{\bar{O}'} \bar{\beta}(\langle e_2, f_2\rangle) \Leftrightarrow \bar{\beta}(y) \prec_{\bar{O}'} \bar{\beta}(h)$.

(2) $g \#_{\bar{O}} h \Leftrightarrow \langle e_1, f_1\rangle \#_{\bar{O}} \langle e_2, f_2\rangle \Leftrightarrow$ (because ϑ_1 and ϑ_2 are linked to the process environment only through their minimal and maximal transitions, and all minimal (maximal) transitions have identical links to this environment) $\forall i, k\ \langle e^{i}_{\min 1}, f_1\rangle \#_{\bar{O}} \langle e^{k}_{\min 2}, f_2\rangle \Leftrightarrow$ (by construction (***)) $f_1 \#_O f_2 \Leftrightarrow$ (because β: $\xi_O \simeq \xi_{O'}$) $\beta(f_1) \#_{O'} \beta(f_2) \Leftrightarrow$ (by construction (***)) $\forall i, k\ \langle e^{i}_{\min 1}, \beta(f_1)\rangle \#_{\bar{O}'} \langle e^{k}_{\min 2}, \beta(f_2)\rangle \Leftrightarrow \langle e_1, \beta(f_1)\rangle \#_{\bar{O}'} \langle e_2, \beta(f_2)\rangle \Leftrightarrow$ (by definition of $\bar{\beta}$) $\bar{\beta}(\langle e_1, f_1\rangle) \#_{\bar{O}'} \bar{\beta}(\langle e_2, f_2\rangle) \Leftrightarrow \bar{\beta}(g) \#_{\bar{O}'} \bar{\beta}(h)$. □

Proposition 6.7. *For nets $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ such that $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ and for the* SM-*net $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$, the following is valid: $N \simeq N' \Rightarrow ref(N, a, D) \simeq ref(N', a, D)$.*

Proof. The proof is evident. □

Theorem 6.1. *Let $\longleftrightarrow \in \{\equiv, \underline{\leftrightarrow}, \simeq\}$ and $\star \in \{i, s, pw, pom, pr, iST, pwST, pomST, prST, pomh, prh, mes, occ\}$. For nets $N = \langle P_N, T_N, F_N, l_N, M_N\rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'}\rangle$ such that $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ and for the* SM-*net $D = \langle P_D, T_D, F_D, l_D, M_D\rangle$, the following is valid: $N \longleftrightarrow_\star N' \Rightarrow ref(N, a, D) \longleftrightarrow_\star ref(N', a, D)$ if and only if the equivalence $\longleftrightarrow_\star$ is enclosed in an oval in Fig. 8.*□

Proof. Can be performed by using Propositions 6.1–6.7.

7. CONCLUSION

In this paper, a group of basic behavioral equivalences is analyzed and augmented by new notions that can be used to analyze systems modeled by Petri nets at various levels of abstraction.

The main result of the paper consists in establishing relationships between equivalence notions both on the whole class of Petri nets and on the subclass of sequential nets. For all equivalences considered, it is verified whether they are preserved under SM-refinements. Thus, we can use equivalence notions that are preserved under SM-refinements for designing concurrent systems by the top-down method.

Let us describe some lines of further investigations.

One such line is aimed at obtaining the complete description of relationships between equivalences on strictly labeled nets (all transitions are labeled by different actions) and on T-nets (without conflicting transitions). Note that the author has proved that some equivalences are identical on these two subclasses of Petri nets [13, 14].

We also intend to extend the field of investigations to the nets with τ-transitions (that are labeled by invisible τ-actions). Since this is a wider class of nets, it is possible that some relationships between equivalences cease to exist on this class. For example, it has been shown in [15] that ST-equivalences and history-preserving equivalences are independent on the structures of events with τ-actions.

Another line of investigation consists in the analysis of bisimulation equivalences of places [2]. We are going to compare them with equivalence relations considered in this paper (for example, relationships between bisimulation place equivalences and ST bisimilation history-preserving equivalences are not known to date). Besides, it is reasonable to verify whether bisimulation place equivalences are preserved under refinements in order to find out whether it is possible to use them for designing multilevel concurrent systems.

ACKNOWLEDGMENTS

This work was supported by the Volkswagen Foundation, grant no. I/70 564, and by the Russian Foundation for Basic Research, project no. 96-01-01655.

I am grateful to my scientific advisor Irina B. Virbitskaite for her important advice and many suggestions.

This paper was written while the author worked at the Institut für Informatik in Hildesheim, Germany. I am grateful to the Institute Director, Aike Best, for the creation of a favorable working atmosphere, as well as for helpful discussions, notes, and questions. I also thank the Institute's researcher, Tom Tilke, for advice that helped to improve the first version of this paper.

REFERENCES

1. Autant, C., Pfister, W., and Schnoebelen, Ph., Place Bisimulations for the Reduction of Labeled Petri Nets with Silent Moves, *Proc. Int. Conf. on Comput. and Inf.*, 1994.
2. Autant, C. and Schnoebelen, Ph., Place Bisimulations in Petri Nets, *LNCS*, 1992, vol. 616, pp. 45–61.
3. Best, E. and Devillers, R., Sequential and Concurrent Behavior in Petri Net Theory, *TCS*, 1987, vol. 55, pp. 87–136.
4. Best, E., Devillers, R., Kiehn, A., and Pomello, L., Concurrent Bisimulations in Petri Nets, *Acta Inf.*, 1991, vol. 28, pp. 231–264.
5. Boudol, G. and Castellani, I., On the Semantics of Concurrency: Partial Orders and Transition Systems, *LNCS*, 1987, vol. 249, pp. 123–137.
6. Engelfriet, J., Branching Processes Petri Nets, *Acta Inf.*, 1991, vol. 28, no. 6, pp. 575–591.
7. Van Glabbeek, R.J. and Vaandrager, F.W., Petri Net Models for Algebraic Theories of Concurrency, *LNCS*, 1987, vol. 259, pp. 224–242.
8. Hoare, C.A.R., On the Construction of Programs, in *Communicating Sequential Processes*, McKeag, R.M. and Macnaghten, A.M., Eds., Cambridge: Cambridge Univ. Press, 1980, pp. 229–254.
9. Nielsen, M. and Thiagarajan, P.S., Degrees of Non-Determinism and Concurrency: A Petri Net View, *LNCS*, 1984, vol. 181, pp. 89–117.
10. Park, D.M.R., Concurrency and Automata on Infinite Sequences, *LNCS*, 1981, vol. 104, pp. 167–183.
11. Pomello, L., Some Equivalence Notions for Concurrent Systems: An Overview, *LNCS*, 1986, vol. 222, pp. 381–400.
12. Rabinovitch, A. and Trakhtenbrot, B.A., Behavior Structures and Nets, *Fundam. Inf.*, 1988, vol. XI, pp. 357–404.
13. Tarasyuk, I.V., An Investigation of Equivalence Notions on Some Subclasses of Petri Nets, *Bulletin of Novosibirsk Computing Center (Comput. Sci. Series)*, 1995, vol. 3, pp. 89–101.
14. Tarasyuk, I.V., Equivalence Notions for Design of Concurrent Systems Using Petri Nets, *Hildesheimer Informatik-Bericht*, 1996, vol. 4, part 1.
15. Vogler, W., Bisimulation and Action Refinement, *LNCS*, 1991, vol. 480, pp. 309–321.