Bull. Nov. Comp. Center, Comp. Science, 3 (1995), 89–100

An investigation of equivalence notions on some subclasses of Petri nets*

I.V. Tarasyuk

In this paper a variety of Petri net equivalences is examined. A correlation of all the considered equivalences is established, and a lattice of implications is obtained. In addition, the equivalences are treated for some subclasses of Petri nets: sequential nets, T-nets and nets with strict labelling.

1. Introduction

In recent years, a wide range of semantic equivalences were defined and investigated in concurrency theory. In linear time semantics, where a process is fully determined by the set of its possible (partial) runs, interleaving, step and pomset trace equivalences [3] are known.

In branching time semantics the information is preserved where two courses of actions diverge. Bisimulation is a fundamental behavioural equivalence in this semantics. Interleaving [6], step [5], partial word [11], pomset [4] and process [1] bisimulation equivalences were proposed in the literature.

(Interleaving) ST-bisimulation equivalence [4] respects the duration of transition occurrences. A definition of the equivalence was extended to partial words and pomsets in [11].

(Pomset) history preserving bisimulation equivalence, which respects the "past" of the processes, was first defined in [8] under the name "bisimulation equivalence of behaviour structures".

In this paper the above mentioned definitions are supplemented by partial word history preserving and by process (ST- and history preserving) bisimulation equivalences. The equivalences are considered in the framework of Petri nets with finite processes. A correlation of all the equivalences is examined on usual Petri nets and their subclasses: sequential nets, T-nets and strictly labelled nets.

In Section 2 the basic definitions are given. Trace equivalences are described in Section 3. Bisimulation equivalences are presented in Section 4. In Section 5 the theorem establishing a correlation of all the introduced equivalences is proved. Section 6 is devoted to the examination of the equivalences on different net subclasses. The concluding Section 7 contains some ideas about further development of the theme. Most of the proofs are omitted because of absence of space. The early results can be found in [9].

*The work was supported in part by International Association for the Promotion of Cooperation with Scientists from the Independent States of the Former Soviet Union (INTAS), contract No. 1010–CT93–0048.

2. Basic definitions

2.1. Multisets

Let X be some set. A *multiset* M over X is a mapping $M : X \to \mathbf{N}$, where $\mathbf{N}$ is a set of natural numbers. For $x \in X$, $M(x)$ is a *multiplicity* x in M. We write $x \in M$ if $M(x) > 0$.

When $\forall x \in X \; M(x) \leq 1$, M is a proper set. M is *finite* if $M(x) = 0$ for all $x \in X$, except maybe a finite number of them. *Cardinality* of multiset M is defined in such a way: $|M| = \sum_{x \in X} M(x)$. From now on we will consider only finite multisets. $\mathcal{M}(X)$ denotes the *set of all finite multisets* over X.

Set-theoretic notions are extended to finite multisets in the standard way. If $M, M' \in \mathcal{M}(X)$, we define $M + M'$ by $(M + M')(x) = M(x) + M'(x)$. We write $M \subseteq M'$, if $\forall x \in X \; M(x) \leq M'(x)$. When $M' \subseteq M$, we define $M - M'$ by $(M - M')(x) = M(x) - M'(x)$. Notation $M + x - y$ is used instead of $M + \{x\} - \{y\}$. We write symbol $\emptyset$ for empty multiset.

2.2. Marked nets

Let $\mathfrak{A} = \{a, b, \ldots\}$ be an alphabet of action names (labels) 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;
- $T_N = \{u, v, \ldots\}$ 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;
- $l_N : T_N \to \mathfrak{A}$ is a labelling of transitions with action names.

It is believed that $P_N \cap T_N = \emptyset$.

Given a labelled net N and some transition $u \in T_N$, the *precondition* and *postcondition* u, written respectively ${}^\bullet u$ and $u^\bullet$, are the multisets defined in such a way: $({}^\bullet u)(p) = F_N(p, u)$ and $(u^\bullet)(p) = F_N(u, p)$. Analogous definitions are introduced for places: $({}^\bullet p)(u) = F_N(u, p)$ and $(p^\bullet)(u) = F_N(p, u)$. A transition u is *unstable* if ${}^\bullet u = \emptyset$. A labelled net is *stable* if it has no unstable transitions. Further we will deal only with stable labelled nets. A labelled net N is *ordinary* if $\forall p \in P_N \; {}^\bullet p$ and $p^\bullet$ are proper sets. A labelled net N is *finite* if $P_N \cup T_N$ is. Let ${}^\circ N = \{p \in P_N \mid {}^\bullet p = \emptyset\}$ is a set of *initial* places of N and $N^\circ = \{p \in P_N \mid p^\bullet = \emptyset\}$ is a set of *final* places of N.

Let N be a labelled net. A *marking* of N is a multiset $M \in \mathcal{M}(P_N)$. A *marked net* is a tuple $N = \langle P_N, T_N, F_N, l_N, M_N \rangle$ such that $\langle P_N, T_N, F_N, l_N \rangle$ is a labelled net and $M_N \in \mathcal{M}(P_N)$ is an *initial* marking. We write "net" instead of "marked net". Let $M \in \mathcal{M}(P_N)$ be a marking of a net N. A transition $u \in T_N$ is *firable* in M if ${}^\bullet u \subseteq M$. If u is firable in M, firing it yields a new marking $M' = M - {}^\bullet u + u^\bullet$, written $M \stackrel{u}{\to} M'$. We write $M \to M'$ if $M \stackrel{u}{\to} M'$ for some u. A marking M' of a net N is *reachable from marking* M of the net, if:

1) $M' = M$, or
2) there exists a reachable from M marking M'' of a net N, such that $M'' \to M'$.

A marking M of a net N is *reachable*, if it is reachable from M_N. $\mathsf{Mark}(N, M)$ denotes a *set of all reachable from* M markings of a net N, and $\mathsf{Mark}(N)$ denotes a *set of all reachable* markings of a net N.

An action $a \in \mathfrak{A}$ is *autoconcurrent* in N if $\exists M \in \mathsf{Mark}(N)\ \exists t, u \in T_N$ such that $l_N(u) = l_N(t) = a$ and ${}^\bullet t + {}^\bullet u \subseteq M$. A net N is *autoconcurrency free* if no action is autoconcurrent in N.

2.3. Processes

A *causal net* is a labelled net $C = \langle P_C, T_C, F_C, l_C \rangle$, where:

1) $\forall r \in P_C\ |{}^\bullet r| \leq 1$ and $|r^\bullet| \leq 1$, i.e., places are unbranched and C is an ordinary labelled net;
2) F_C is well-founded, i.e., there is no backward infinite chain $\cdots (r_n, v_n)(v_n, r_{n-1}) \cdots (r_1, v_1)(v_1, r_0)$ in F_C.

The fundamental property of causal nets is known: if C is a causal net, then there exists a transition sequence ${}^\circ C = L_0 \stackrel{v_1}{\rightarrow} \cdots \stackrel{v_n}{\rightarrow} L_n = C^\circ$ 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.

Given a net N and a causal net C. A mapping $f : P_C \cup T_C \to P_N \cup T_N$ is an *embedding* C into N, written $f : C \to N$, if:

1) $f(P_C) \in \mathcal{M}(P_N)$ and $f(T_C) \in \mathcal{M}(T_N)$;
2) $\forall v \in T_C\ l_C(v) = l_N(f(v))$;
3) $\forall v \in T_C\ {}^\bullet f(v) = f({}^\bullet v)$ and $f(v)^\bullet = f(v^\bullet)$.

Point 3 means that embeddings respect the flow relation. Consequently, if ${}^\circ C \stackrel{v_1}{\rightarrow} \cdots \stackrel{v_n}{\rightarrow} C^\circ$ is a full execution of C, then $M = f({}^\circ C) \stackrel{f(v_1)}{\longrightarrow} \cdots \stackrel{f(v_n)}{\longrightarrow} f(C^\circ) = M'$ is a transition sequence in N, *corresponding* to this full execution, written $M \stackrel{C,f}{\rightarrow} M'$. Conversely, for any transition sequence $M \stackrel{u_1}{\rightarrow} \cdots \stackrel{u_n}{\rightarrow} M'$ of a net N there exists a causal net C and an embedding $f : C \to N$ such that $M = f({}^\circ C)$, $M' = f(C^\circ)$, $u_i = f(v_i)$ $(0 \leq i \leq n)$ and ${}^\circ C \stackrel{v_1}{\rightarrow} \cdots \stackrel{v_n}{\rightarrow} C^\circ$ is a full execution of C.

A *firable in marking* M process of a net N is a pair $\pi = (C, f)$, where C is a causal net and $f : C \to N$ is an embedding such that $M = f({}^\circ C)$. A firable in M_N process is a *process* of N. We write $\Pi(N, M)$ for a *set of all firable* in M processes of N and $\Pi(N)$ for a *set of all processes* of N. Processes and reachable markings of a net N are connected in the following way: $\mathsf{Mark}(N, M) = \{f(C^\circ) \mid \pi = (C, f) \in \Pi(N, M)\}$. Further we will deal only with *finite* processes, i.e., with processes having finite causal nets.

If $\pi \in \Pi(N, M)$, then firing of this process transforms a marking M into $M' = M - f({}^\circ C) + f(C^\circ) = f(C^\circ)$, written $M \stackrel{\pi}{\rightarrow} M'$. A causal net sets an ordering on transitions (the *causal dependence* relation) $\prec_C$, defined in such a way: $\prec_C = F_C^+ \lceil_{T_C \times T_C}$, where F_C^+ is a transitive closure of F_C. The *initial* process of a net N is $\pi_N = (C_N, f_N) \in \Pi(N)$, where $T_{C_N} = \emptyset$. Let $\pi = (C, f)$, $\tilde{\pi} = (\tilde{C}, \tilde{f}) \in \Pi(N)$, $\hat{\pi} = (\hat{C}, \hat{f}) \in \Pi(N, f(C^\circ))$, $C = \langle P_C, T_C, F_C, l_C \rangle$, $\tilde{C} = \langle P_{\tilde{C}}, T_{\tilde{C}}, F_{\tilde{C}}, l_{\tilde{C}} \rangle$, $\hat{C} = \langle P_{\hat{C}}, T_{\hat{C}}, F_{\hat{C}}, l_{\hat{C}} \rangle$.

We write $\pi \stackrel{\hat{\pi}}{\rightarrow} \tilde{\pi}$, if:

1) $P_C \cup P_{\hat{C}} = P_{\tilde{C}}, T_C \cup T_{\hat{C}} = T_{\tilde{C}}, F_C \cup F_{\hat{C}} = F_{\tilde{C}}, F_C \cup l_{\hat{C}} = l_{\tilde{C}}$;
2) $f \cup \hat{f} = \tilde{f}$.

In such a case $\tilde{\pi}$ is an extension of π *by process* $\hat{\pi}$, and $\hat{\pi}$ is an *extending* process for π. We write $\pi \rightarrow \tilde{\pi}$, if $\pi \stackrel{\hat{\pi}}{\rightarrow} \tilde{\pi}$ for some extending process $\hat{\pi}$.

Let $\pi \stackrel{\hat{\pi}}{\rightarrow} \tilde{\pi}$. A process $\tilde{\pi}$ is an extension of π *by one action*, if $|T_{\hat{C}}| = 1$. In such a case we write $\pi \stackrel{v}{\rightarrow} \tilde{\pi}$ or $\pi \stackrel{a}{\rightarrow} \tilde{\pi}$, if $T_{\hat{C}} = \{v\}$ and $l_{\hat{C}}(v) = a$. A process $\tilde{\pi}$ is an extension of π *by multiset of actions*, or *step*, if $\prec_{\hat{C}} = \emptyset$. In such a case we write $\pi \stackrel{V}{\rightarrow} \tilde{\pi}$ or $\pi \stackrel{A}{\rightarrow} \tilde{\pi}$, if $T = V_{\hat{C}}$ and $l_{\hat{C}}(T_{\hat{C}}) = A$, $A \in \mathcal{M}(\mathfrak{A})$.

2.4. Mappings

Given nets $N = \langle P_N, T_N, F_N, l_N, M_N \rangle$ and $N' = \langle P_{N'}, T_{N'}, F_{N'}, l_{N'}, M_{N'} \rangle$. We call β a *mapping* of N *into* N', written $\beta : N \rightarrow N'$, if $\beta : P_N \cup T_N \rightarrow P_{N'} \cup T_{N'}$, $\beta(P_N) \subseteq P_{N'}$ and $\beta(T_N) \subseteq T_{N'}$. We write $\beta(N) = N'$, when $\beta(P_N) = P_{N'}$ and $\beta(T_N) = T_{N'}$.

A mapping $\beta : N \rightarrow N'$ is an *isomorphism* between N and N', written $\beta : N \simeq N'$, if:

1) β is a bijection and $\beta(N) = N'$;
2) $\forall u \in T_N \;\; l_N(u) = l_{N'}(\beta(u))$;
3) $\forall u \in T_N \;\; {}^\bullet\beta(u) = \beta({}^\bullet u)$ and $\beta(u)^\bullet = \beta(u^\bullet)$.

Nets N and N' are *isomorphic*, written $N \simeq N'$, if there exists an isomorphism $\beta : N \simeq N'$.

Given two labelled causal nets

$$C = \langle P_C, T_C, F_C, l_C \rangle \quad \text{and} \quad C' = \langle P_{C'}, T_{C'}, F_{C'}, l_{C'} \rangle.$$

A mapping $\beta : T_C \rightarrow T_{C'}$ is a *label preserving bijection* between T_C and $T_{C'}$, written $\beta : T_C \approx T_{C'}$, if:

1) β is a bijection and $\beta(T_C) = T_{C'}$;
2) $\forall v \in T_C \;\; l_C(v) = l_{C'}(\beta(v))$.

We write $T_C \approx T_{C'}$, if there exists a label-preserving bijection $\beta : T_C \approx T_{C'}$.

A mapping $\beta : T_C \rightarrow T_{C'}$ is a *homomorphism* between T_C and $T_{C'}$, written $\beta : T_C \sqsubseteq T_{C'}$, if:

1) $\beta : T_C \approx T_{C'}$;
2) $\forall v, w \in T_C \;\; v \prec_C w \Rightarrow \beta(v) \prec_{C'} \beta(w)$.

We write $T_C \sqsubseteq T_{C'}$, if there exists a homomorphism $\beta : T_C \sqsubseteq T_{C'}$.

A mapping $\beta : T_C \rightarrow T_{C'}$ is an *isomorphism* between T_C and $T_{C'}$, written $\beta : T_C \simeq T_{C'}$, if $\beta : T_C \sqsubseteq T_{C'}$ and $\beta^{-1} : T_{C'} \sqsubseteq T_C$. We write $T_C \simeq T_{C'}$, if there exists an isomorphism $\beta : T_C \simeq T_{C'}$.

3. Trace equivalences

A *sequential trace* of a net N is a sequence $a_1 \cdots a_n \in \mathfrak{A}^*$ such that $\pi_N \xrightarrow{a_1} \pi_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} \pi_n$, where $\pi_i \in \Pi(N)$ $(1 \leq i \leq n)$ and π_N is an initial process of N. $\mathsf{SeqTraces}(N)$ denotes a *set of all sequential traces* of N. Two nets N and N' are *interleaving trace equivalent*, written $N \equiv_i N'$, if $\mathsf{SeqTraces}(N) = \mathsf{SeqTraces}(N')$.

A *step trace* of a net N is a sequence $A_1 \cdots A_n \in (\mathcal{M}(\mathfrak{A}))^*$ such that $\pi_N \xrightarrow{A_1} \pi_1 \xrightarrow{A_2} \ldots \xrightarrow{A_n} \pi_n$, where $\pi_i \in \Pi(N)$ $(0 \leq i \leq n)$, and π_N is an initial process of N. $\mathsf{StepTraces}(N)$ denotes a *set of all step traces* of N. Two nets N and N' are *step trace equivalent*, written $N \equiv_s N'$, if $\mathsf{StepTraces}(N) = \mathsf{StepTraces}(N')$.

A *pomset trace* of a net N is a pomset ρ, an isomorphism class of T_C for $\pi = (C, f) \in \Pi(N)$, where $C = \langle P_C, T_C, F_C, l_C \rangle$. We write $\rho \sqsubseteq \rho'$, if $T_C \sqsubseteq T_{C'}$ for $T_C \in \rho$ and $T_{C'} \in \rho'$. In such a case we say that pomset ρ is *less sequential* or *more parallel* than ρ'. Let us denote a *set of all pomset traces* of N by $\mathsf{Pomsets}(N)$. Two nets N and N' are *partial word trace equivalent*, written $N \equiv_{pw} N'$, if $\mathsf{Pomsets}(N) \sqsubseteq \mathsf{Pomsets}(N')$ and $\mathsf{Pomsets}(N') \sqsubseteq \mathsf{Pomsets}(N)$, i.e., for any $\rho' \in \mathsf{Pomsets}(N')$ there exists $\rho \in \mathsf{Pomsets}(N)$ such that $\rho \sqsubseteq \rho'$ and vice versa. Two nets N and N' are *pomset trace equivalent*, written $N \equiv_{pom} N'$, if $\mathsf{Pomsets}(N) = \mathsf{Pomsets}(N')$.

A *process trace* of a net N is an isomorphism class of C for $\pi = (C, f) \in \Pi(N)$. $\mathsf{ProcessNets}(N)$ denotes a *set of all process traces* of N. Two nets N and N' are *process trace equivalent*, written $N \equiv_{pr} N'$, if $\mathsf{ProcessNets}(N) = \mathsf{ProcessNets}(N')$.

4. Bisimulation equivalences

In this section we consider the definitions of different bisimulations. A notation $\mathcal{R} : N \leftrightarrow_\alpha N'$ means that $\mathcal{R}$ is a bisimulation of α type between nets N and N'. Nets N and N' are called α-bisimulation equivalent, written $N \leftrightarrow_\alpha N'$, if $\mathcal{R} : N \leftrightarrow_\alpha N'$ for some α-bisimulation $\mathcal{R}$.

4.1. Simple bisimulations

Let $\mathcal{R} \subseteq \Pi(N) \times \Pi(N')$. In the following definition $\hat{\pi} = (\hat{C}, \hat{f})$, $\hat{\pi}' = (\hat{C}', \hat{f}')$.

$\mathcal{R}$ is a α-*bisimulation* between N and N', $\alpha \in \{$*interleaving, step, partial word, pomset, process*$\}$, written $\mathcal{R} : N \leftrightarrow_\alpha N'$, $\alpha \in \{i, s, pw, pom, pr\}$, 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 $\alpha = i$;
   (b) $\prec_{\hat{C}} = \emptyset$, if $\alpha = s$;

   then $\exists \tilde{\pi}' : \pi' \xrightarrow{\hat{\pi}'} \tilde{\pi}'$, $(\tilde{\pi}, \tilde{\pi}') \in \mathcal{R}$ and
   (a) $T_{\hat{C}'} \sqsubseteq T_{\hat{C}}$, if $\alpha = pw$;
   (b) $T_{\hat{C}} \simeq T_{\hat{C}'}$, if $\alpha \in \{i, s, pom\}$;
   (c) $\hat{C} \simeq \hat{C}'$, if $\alpha = pr$;
3. As previous item but N and N' are transposed.

4.2. ST-bisimulations

A *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 such a case π_E is a process which began to work, i.e., all actions of π_E began working. A process π_P corresponds to the terminated part of π_E, and π_W corresponds to the still working part. Clearly, $\prec_{C_W} = \emptyset$. $ST - \Pi(N)$ denotes a *set of all ST-processes* of N. (π_N, π_N) will be an *initial ST-process* of N. Let (π_E, π_P), $(\tilde{\pi}_E, \tilde{\pi}_P) \in ST - \Pi(N)$. We write $(\pi_E, \pi_P) \rightarrow (\tilde{\pi}_E, \tilde{\pi}_P)$, if $\pi_E \rightarrow \tilde{\pi}_E$ and $\pi_P \rightarrow \tilde{\pi}_P$.

Let $\mathcal{R} \subseteq ST - \Pi(N) \times ST - \Pi(N') \times \mathcal{B}$, where $\mathcal{B} = \{\beta \mid \beta : T_C \rightarrow T_{C'},\ \pi = (C, f) \in \Pi(N),\ \pi' = (C', f') \in \Pi(N')\}$. In the following definitions $\pi_E = (C_E, f_E)$, $\pi_P = (C_P, f_P)$, $\pi'_E = (C'_E, f'_E)$, $\pi'_P = (C'_P, f'_P)$, $\pi = (C, f)$, $\pi' = (C', f')$.

$\mathcal{R}$ is a *α-ST-bisimulation* between N and N', $\alpha \in \{$*interleaving, partial word, pomset, process*$\}$, written $\mathcal{R} : N \leftrightarrow_{\alpha ST} N'$, $\alpha \in \{i, pw, pom, pr\}$, if:

1. $((\pi_N, \pi_N), (\pi_{N'}, \pi_{N'}), \emptyset) \in \mathcal{R}$;
2. $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R} \Rightarrow \beta : T_{C_E} \approx T_{C'_E}$ and $\beta(T_{C_P}) = T_{C'_P}$;
3. $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R}$, $(\pi_E, \pi_P) \rightarrow (\tilde{\pi}_E, \tilde{\pi}_P) \Rightarrow \exists \tilde{\beta}, (\tilde{\pi}'_E, \tilde{\pi}'_P)$: $(\pi'_E, \pi'_P) \rightarrow (\tilde{\pi}'_E, \tilde{\pi}'_P)$, $\tilde{\beta}\lceil_{T_{C_E}} = \beta$, $((\tilde{\pi}_E, \tilde{\pi}_P), (\tilde{\pi}'_E, \tilde{\pi}'_P), \tilde{\beta}) \in \mathcal{R}$, and if $\pi_P \xrightarrow{\pi} \tilde{\pi}_E$, $\pi'_P \xrightarrow{\pi'} \tilde{\pi}'_E$ then:
   (a) $(\tilde{\beta}\lceil_{T_C})^{-1} : T_{C'} \sqsubseteq T_C$, if $\alpha = pw$;
   (b) $\tilde{\beta}\lceil_{T_C} : T_C \simeq T_{C'}$, if $\alpha \in \{pom, pr\}$;
   (c) $C \simeq C'$, if $\alpha = pr$;
4. As previous item but N and N' are transposed.

4.3. History preserving bisimulations

Let $\mathcal{R} \subseteq \Pi(N) \times \Pi(N') \times \mathcal{B}$, where $\mathcal{B} = \{\beta \mid \beta : T_C \rightarrow T_{C'},\ \pi = (C, f) \in \Pi(N),\ \pi' = (C', f') \in \Pi(N')\}$. In the following definitions $\pi = (C, f)$, $\tilde{\pi} = (\tilde{C}, \tilde{f})$, $\pi' = (C', f')$, $\tilde{\pi}' = (\tilde{C}', \tilde{f}')$.

$\mathcal{R}$ is a *α-history preserving bisimulation* between N and N', $\alpha \in \{$*partial word, pomset, process*$\}$, written $N \leftrightarrow_{\alpha h} N'$, $\alpha \in \{pw, pom, pr\}$, if:

1. $(\pi_N, \pi_{N'}, \emptyset) \in \mathcal{R}$;
2. $(\pi, \pi', \beta) \in \mathcal{R} \Rightarrow \beta : T_C \approx T_{C'}$;
3. $(\pi, \pi', \beta) \in \mathcal{R}$, $\pi \rightarrow \tilde{\pi} \Rightarrow \exists \tilde{\beta}, \tilde{\pi}' : \pi' \rightarrow \tilde{\pi}'$, $\tilde{\beta}\lceil_{T_C} = \beta$, $(\tilde{\pi}, \tilde{\pi}', \tilde{\beta}) \in \mathcal{R}$ and
   (a) $\tilde{\beta}^{-1} : T_{\tilde{C}'} \sqsubseteq T_{\tilde{C}}$, if $\alpha = pw$;
   (b) $\tilde{\beta} : T_{\tilde{C}} \simeq T_{\tilde{C}'}$, if $\alpha \in \{pom, pr\}$;
   (c) $\tilde{C} \simeq \tilde{C}'$, if $\alpha = pr$;
4. As previous item but N and N' are transposed.

5. A comparison of the equivalences

In this section a theorem establishing a correlation of all introduced equivalences is proved.

Theorem 1. *Let* $\sim \in \{\equiv, \underline{\leftrightarrow}\}$ *and* $\alpha, \beta \in \{i, s, pw, pom, pr, iST, pwST, pomST, prST, pwh, pomh, prh\}$. *For nets* N *and* N', $N \sim_\alpha N' \Rightarrow N \sim_\beta N'$ *iff there exists a directed path* $\sim_\alpha \rightarrow \cdots \rightarrow \sim_\beta$ *in a graph in Figure* 1.

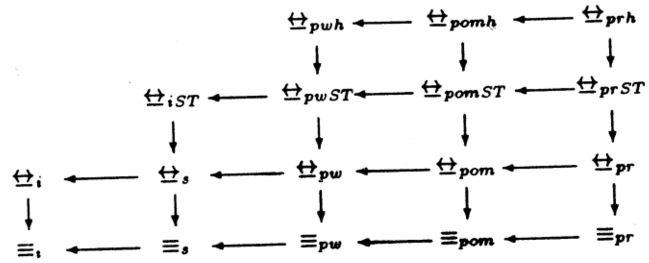

Figure 1. Correlation of the equivalences

Proof.

$\Leftarrow$ By definitions of the equivalences.

$\Rightarrow$ It is sufficient to consider the following examples.

- In Figure 2.1: $N \underline{\leftrightarrow}_i N'$ but $N \not\equiv_s N'$ since there exists a step trace $\{a, b\}$ in N which is not in N'.
- In Figure 2.2: $N \equiv_{pr} N'$ but $N \not\underline{\leftrightarrow}_i N'$ since only in N an action a can happen such that it is impossible to run b after it.
- In Figure 2.3: $N \underline{\leftrightarrow}_{pwh} N'$ but $N \not\equiv_{pom} N'$ since b can depend on a in N.
- In Figure 2.4: $N \underline{\leftrightarrow}_{pomh} N'$ but $N \not\equiv_{pr} N'$ since N is a causal net which is not isomorphic to causal net N'.
- In Figure 2.5: $N \underline{\leftrightarrow}_{iST} N'$ but $N \not\equiv_{pw} N'$ since a net N is corresponded by a pomset such that there is not even less sequential pomset in N'.
- In Figure 3.1: $N \underline{\leftrightarrow}_{pr} N'$ but $N \not\underline{\leftrightarrow}_{iST} N'$ since an action a is able to begin working in N' so that no b can start later.
- In Figure 3.2: $N \underline{\leftrightarrow}_{prST} N'$ but $N \not\underline{\leftrightarrow}_{pwh} N'$ since only in N' actions a and b can happen so that the next action, c, must depend on a. □

6. Equivalences on different net subclasses

In the literature a several subclasses of nets were proposed by introduction some restrictions on the initial definition of nets, and merging of equivalences was obtained on these types of nets. See for example [2, 7]. We will consider the introduced equivalences on sequential nets, on T-nets and on nets with strict labelling.

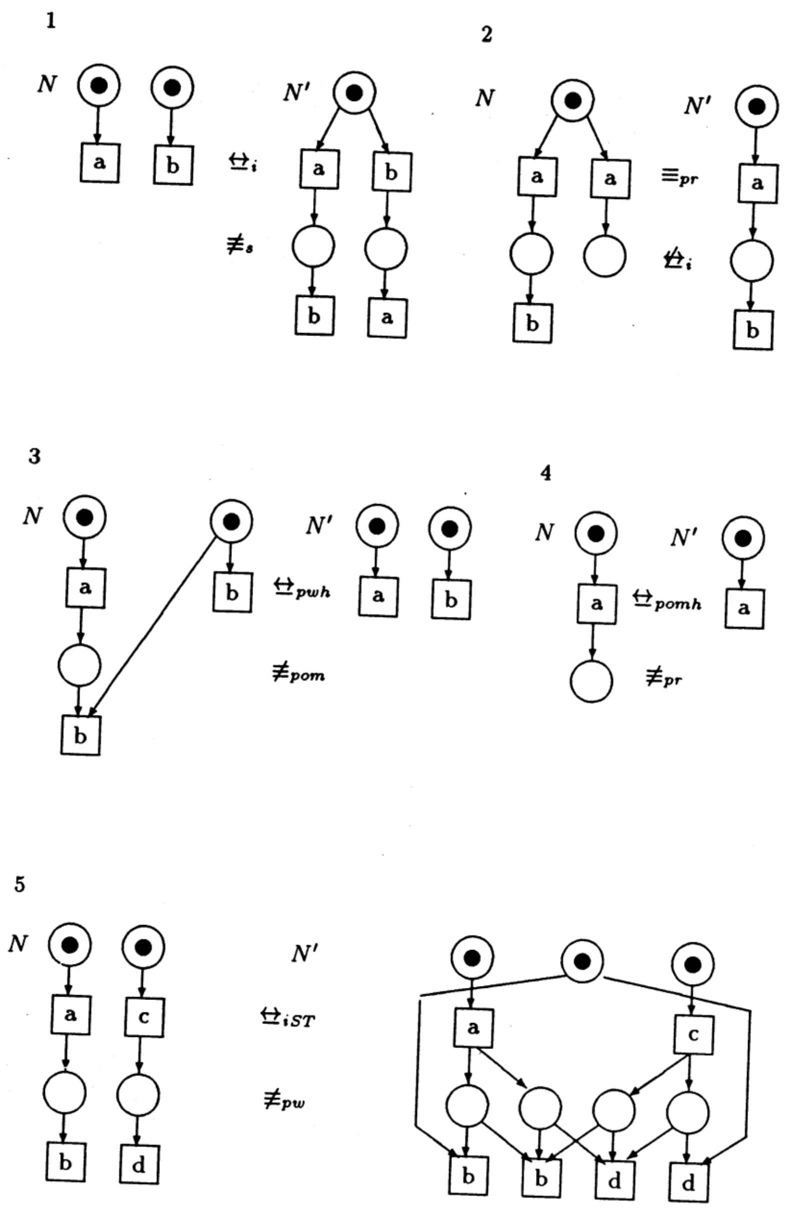

Figure 2. Examples of nets

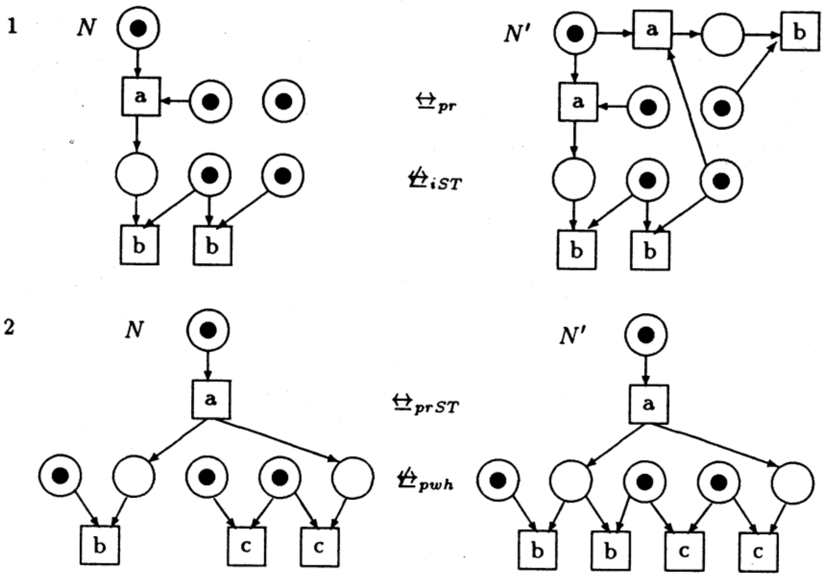

Figure 3. Examples of nets (continued)

A net $N = \langle P_N, T_N, F_N, l_N, M_N \rangle$ is *sequential* if $\forall \pi = (C, f) \in \Pi(N)$, $\forall v, w \in T_C$ $(v \prec_C w) \vee (w \prec_C v)$, i.e., $\prec_C$ is a strict (total) ordering on causal net transitions of any process $\pi = (C, f)$ of the net N.

Proposition 1. *For sequential nets N and N',*

1. [2] $N \underline{\leftrightarrow}_i N' \Leftrightarrow N \underline{\leftrightarrow}_{pomh} N'$;
2. $N \equiv_i N' \Leftrightarrow N \equiv_{pom} N'$.

Theorem 2. *Let $\sim \in \{\equiv, \underline{\leftrightarrow}\}$, $\alpha, \beta \in \{i, pr, prST, prh\}$. For sequential nets N and N' $N \sim_\alpha N' \Rightarrow N \sim_\beta N'$ iff there exists a directed path $\sim_\alpha \rightarrow \cdots \rightarrow \sim_\beta$ in graph in Figure* 4.

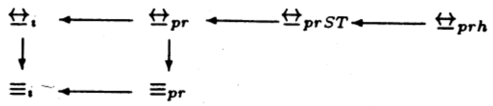

Figure 4. Equivalences on sequential nets

Proof. $\Leftarrow$ By Theorem 1.

$\Rightarrow$ It is sufficient to consider the following examples on sequential nets.

- In Figure 2.4: $N \underline{\leftrightarrow}_i N'$ but $N \not\equiv_{pr} N'$.
- In Figure 2.2: $N \equiv_{pr} N'$ but $N \not\underline{\leftrightarrow}_i N'$.
- In Figure 5.1: $N \underline{\leftrightarrow}_{pr} N'$ but $N \not\underline{\leftrightarrow}_{prST} N'$ since only in N' we can begin running a process with action a so that it may be extended by action b in the only way (i.e., so that extended process be only one).

- In Figure 5.2: $N \underline{\leftrightarrow}_{prST} N'$ but $N \not\underline{\leftrightarrow}_{prh} N'$ since only in N' it is possible to run a process with sequential occuring actions a and b so that the next action, c, may extend this process only in one way (i.e., causal net with action c, extending a causal net corresponding to sequence ab, connects with its subnet containing a, in the only way). □

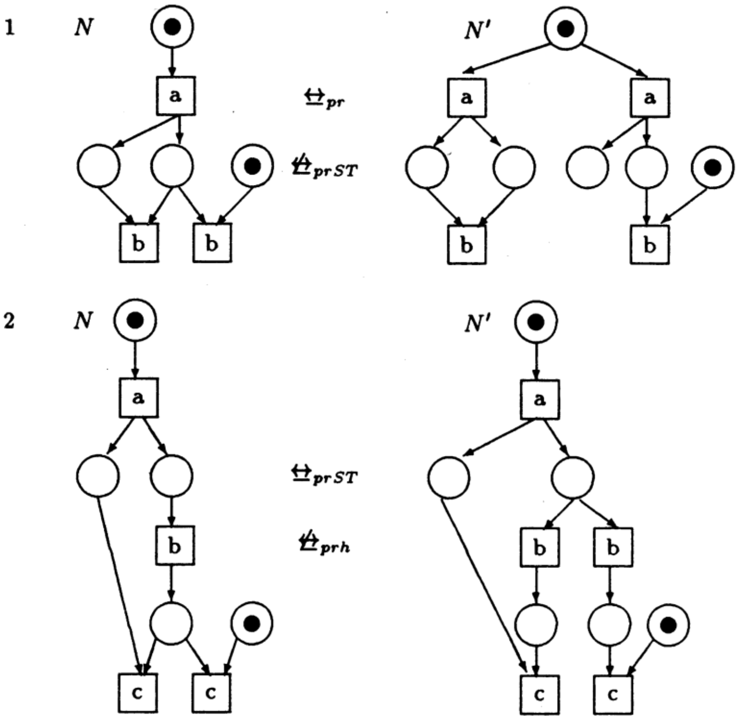

Figure 5. Examples of sequential nets

A *T-net* is a net $N = \langle P_N, T_N, F_N, l_N, M_N \rangle$ such that $\forall p \in P_N \ |{}^\bullet p| \leq 1$ and $|p^\bullet| \leq 1$.

Proposition 2. *For autoconcurrency free T-nets N and N', $N \equiv_i N' \Leftrightarrow N \underline{\leftrightarrow}_{iST} N'$.*

No pomset equivalence is a consequence of partial word one, and no process equivalence is a consequence of pomset one on T-nets without autoconcurrency. It is demonstrated correspondently by Figure 6.2 where $N \underline{\leftrightarrow}_{pwh} N'$ but $N \not\equiv_{pom} N'$ since only in N' an action b can depend on a and by Figure 2.4 where $N \underline{\leftrightarrow}_{pomh} N'$ and $N \not\equiv_{pr} N'$. Let us note that for *safe* autoconcurrency free T-nets we can use the results of [10] and establish the coincidence of interleaving and pomset trace equivalences.

A net $N = \langle P_N, T_N, F_N, l_N, M_N \rangle$ is a *strictly labelled*, if its labelling function is l_N bijective, i.e., $\forall t, u \in T_N \ t \neq u \Rightarrow l_N(t) \neq l_N(u)$.

Proposition 3. *For strictly labelled nets N and N', $N \equiv_\alpha N' \Leftrightarrow N \underline{\leftrightarrow}_\alpha N'$, $\alpha \in \{i, s, pw, pom, pr\}$.*

For strictly labelled nets we can not draw any arrow in a graph in Figure 1 from interleaving to step, from partial word to pomset and from pomset to process equivalences. In addition, in all semantics from interleaving to pomset the history preserving bisimulation equivalences are strictly stronger than ST-bisimulation ones. It is proved by the following examples.

- In Figure 6.1: $N \underline{\leftrightarrow}_i N'$ but $N \not\equiv_s N'$, since only in N actions a and b can work concurrently.
- In Figure 6.2: $N \underline{\leftrightarrow}_{pwh} N'$ but $N \not\equiv_{pom} N'$.
- In Figure 2.4: $N \underline{\leftrightarrow}_{pomh} N'$ but $N \not\equiv_{pr} N'$.
- In Figure 6.3: $N \underline{\leftrightarrow}_{pomST} N'$ but $N \not\underline{\leftrightarrow}_{pwh} N'$, since in N' the sequence ab can happen so that the next action, c, must depend on a.

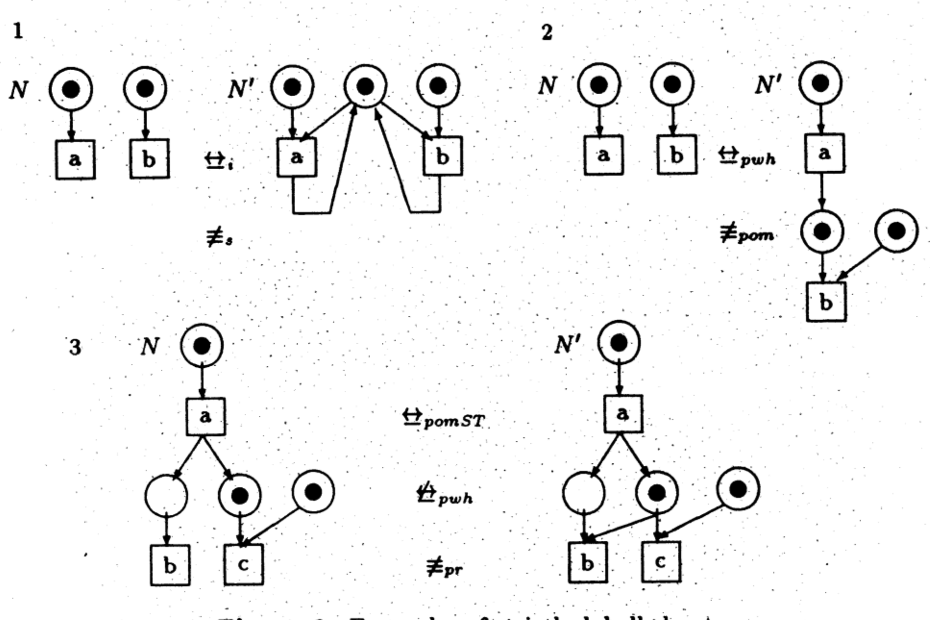

Figure 6. Examples of strictly labelled nets

7. Conclusion

A group of the Petri net equivalences is introduced in the paper. A correlation of these equivalences on nets with finite processes without λ-actions is found. In addition, it is considered which equivalences coincide on different subclasses of nets.

The development of the subject consists in further exploration of the introduced equivalences on T-nets and strictly labelled nets.

The next direction of the development of this theme may be an examination of the proposed equivalences on the wider net class, exactly, on nets with λ-actions. Probably some equivalences will not be connected on such nets. In [11] the example of event structures with λ-actions was considered. It is demonstrated the independence of ST-bisimulation equivalences and h-bisimulation equivalence on such event structures.

Finally, it would be interesting to find out how ST- and history preserving equivalences are connected with place bisimulation equivalences introduced in [1].

Acknowledgements. I would like to thank I.B. Virbitskaite for setting the theme, supervision in research work and for many helpful discussions which improved the style and contents of the paper.

References

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

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

[3] R.J. van Glabbeek, *The refinement theorem for ST-bisimulation semantics*, In: M.Broy and C.B.Jones, editors. Proc. IFIP Working Conference of Programming Concepts and Methods, Sea of Galilee, Israel, 1990.

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

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

[6] D. Park, *Concurrency and automata on infinite sequences*, LNCS 104, March 1981, 167–183.

[7] L. Pomello, G. Rozenberg, C. Simone, *A survey of equivalence notions for net based systems*, LNCS 609, 1992, 410–472.

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

[9] I.V. Tarasyuk, *Equivalences on Petri nets*, In: Specification, Verification and Net Models of Concurrent Systems. Institut of Informatics Systems, Novosibirsk, 1994, to appear.

[10] F.W. Vaandrager, *Determinism $\rightarrow$ (event structure isomorphism = step sequence equivalence)*, TCS 79, February 1991, 275–294.

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