Joint NCC & IIS Bull., Comp.Science, 5 (1996), 83–100

An algebra of labelled nondeterministic processes*

Igor V. Tarasyuk

A new calculus of labelled nondeterministic processes $AFLP_2$ is proposed which is an extension of the known calculus AFP_2 [3] by labelling function. The denotational and operational semantics and complete axiomatization of the semantic equivalence are presented. The interrelation of net equivalences from [11, 12] with equivalences of the algebra (semantic and observational) is considered. Analogs of the net equivalences are defined in $AFLP_2$, allowing one to consider the processes specified by formulas of the algebra at different levels of abstraction.

1. Introduction

Algebraic calculi are one of the popular formal models for specification of concurrent systems and processes and investigation of their behavioural properties.

The calculus AFP_2 (Algebra of Finite Processes) proposed by V.E. Kotov and L.A. Cherkasova [3] combines mechanisms both for specification and analysis of nondeterministic concurrent processes. An advantage of AFP_2 is a mechanism of action synchronization by names which means that all actions with the same name are synchronized. But it is impossible to specify the process with several actions which have the same name within AFP_2.

We introduce the new calculus $AFLP_2$ (Algebra of Finite Labelled Processes) based on AFP_2 by imposing the global labelling on its formulas. Hence, the formulas of $AFLP_2$ specify much wider class of *labelled* nondeterministic processes where some different events may be equally labelled. The denotational and operational semantics of $AFLP_2$ are introduced. A sound and complete set of axioms corresponding to the semantic equivalence of $AFLP_2$ is presented. By means of $AFLP_2$ one can analyse a behaviour of weakly labelled A-nets (i.e., A-nets [6] with possibly noninjective labelling function). The semantic and observational equivalences of $AFLP_2$ are transferred to weakly labelled A-nets, and their interrelation with the net equivalences from [11] is examined. Analogs of the net equivalences are introduced on formulas of $AFLP_2$. We prove that the semantic equivalence of $AFLP_2$ is the only congruence w.r.t. operations of the algebra.

*The work is supported by Volkswagen Fund, grant I/70 564 and Russian Foundation for Basic Research, grant 96-01-01655

Notice that formal definitions of multisets, nets, lposets, pomsets, causal nets, processes, ST-processes and mappings (label-preserving bijection $\approx$, homomorphism $\sqsubseteq$, isomorphism $\simeq$) and other concepts which are used in the paper can be found in [11, 12]. Complete proofs are given in [13].

2. Algebra $AFLP_2$

2.1. Syntax

Let $Ev = \{e, f, \ldots\}$ be symbols of *events*, $\overline{Ev} = \{\bar{e}, \bar{f}, \ldots\}$ be symbols of *non-events* and $\Delta_{Ev} = \{\delta_e, \delta_f, \ldots\}$ be symbols of *deadlocked events*. Let us denote $\widehat{Ev} = Ev \cup \overline{Ev} \cup \Delta_{Ev}$. The symbols of $\widehat{Ev}$ are combined into formulas by operations ; (*precedence*), $\bigtriangledown$ (*exclusive or, alternative*), $\|$ (*concurrency*), $\vee$ (*disjunction, union*), $\rceil\!\rceil$ (*"not occur"*), $\tilde{\rceil\!\rceil}$ (*"not occur by mistake"*). Let $Act = \{a, b, \ldots\}$ be *action* symbols *(labels)*. The *global labelling function* $lab : Ev \to Act$ binds an action with each event. The function is extended to $\overline{Ev} \cup \Delta_{Ev}$ as follows: $lab(\bar{e}) = \overline{lab(e)}$ and $lab(\delta_e) = \delta_{lab(e)}$.

A *formula* of $AFLP_2$ in the basis $\widehat{Ev}$ is defined as follows.

$$E ::= \ e \mid \bar{e} \mid \delta_e \mid \rceil\!\rceil E \mid \tilde{\rceil\!\rceil} E \mid E;F \mid E\|F \mid E \bigtriangledown F \mid E \vee F$$

Here $e \in Ev$, $\bar{e} \in \overline{Ev}$, $\delta_e \in \Delta_{Ev}$ are *elementary formulas.* $\mathbf{AFLP_2}$ denotes the *set of all formulas* of $AFLP_2$.

For a formula E the set $Ev(E)$ is defined as follows: $Ev(e) = Ev(\bar{e}) = Ev(\delta_e) = e$; $Ev(\neg E) = Ev(E)$, $\neg \in \{\rceil\!\rceil, \tilde{\rceil\!\rceil}\}$; $Ev(E \circ F) = Ev(E) \cup Ev(F)$, $\circ \in \{;, \|, \bigtriangledown, \vee\}$. Let $\overline{Ev}(E) = \{\bar{e} \mid e \in Ev(E)\}$, $\Delta_{Ev}(E) = \{\delta_e \mid e \in Ev(E)\}$ and $\widehat{Ev}(E) = Ev(E) \cup \overline{Ev}(E) \cup \Delta_{Ev}(E)$. One can associate with every formula E a *local labelling function* $l_E = lab|_{Ev(E)}$.

The *contents* of E, denoted by $cont(E)$, is defined as follows: $cont(e) = e$, $cont(\bar{e}) = \bar{e}$, $cont(\delta_e) = \delta_e$; $cont(\neg E) = cont(E)$, $\neg \in \{\rceil\!\rceil, \tilde{\rceil\!\rceil}\}$; $cont(E \circ F) = cont(E) \cup cont(F)$, $\circ \in \{;, \|, \bigtriangledown, \vee\}$. Let $cont^+(E) = cont(E) \cap Ev$ be the *set of events* of E, $cont^-(E) = cont(E) \cap \overline{Ev}$ be the *set of non-events* of E, and $\Delta_{cont}(E) = cont(E) \cap \Delta_{Ev}$ be the *set of deadlocked events* of E.

Two formulas E and E' are *isomorphic*, denoted by $E \simeq E'$, if they coincide up to associativity rules w.r.t. $;, \|, \vee, \bigtriangledown$ and up to commutativity rules w.r.t. $\|, \vee, \bigtriangledown$.

2.2. Denotational semantics

An *lposet* is a triple $\rho = \langle X, \prec, l \rangle$ where:

- $X \subseteq \widehat{Ev}$;
- $\prec \subseteq X \times X$ is a strict partial order over X, a *precedence relation*;

- $l : Ev(X) \to Act$ is a *labelling function.*

Here $Ev(X) = \{e \mid (e \in X) \vee (\bar{e} \in X) \vee (\delta_e \in X)\}$. Let $\overline{Ev}(X) = \{\bar{e} \mid e \in Ev(X)\}$, $\Delta_{Ev}(X) = \{\delta_e \mid e \in Ev(X)\}$ and $\widehat{Ev}(X) = Ev(X) \cup \overline{Ev}(X) \cup \Delta_{Ev}(X)$. $X^+ = X \cap Ev$ denotes the *subset of events* of X, $X^- = X \cap \overline{Ev}$ denotes the *subset of non-events* of X and $\Delta_X = X \cap \Delta_{Ev}$ denotes the *subset of deadlocked events* of X.

Since now on we consider only lposets with the following properties.

1. e, $\bar{e}$ and δ_e do not occur in X together, i.e. exactly one of the following three cases is possible: either e occurs in X, or $\bar{e}$, or δ_e;
2. partial order $\prec$ is irreflexive;
3. $\forall x, y \in X^- \cup \Delta_X \ (x \not\prec y) \wedge (y \not\prec x)$;
4. $\forall x \in X^+ \ \forall y \in X^- \cup \Delta_X \ (x \not\prec y) \wedge (y \not\prec x)$.

The *modified union* of lposets absorbs equal and incomplete computations:

$$\rho \,\tilde{\cup}\, \rho' = \begin{cases} \rho, & \rho' \text{ is a prefix of } \rho; \\ \rho', & \rho \text{ is a prefix of } \rho'; \\ \{\rho, \rho'\}, & \text{otherwise.} \end{cases}$$

To define the denotational semantics of $AFLP_2$, analogs of algebraic operations over lposets are introduced. If lposet ρ constructed by means of these operations, does not satisfy the conditions 1-4 mentioned above, we "correct" it using a new auxilary *regularization* operation $[\rho]$.

Let $D_1 = \{\delta_e \mid (e \in X) \wedge (e \prec e)\} \cup \{\delta_e \mid (e \in X) \wedge (\bar{e} \in X)\} \cup \{\delta_e \mid (e \in X) \wedge (\delta_e \in X)\} \cup \{\delta_e \mid (\bar{e} \in X) \wedge (\delta_e \in X)\} \cup \Delta_X$, $D_2 = \{\delta_e \mid (e \in X) \wedge (\delta_f \in D_1) \wedge (\delta_f \prec e)\}$ and $D_3 = \{\delta_e \mid \bar{e} \in X\}$. We define the set D as follows.

$$D = \begin{cases} \emptyset, & D_1 = \emptyset; \\ D_1 \cup D_2 \cup D_3, & \text{otherwise.} \end{cases}$$

Then $[\rho] = \langle D, \emptyset, l|_{Ev(D)} \rangle \cup \langle Y, \prec \cap (Y \times Y), l|_{Ev(Y)} \rangle$, where $Y = X \setminus \widehat{Ev}(D)$.

Let $\rho = \langle X, \prec, l \rangle$, $\rho' = \langle X, \prec', l' \rangle$. We introduce the lposet operations in the following way.

Not occur $\urcorner\!\!\urcorner \rho = \langle \overline{Ev}(X), \emptyset, l \rangle$.

Not occur by mistake $\tilde{\urcorner\!\!\urcorner} \rho = \langle \Delta_{Ev}(X), \emptyset, l \rangle$.

Precedence $\rho ; \rho' = [\langle X \cup X', \prec \cup \prec' \cup (X^+ \times (X')^+) \cup (\Delta_X \times (X')^+), l \cup l' \rangle]$.

Concurrency $\rho \| \rho' = [\langle X \cup X', (\prec \cup \prec')^+, l \cup l' \rangle]$, where $(\prec \cup \prec')^+$ is a transitive closure of relation $\prec \cup \prec'$.

Alternative $\rho \bigtriangledown \rho' = [\langle X \cup \overline{Ev}(X'), \prec, l \cup l' \rangle] \tilde{\cup} [\langle \overline{Ev}(X) \cup X', \prec', l \cup l' \rangle]$.

Let $\mathcal{P} = \cup_{i=1}^{n} \rho_i$ and $\mathcal{P}' = \cup_{j=1}^{m} \rho'_j$ be sets of lposets. Then $\neg\mathcal{P} = \tilde{\cup}_{i=1}^{n} \neg\rho_i$, where $\neg \in \{\rceil\!\rceil, \tilde{\rceil\!\rceil}\}$ and $\mathcal{P} \circ \mathcal{P}' = \tilde{\cup}_{i=1}^{n} (\tilde{\cup}_{j=1}^{m} \rho_i \circ \rho'_j)$, where $\circ \in \{;, \|, \bigtriangledown\}$.

The *denotational semantics* of $AFLP_2$ is a mapping $\mathcal{D}_{FL2}$ from **AFLP$_2$** into the set of lposets defined as follows.

1. $\mathcal{D}_{FL2}[e] = \langle \{e\}, \emptyset, l_e \rangle$, $\mathcal{D}_{FL2}[\bar{e}] = \langle \{\bar{e}\}, \emptyset, l_e \rangle$, $\mathcal{D}_{FL2}[\delta_e] = \langle \{\delta_e\}, \emptyset, l_e \rangle$;
2. $\mathcal{D}_{FL2}[\neg E] = \neg\mathcal{D}_{FL2}[E]$, $\neg \in \{\rceil\!\rceil, \tilde{\rceil\!\rceil}\}$;
3. $\mathcal{D}_{FL2}[E \circ F] = \mathcal{D}_{FL2}[E] \circ \mathcal{D}_{FL2}[F]$, $\circ \in \{;, \|, \bigtriangledown\}$;
4. $\mathcal{D}_{FL2}[E \vee F] = \mathcal{D}_{FL2}[E] \,\tilde{\cup}\, \mathcal{D}_{FL2}[F]$.

Two formulas E and E' are *semantically equivalent*, denoted by $E \approx_{FL2} E'$ iff $\mathcal{D}_{FL2}[E] = \mathcal{D}_{FL2}[E']$.

If $\rho = \langle X, \prec, l \rangle$ is an lposet, then $\rho^+ = \langle X^+, \prec, l|_{X^+} \rangle$ is the lposet corresponding to the "observable" part of ρ over Ev. If some formula E of $AFLP_2$ $\mathcal{D}_{FL2}[E] = \cup_{i=1}^{n} \rho_i$, then the *"observable"* part of this set is defined as follows: $\mathcal{D}_{FL2}^{+}[E] = \cup_{i=1}^{n} \rho_i^+$. Two formulas E and E' are *observationally equivalent*, denoted by $E \approx_{FL2+} E'$ iff $\mathcal{D}_{FL2}^{+}[E] = \mathcal{D}_{FL2}^{+}[E']$.

Proposition 1. *For any two formulas of $AFLP_2$ E and E' and context C the following holds:* $E \approx_{FL2} E' \Leftrightarrow \forall C\ C[E] \approx_{FL2} C[E']$.

Thus, $\approx_{FL2}$ is a congruence w.r.t. operations of $AFLP_2$.

2.3. Axiomatization

In accordance with the equivalence $\approx_{FL2}$ the axiom system Θ_{FL2} is introduced. It is represented in Table 1, where E, F, G are formulas of $AFLP_2$, and $e \in Ev$, $\bar{e} \in \overline{Ev}$ and $\delta_e \in \Delta_{Ev}$.

The definition of a *canonical form* of $AFLP_2$-formula coincides with that of AFP_2-formula [3]. Informally, a formula E is in canonical form, if it has the form $E = \vee_{i=1}^{n} \|_{j=1}^{m_i} E_{ij}$, s.t. E_{ij} $(1 \le i \le n,\ 1 \le j \le m_i)$ are elementary formulas or *elementary precedences* of the form $(e; f)$ where $e, f \in Ev$, and some additional conditions are satisfied which guarantee that each disjunctive member of a canonical form has a special form practically coinciding with lposet corresponding to one of possible alternative behaviours of the nondeterministic process specified by the formula.

Table 1. Axiom system Θ_{FL2}

1. Associativity	**5. Structural properties**
1.1 $E\parallel(F\parallel G) = (E\parallel F)\parallel G$	5.1 $\bar{e}; E = \bar{e}\parallel E$
1.2 $E \bigtriangledown (F \bigtriangledown G) = (E \bigtriangledown F) \bigtriangledown G$	5.2 $E; \bar{e} = E\parallel \bar{e}$
1.3 $E \vee (F \vee G) = (E \vee F) \vee G$	5.3 $E\parallel(E; F) = (E; F)$
1.4 $E; (F; G) = (E; F); G$	5.4 $F\parallel(E; F) = (E; F)$
2. Commutativity	5.5 $E; F; G = (E; F)\parallel(F; G)$
2.1 $E\parallel F = F\parallel E$	5.6 $(E; F)\parallel(F; G) = (E; F)\parallel (F; G)\parallel(E; G)$
2.2 $E \bigtriangledown F = F \bigtriangledown E$	5.7 $E\parallel E = E$
2.3 $E \vee F = F \vee E$	5.8 $E \vee E = E$
3. Distributivity	5.9 $E \vee F = E$, if $F \lhd E$
3.1 $(E\parallel F); G = (E; G)\parallel(F; G)$	(a concept of strict prefix $\lhd$ for formulas is defined in [13])
3.2 $E; (F\parallel G) = (E; F)\parallel(E; G)$	**6. Axioms for Δ_{Ev} and $\tilde{\prod}$**
3.3 $(E \vee F); G = (E; G) \vee (F; G)$	6.1 $e\parallel\bar{e} = \delta_e$
3.4 $E; (F \vee G) = (E; F) \vee (E; G)$	6.2 $e; e = \delta_e$
3.5 $(E \vee F)\parallel G = (E\parallel G) \vee (F\parallel G)$	6.3 $e\parallel\delta_e = \delta_e$
3.6 $E \bigtriangledown (F\parallel G) = (E \bigtriangledown F)\parallel(E \bigtriangledown G)$	6.4 $\delta_e; E = \delta_e\parallel(\tilde{\prod} E)$
4. Axioms for $\bigtriangledown$ and $\prod$	6.5 $E; \delta_e = E\parallel\delta_e$
4.1 $E \bigtriangledown F = (E\parallel(\prod F)) \vee ((\prod E)\parallel F)$	6.6 $\delta_e\parallel(\prod E) = \delta_e\parallel(\tilde{\prod} E)$
4.2 $\prod(E\parallel F) = (\prod E)\parallel(\prod F)$	6.7 $\tilde{\prod} e = \delta_e$
4.3 $\prod(E \vee F) = (\prod E) \vee (\prod F)$	6.8 $\tilde{\prod} \bar{e} = \delta_e$
4.4 $\prod(E; F) = (\prod E)\parallel(\prod F)$	6.9 $\tilde{\prod} \delta_e = \delta_e$
4.5 $\prod e = \bar{e}$	6.10 $\tilde{\prod}(E\parallel F) = (\tilde{\prod} E)\parallel(\tilde{\prod} F)$
4.6 $\prod \bar{e} = \bar{e}$	6.11 $\tilde{\prod}(E; F) = (\tilde{\prod} E)\parallel(\tilde{\prod} F)$
4.7 $\prod \delta_e = \bar{e}$	6.12 $\tilde{\prod}(E \vee F) = (\tilde{\prod} E) \vee (\tilde{\prod} F)$

Theorem 1. *Using Θ_{FL2}, it is posiible to prove that any formula of $AFLP_2$ may is equal to a unique (up to isomorphism) canonical form.*

Let *canon*(E) denotes the *set of all canonical forms* of a formula E. The notation $E =_{\Theta_{FL2}} E'$ means that the equation may be proved using the axiom system Θ_{FL2}.

Theorem 2. *For any two formulas of $AFLP_2$ E and E' the following holds:* $E \approx_{FL2} E' \Leftrightarrow E =_{\Theta_{FL2}} E'$.

Hence, to find whether any two formulas E and E' are semantically equivalent, it is sufficient to reduce them to their canonical forms F and F' and check these forms by isomorphism.

2.4. Operational semantics

A *transition system* is a quadruple $TS = \langle S, L, \rightarrow, s_{TS}\rangle$, where:

- S is a set of *states*;
- L is a set of *labels*;
- $\rightarrow \subseteq S \times L \times S$ is a set of *transitions*;
- $s_{TS} \in S$ is an *initial state*.

A transition $(s, a, \tilde{s})$ is denoted by $s \xrightarrow{a} \tilde{s}$.

Let us consider the following transition system over formulas of $AFLP_2$. If F is a formula of $AFLP_2$ in the canonical form, then $TS(F) = \langle \mathbf{AFLP_2} \cup \{\nu\}, \mathbf{AFLP_2}, \rightarrow_{TS}, F\rangle$, where:

- The set of states, $\mathbf{AFLP_2} \cup \{\nu\}$, consists of the formulas of $AFLP_2$ supplemented by a special symbol of "empty" formula ν denoting the process which does nothing and successfully terminates. For any formula E of $AFLP_2$, the following is supposed: $E\|\nu = \nu\|E = E$ and $cont(\nu) = \emptyset$.
- The set of labels consists of conjunctions of $AFLP_2$ over alphabet Ev. Each conjunction G is a representation of lposet $\rho_G = \langle cont(G), \prec_G^+, l_G\rangle$, where $e \prec_G f \Leftrightarrow (e; f)$ is a conjunctive member of G, and $\prec_G^+$ is a transitive closure of $\prec_G$.
- A transition $E \xrightarrow{G} \tilde{E} \in \rightarrow_{TS}$ represents the transformation of the formula E into $\tilde{E}$ as a result of execution of lposet ρ_G.
- The initial state of the transition system is F.

The set of transitions of $TS(F)$ is defined by the following inference rules.

1. Elementary event

   1.1 $e \xrightarrow{e} \nu$

2. Elementary precedence

   2.1 $e; f \xrightarrow{e} f$

   2.2 $e; f \xrightarrow{e;f} \nu$

3. Concurrency

   3.1 $\dfrac{E \xrightarrow{G} \tilde{E}}{E\|F \xrightarrow{G} \tilde{E}\|F}$ $cont(G) \cap cont(F) = \emptyset$

   3.2 $\dfrac{F \xrightarrow{G} \tilde{F}}{E\|F \xrightarrow{G} E\|\tilde{F}}$ $cont(G) \cap cont(E) = \emptyset$

3.3 $\frac{E \xrightarrow{G} \tilde{E},\ F \xrightarrow{H} \tilde{F}}{E \| F \xrightarrow{G\|H} \tilde{E} \| \tilde{F}}$ $cont(G) \cap cont(\tilde{F}) = \emptyset,\ cont(H) \cap cont(\tilde{E}) = \emptyset$

4. Disjunction

4.1 $\frac{E \xrightarrow{G} \tilde{E}}{E \vee F \xrightarrow{G} \tilde{E}}$ $cont(G) \not\subseteq cont(F)$

4.2 $\frac{F \xrightarrow{G} \tilde{F}}{E \vee F \xrightarrow{G} \tilde{F}}$ $cont(G) \not\subseteq cont(E)$

4.3 $\frac{E \xrightarrow{G} \tilde{E},\ F \xrightarrow{H} \tilde{F}}{E \vee F \xrightarrow{G} \tilde{E} \vee \tilde{F}}$ $canon(G) \simeq canon(H)$

Let $\mathbf{TS}(F) = \{G \mid \exists \tilde{F} :\ F \xrightarrow{G} \tilde{F}\}$ be the *set of the formulas* of $TS(F)$. If $F \xrightarrow{G} \tilde{F}$ is a transition of $TS(F)$, and no inference rule is applied to $\tilde{F}$, then $\tilde{F}$ is a *terminal formula* of $TS(F)$. $\mathbf{TS_{max}}(F)$ denotes the *set of maximal formulas* of $TS(F)$.

The *operational semantics* of $AFLP_2$ is a mapping $\mathcal{O}_{FL2}$ from $\mathbf{AFLP_2}$ into the set of lposets defined as follows. Let E be a formula of $AFLP_2$ and $F \in canon(E)$. Then $\mathcal{O}_{FL2}[E] = \{\rho_{G\|\tilde{F}} \mid G \in \mathbf{TS_{max}}(F)\ \wedge\ F \xrightarrow{G} \tilde{F}\}$.

Theorem 3. *Let E be a formula of $AFLP_2$. Then $\mathcal{O}_{FL2}[E] = \mathcal{D}_{FL2}[E]$.*

3. Equivalences on weakly labelled A-nets

The descriptive calculus AFP_0 [3] is dual to AFP_2. Its formulas specify finite A-nets. Labelling on formulas of AFP_0 may be introduced, and a new algebra $AFLP_0$ may be obtained as a result. Then the formulas of $AFLP_0$ will specify finite *weakly labelled A-nets* (i.e. A-nets [6] with possibly noninjective labelling function).

Let us define a mapping $\Psi_L : \mathbf{AFLP_0} \to \mathbf{AFLP_2}$ as follows.

1. $\Psi_L(e) = e$,
2. $\Psi_L(E;_{FL0} F) = E;_{FL2} F$,
3. $\Psi_L(E\|_{FL0} F) = E\|_{FL2} F$,
4. $\Psi_L(E \nabla_{FL0} F) = E \nabla_{FL2} F$.

The symbol "$FL0$" marks the operations of $AFLP_0$, and the symbol "$FL2$" is used for operations of $AFLP_2$. The denotational semantics of $AFLP_0$ is a mapping $\mathcal{D}_{FL0}$ which associates with every formula E of the algebra a set of maximal causal subnets (O-subnets, in terms of [3]) of finite A-net N specified by the formula. Note that with every causal net $C = \langle P_C, T_C, F_C, l_C\rangle$ we can associate lposet $\rho_C = \langle T_C, F_C^+ \cap (T_C \times T_C), l_C\rangle$, where F_C^+ is a transitive closure of F_C.

Theorem 4. *Let E be a formula of $AFLP_0$ and F be a formula of $AFLP_2$ s.t. $F = \Psi_L(E)$. Then $\{\rho_C \mid C \in \mathcal{D}_{FL0}[E]\} = \mathcal{D}^+_{FL2}[F]$.*

Note that the mapping Ψ_L only replaces the operations of $AFLP_0$ by those of $AFLP_2$. Consequently, if we have a finite weakly labelled A-net N specified by $AFLP_0$-formula E, we can analyze its behaviour by means of the same formula E of $AFLP_2$.

From the literature, the following net equivalences are known. *Trace equivalences*: interleaving (denoted by $\equiv_i$) [5], step ($\equiv_s$) [9], partial word ($\equiv_{pw}$) [11], pomset ($\equiv_{pom}$) [4] and process ($\equiv_{pr}$) [11]. *Bisimulation equivalences*: interleaving ($\underline{\leftrightarrow}_i$) [8], step ($\underline{\leftrightarrow}_s$) [7], partial word ($\underline{\leftrightarrow}_{pw}$) [14], pomset ($\underline{\leftrightarrow}_{pom}$) [2] and process ($\underline{\leftrightarrow}_{pr}$) [1]. *ST-bisimulation equivalences*: interleaving ($\underline{\leftrightarrow}_{iST}$) [4], partial word ($\underline{\leftrightarrow}_{pwST}$) [14], pomset ($\underline{\leftrightarrow}_{pomST}$)[14] and process ($\underline{\leftrightarrow}_{prST}$) [11]. *History preserving bisimulation equivalences*: pomset ($\underline{\leftrightarrow}_{pomh}$) [10] and process ($\underline{\leftrightarrow}_{prh}$) [11]. *Conflict preserving equivalences*: prime event structure (PES) ($\equiv_{pes}$) [12], occurrence ($\equiv_{occ}$) [4], and *isomorphism* ($\simeq$). The interrelations of all the equivalences are depicted by graph in Figure 4 (without $\approx_{FL2}$ and $\approx_{FL2+}$) where no additional nontrivial arrow may be added [12].

Now we will consider the equivalences on weakly labelled A-nets. Unlike A-nets, where most of the equivalence notions are merged, the interrelations of the equivalences on weakly labelled A-nets and on nets without any restrictions are the same, and may be represented by the same graph.

Theorem 5. *Let N and N' be weakly labelled A-nets and $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}, \simeq\}$, $\star, \star\star \in \{i, s, pw, pom, pr, iST, pwST, pomST, prST, pomh, prh, pes, occ\}$. Then $N \leftrightarrow_\star N' \Rightarrow N \leftrightarrow_{\star\star} N'$ iff there exists a directed path from $\leftrightarrow_\star$ to $\leftrightarrow_{\star\star}$ in the graph in Figure 4 (without $\approx_{FL2}$ and $\approx_{FL2+}$).*

Proof. $\Leftarrow$ By Theorem 1 from [11].

$\Rightarrow$ The absence of additional nontrival arrows is proved by the following examples on weakly labelled A-nets.

- In Figure 1(a) $N \underline{\leftrightarrow}_i N'$, but $N \not\equiv_s N'$, since only in N' actions a and b cannot be executed concurrently.
- In Figure 1(e) $N \underline{\leftrightarrow}_{iST} N'$, but $N \not\equiv_{pw} N'$, since the net N corresponds to the pomset s.t. even a less sequential pomset cannot be executed in N'.
- In Figure 1(c) $N \underline{\leftrightarrow}_{pwST} N'$, but $N \not\equiv_{pom} N'$, since only in N an action b can depend on a.
- In Figure 1(d) $N \equiv_{pes} N'$, but $N \not\equiv_{pr} N'$, since only in N a-labelled transition has an additional output place.

- In Figure 1(b) $N \equiv_{pr} N'$, but $N \not\leftrightarrow_i N'$, since only in N an action a can happen so that b cannot happen after it.
- In Figure 2(a) $N \underline{\leftrightarrow}_{pr} N'$, but $N \not\underline{\leftrightarrow}_{iST} N'$, since only in N' an action a can begin working so that no b can start unless a finishes.
- In Figure 2(b) $N \underline{\leftrightarrow}_{prST} N'$, but $N \not\underline{\leftrightarrow}_{pomh} N'$, since only in N' actions a and b can happen so that the next action c must depend on a.
- In Figure 2(c) $N \underline{\leftrightarrow}_{prh} N'$, but $N \not\equiv_{pes} N'$, since only the labelled event structure (LES) that corresponds to N' has two conflict actions a.
- In Figure 2(d) $N \equiv_{occ} N'$, but $N \not\simeq N'$, since only in N' there is a c-labelled transition (which can never be fired).

Example 1. Let us consider the net N' in Figure 1(e). The corresponding formula of $AFLP_2$ is

$$E' = (e;(f_1 \bigtriangledown f_2))\|(e;(f_2 \bigtriangledown h_1))\|(g;(f_2 \bigtriangledown h_1))\|(g;(h_1 \bigtriangledown h_2))\|(f_1 \bigtriangledown h_2),$$

where $lab(e) = a$, $lab(f_1) = lab(f_2) = b$, $lab(g) = c$, $lab(h_1) = lab(h_2) = d$. Its canonical form is

$$F' = ((e;f_1)\|(e;h_1)\|(g;h_1)\|\bar{f}_2\|\bar{h}_2) \vee ((e;f_2)\|(g;f_2)\|(g;h_2)\|\bar{f}_1\|\bar{h}_1).$$

The labelled nondeterministic process specified by E' has two lposets presented in Figure 3. In this figure, the labels of events are in parentheses and the partial order is depicted by arrows.

Let us demonstrate that in $TS(F')$ from the initial formula F' the part of the first lposet which does not contain the event f_1 can be executed. In the following instances of transition rules of $TS(F)$, the numbers of applied rules are under arrows, and verification of conditions associated with the rules is in parentheses.

1. $e;f_1 \xrightarrow{e}_{2.1} f_1$
2. $e;h_1 \xrightarrow{e;h_1}_{2.2} \nu$
3. $(e;f_1)\|(e;h_1) \xrightarrow{e\|(e;h_1)}_{3.3} f_1\|\nu$ $(\{e\} \cap \emptyset = \emptyset,\ \{e,h_1\} \cap \{f_1\} = \emptyset)$
4. $g;h_1 \xrightarrow{g;h_1}_{2.2} \nu$
5. $(e;f_1)\|(e;h_1)\|(g;h_1) \xrightarrow{e\|(e;h_1)\|(g;h_1)}_{3.3} f_1\|\nu\|\nu$ $(\{e,h_1\} \cap \emptyset = \emptyset,\ \{g,h_1\} \cap \{f_1\} = \emptyset)$
6. $(e;f_1)\|(e;h_1)\|(g;h_1)\|\bar{f}_2 \xrightarrow{e\|(e;h_1)\|(g;h_1)}_{3.1} f_1\|\nu\|\nu\|\bar{f}_2$ $(\{e,g,h_1\} \cap \{\bar{f}_2\} = \emptyset)$
7. $(e;f_1)\|(e;h_1)\|(g;h_1)\|\bar{f}_2\|\bar{h}_2 \xrightarrow{e\|(e;h_1)\|(g;h_1)}_{3.1} f_1\|\nu\|\nu\|\bar{f}_2\|\bar{h}_2$ $(\{e,g,h_1\} \cap \{\bar{h}_2\} = \emptyset)$
8. $((e;f_1)\|(e;h_1)\|(g;h_1)\|\bar{f}_2\|\bar{h}_2) \vee ((e;f_2)\|(g;f_2)\|(g;h_2)\|\bar{f}_1\|\bar{h}_1)$ $\xrightarrow{e\|(e;h_1)\|(g;h_1)}_{4.1} f_1\|\nu\|\nu\|\bar{f}_2\|\bar{h}_2$ $(\{e,g,h_1\} \not\subseteq \{e,g,f_2,h_2,\bar{f}_1,\bar{h}_1\})$

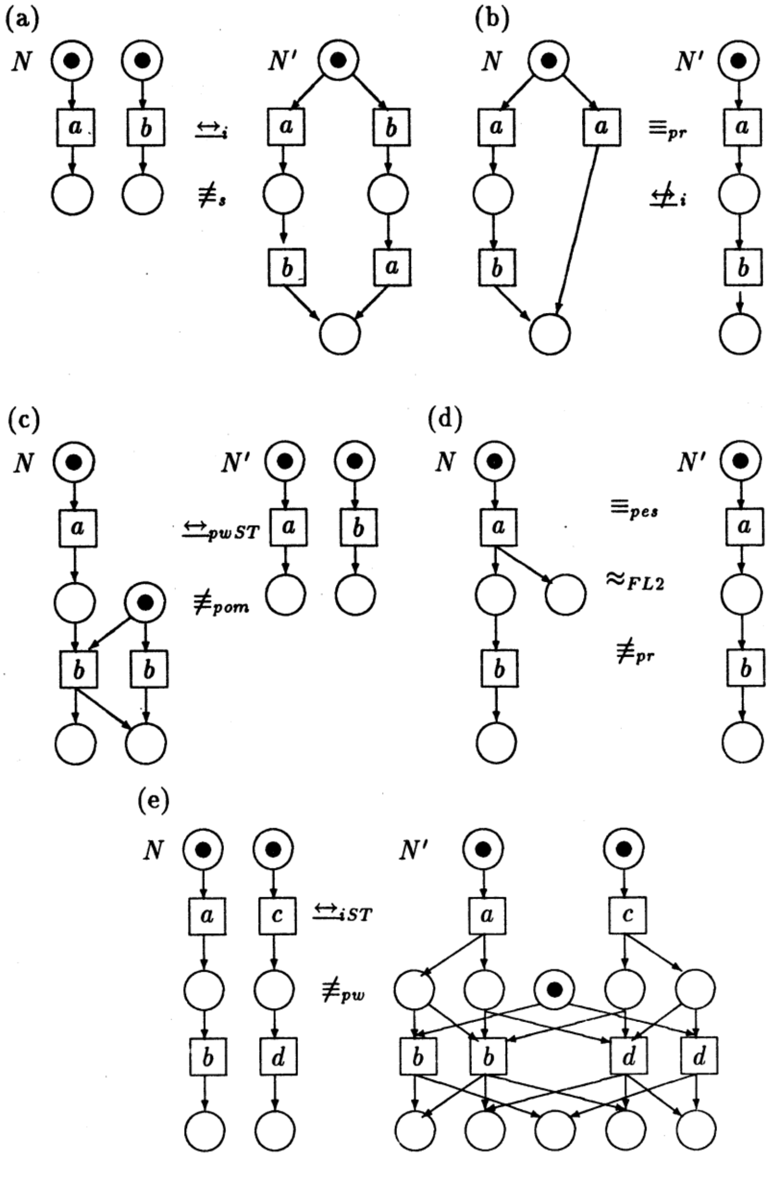

Figure 1. Examples of weakly labelled A-nets

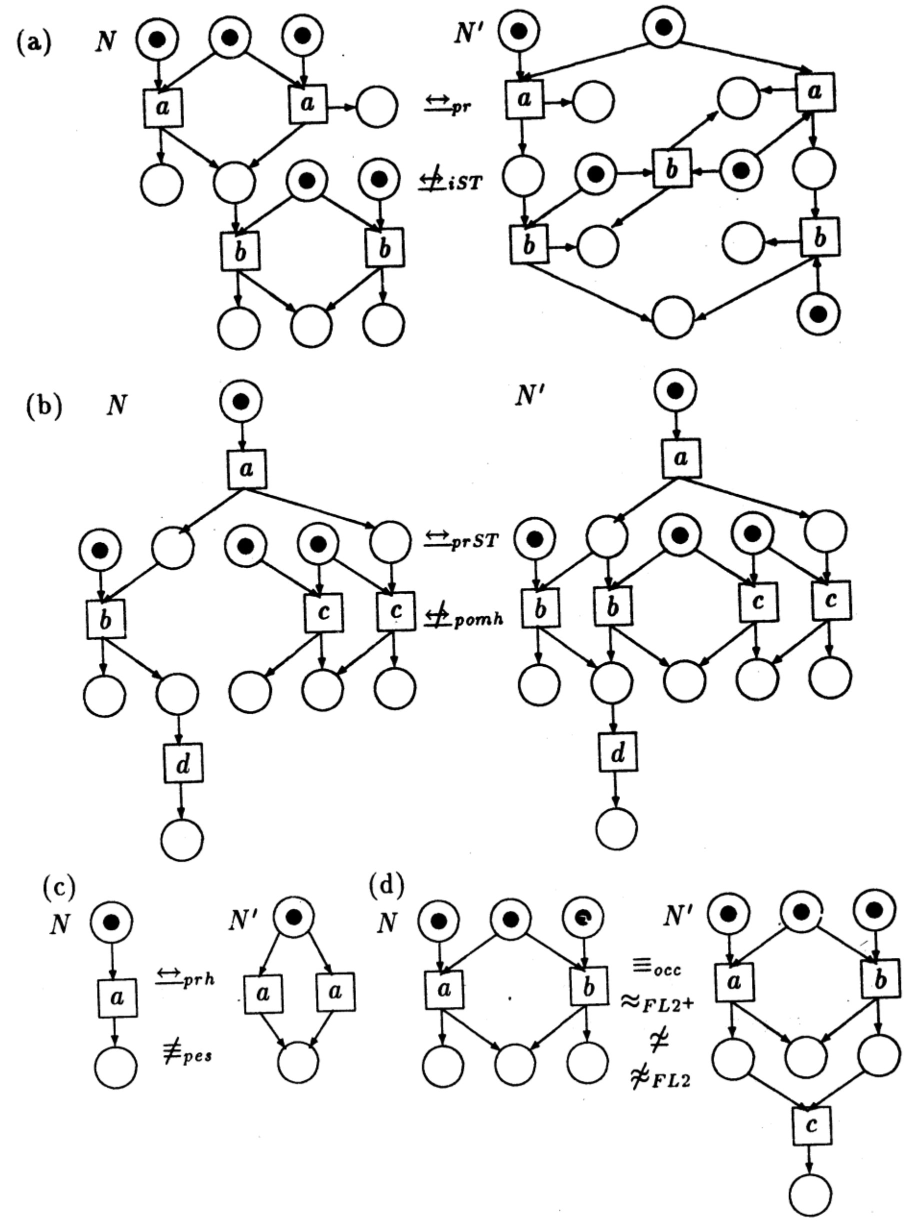

Figure 2. Examples of weakly labelled A-nets (continued)

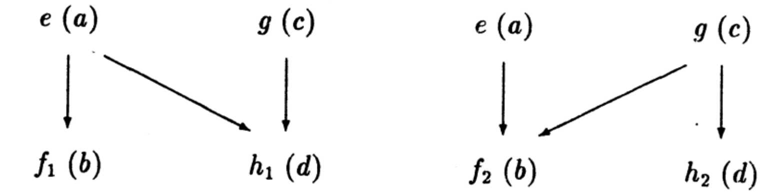

Figure 3. Set of lposets of the labelled nondeterministic process

Thus, $F' \xrightarrow{G} \tilde{F}'$ is a transition of $TS(F')$, where $G = e\|(e;h_1)\|(g;h_1)$, $\tilde{F}' = f_1\|\bar{f}_2\|\bar{h}_2$. Hence, in $TS(F')$ the lposet $\rho_G = \langle\{e,g,h_1\},\prec,l\rangle$ can be executed from the initial state, where $e \prec h_1$, $g \prec h_1$, $l(e) = a$, $l(g) = c$, $l(h_1) = d$.

As a result, we obtain the formula $\tilde{F}' = f_1\|\bar{f}_2\|\bar{h}_2$ containing the following information: in the present behaviour of the labelled nondeterministic process specified by E' the events f_2 and h_2 did not happen since some event(s) alternative to them (namely h_1) happened. In addition one can see that in the present state specified by $\tilde{F}'$ the event f_1 can happen. As a result, we will reach the state specified by the terminal formula $\bar{f}_2\|\bar{h}_2$ of $TS(F')$.

Let us find the denotational semantics of E'.

$$\mathcal{D}_{FL2}[E'] = \{\langle\{e,f_1,g,h_1,\bar{f}_2,\bar{h}_2\},\prec_1,l\rangle,\langle\{e,f_2,g,h_2,\bar{f}_1,\bar{h}_1\},\prec_2,l\rangle\},$$
$$\mathcal{D}^+_{FL2}[E'] = \{\langle\{e,f_1,g,h_1\},\prec_1,l_1\rangle,\langle\{e,f_2,g,h_2\},\prec_2,l_2\rangle\},$$

where $e \prec_1 f_1$, $e \prec_1 h_1$, $g \prec_1 h_1$, $e \prec_2 f_2$, $g \prec_2 f_2$, $g \prec_2 h_2$, $l(e) = l_1(e) = l_2(e) = a$, $l(f_1) = l(f_2) = l_1(f_1) = l_2(f_2) = b$, $l(g) = l_1(g) = l_2(g) = c$, $l(h_1) = l(h_2) = l_1(h_1) = l_2(h_2) = d$.

4. Interrelation of the net equivalences and equivalences of $AFLP_2$

Any finite A-net may be represented by a formula of AFP_0 using regularization algorithm [3]. Therefore, any finite weakly labelled A-net may be represented by a formula of $AFLP_0$ with the use of analogous algorithm. The mapping Ψ_L associates a formula of $AFLP_2$ with every formula of $AFLP_0$. Hence, one can associate a formula of $AFLP_2$ with every finite weakly labelled A-net. Given some formula equivalence, we consider two nets equivalent iff the formulas associated with these nets are equivalent.

Theorem 6. *Let N and N' be weakly labelled A-nets and* $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}, \simeq, \approx\}$, $\star$, $\star\star \in \{i,\ s,\ pw,\ pom,\ pr,\ iST,\ pwST,\ pomST,\ prST,\ pomh,\ prh,\ pes,\ occ,$

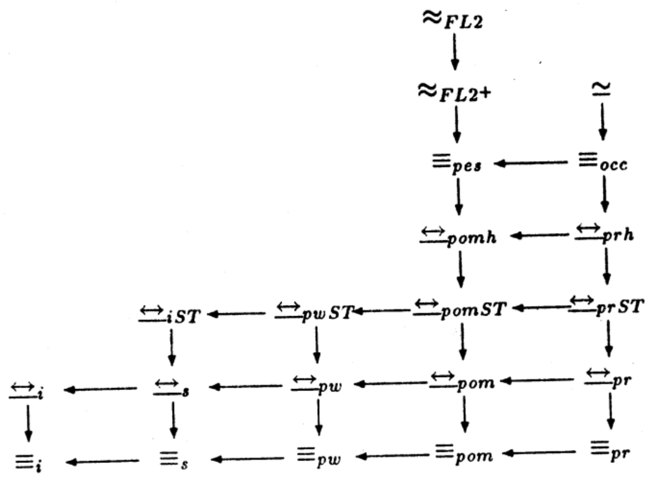

Figure 4. The interrelations of the net equivalences and equivalences of $AFLP_2$

$FL2$, $FL2^+\}$. Then $N \leftrightarrow_\star N' \;\Rightarrow\; N \leftrightarrow_{\star\star} N'$ iff there exists a directed path from $\leftrightarrow_\star$ to $\leftrightarrow_{\star\star}$ in the graph in Figure 4.

5. Analogs of the net equivalences on formulas of $AFLP_2$

5.1. Process subformulas

If $F \in canon(E)$ for some formula E of $AFLP_2$, then the *set of the process subformulas* of E is defined as follows: $PSF(E) = \{G \mid G \in canon(H) \wedge H \in \mathbf{TS}(F)\} \cup \{\nu\}$. We consider the process subformulas up to isomorphism.

We write $G \xrightarrow{\hat{G}} \tilde{G}$, if $F \xrightarrow{H} F'$, $F' \xrightarrow{\hat{H}} F''$ and $F \xrightarrow{\tilde{H}} F''$ are transitions of $TS(F)$ and $G \in canon(H)$, $\hat{G} \in canon(\hat{H})$, $\tilde{G} \in canon(\tilde{H})$. In such a case the process subformula $\tilde{G}$ is an *extension* of G *by* $\hat{G}$, and $\hat{G}$ is an *extending* process subformula. Let $\forall G \in PSF(E)\ \nu \xrightarrow{G} G$. We write $G \rightarrow \tilde{G}$, if $G \xrightarrow{\hat{G}} \tilde{G}$ fore some $\hat{G}$.

$\tilde{G}$ is an extension of G *by one action*, if $G \xrightarrow{\hat{G}} \tilde{G}$ and $\hat{G} = e$, $e \in Ev$. In such a case we write $G \xrightarrow{e} \tilde{G}$ or $G \xrightarrow{a} \tilde{G}$, if $lab(e) = a \in Act$.

$\tilde{G}$ is an extension of G *by a multiset of actions* or a *step*, if $G \xrightarrow{\hat{G}} \tilde{G}$ and $\hat{G} = \|_{i=1}^{n} e_i$, $e_i \in Ev$ $(1 \leq i \leq n)$. In such a case we write $G \xrightarrow{U} \tilde{G}$ or $G \xrightarrow{A} \tilde{G}$, if $U = \{e_1, \ldots, e_n\}$, $A = \{lab(e_1), \ldots, lab(e_n)\} \in \mathcal{M}(Act)$ (here $\mathcal{M}(Act)$ is a set of all multisets over Act).

5.2. Trace equivalences

An *interleaving trace* of a formula E is a sequence $a_1 \cdots a_n \in Act^*$ s.t. $\nu \stackrel{a_1}{\to} G_1 \stackrel{a_2}{\to} \dots \stackrel{a_n}{\to} G_n$, where $G_i \in PSF(E)$ $(1 \le i \le n)$. $SeqTraces(E)$ denotes a *set of all interleaving traces* of E. Two formulas E and E' are *interleaving trace equivalent*, denoted by $E \equiv_i E'$, iff $SeqTraces(E) = SeqTraces(E')$.

A *step trace* of a formula E is a sequence $A_1 \cdots A_n \in (\mathcal{M}(Act))^*$ s.t. $\nu \stackrel{A_1}{\to} G_1 \stackrel{A_2}{\to} \dots \stackrel{A_n}{\to} G_n$, where $G_i \in PSF(E)$ $(1 \le i \le n)$. $StepTraces(E)$ denotes a *set of all step traces* of E. Two formulas E and E' are *step trace equivalent*, denoted by $E \equiv_s E'$, iff $StepTraces(E) = StepTraces(E')$.

A *pomset trace* of a formula E is a pomset ρ which is an isomorphism class of lposet ρ_G for $G \in PSF(E)$. We write $\rho \sqsubseteq \rho'$, if $\rho_G \sqsubseteq \rho_{G'}$ for $\rho_G \in \rho$ and $\rho_{G'} \in \rho'$. In such a case we say that ρ is *less sequential* or *more parallel* than ρ'. $Pomsets(E)$ denotes a *set of all pomset traces* of E. Two formulas E and E' are *partial word trace equivalent*, denoted by $E \equiv_{pw} E'$, iff $Pomsets(E) \sqsubseteq Pomsets(E')$ and $Pomsets(E') \sqsubseteq Pomsets(E)$, i.e. for any $\rho' \in Pomsets(E')$ there exists $\rho \in Pomsets(E)$ s.t. $\rho \sqsubseteq \rho'$ and vice versa. Two formulas E and E' are *pomset trace equivalent*, denoted by $E \equiv_{pom} E'$, iff $Pomsets(E) = Pomsets(E')$.

5.3. Bisimulation equivalences

5.3.1. Usual bisimulations

Let $\mathcal{R} \subseteq PSF(E) \times PSF(E')$.

$\mathcal{R}$ is a *$\star$-bisimulation* between E and E', $\star \in$ {*interleaving, step, partial word, pomset*}, denoted by $\mathcal{R} : E \underline{\leftrightarrow}_\star E'$, $\star \in \{i, s, pw, pom\}$, iff:

1. $(\nu, \nu) \in \mathcal{R}$;
2. $(G, G') \in \mathcal{R}$, $G \stackrel{\hat{G}}{\longrightarrow} \tilde{G}$, and if

   (a) $|cont(\hat{G})| = 1$, if $\star = i$;
   (b) $\prec_{\hat{G}} = \emptyset$, if $\star = s$;

   then $\exists \tilde{G}' : G' \stackrel{\hat{G}'}{\longrightarrow} \tilde{G}'$, $(\tilde{G}, \tilde{G}') \in \mathcal{R}$ and

   (a) $\rho_{\hat{G}'} \sqsubseteq \rho_{\hat{G}}$, if $\star = pw$;
   (b) $\rho_{\hat{G}} \simeq \rho_{\hat{G}'}$, if $\star \in \{i, s, pom\}$.

3. The same as item 2, but the roles of E and E' are reversed.

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

5.3.2. ST-process subformulas

An *ST-process subformula* of a formula E is a pair (G, H) s.t. $G, H \in PSF(E), H \xrightarrow{K} G$ and $\forall e, f \in cont(G)\ e \prec_G f \Rightarrow e \in cont(H)$. In such a case G is the process subformula which has started, i.e. all events of G have started. The process subformula H corresponds to that part of G which has finished and K corresponds to the part which has started but has not finished yet. Clearly, $\prec_K = \emptyset$. $ST - PSF(E)$ denotes a *set of all ST-process subformulas* of E.

Let (ν, ν) be an *initial ST-process subformula.* Let $(G, H), (\tilde{G}, \tilde{H}) \in ST - PSF(E)$. We write $(G, H) \to (\tilde{G}, \tilde{H})$, if $G \to \tilde{G}$ and $H \to \tilde{H}$.

5.3.3. ST-bisimulations

Let $\mathcal{R} \subseteq ST - PSF(E) \times ST - PSF(E') \times \mathcal{B}$, where $\mathcal{B} = \{\beta \mid \beta : cont(G) \to cont(G'),\ G \in PSF(E),\ G' \in PSF(E')\}$.

$\mathcal{R}$ is a $\star$*-ST-bisimulation* between E and E', $\star \in$\{*interleaving, partial word, pomset*\}, denoted by $\mathcal{R} : E \underline{\leftrightarrow}_{\star ST} E'$, $\star \in \{i, pw, pom\}$, iff:

1. $((\nu, \nu), (\nu, \nu), \emptyset) \in \mathcal{R}$;
2. $((G, H), (G', H'), \beta) \in \mathcal{R} \Rightarrow \beta : \rho_G \approx \rho_{G'}$ and $\beta(cont(H)) = cont(H')$;
3. $((G, H), (G', H'), \beta) \in \mathcal{R},\ (G, H) \to (\tilde{G}, \tilde{H}) \Rightarrow$ $\exists \tilde{\beta},\ (\tilde{G}', \tilde{H}') :\ (G', H') \to (\tilde{G}', \tilde{H}'),\ \tilde{\beta}|_{cont(G)}$ $= \beta,\ ((\tilde{G}, \tilde{H}), (\tilde{G}', \tilde{H}'), \tilde{\beta}) \in \mathcal{R}$, and if $H \xrightarrow{K} \tilde{G},\ H' \xrightarrow{K'} \tilde{G}'$ then:
   (a) $(\tilde{\beta}|_{cont(K)})^{-1} : \rho_{K'} \sqsubseteq \rho_K$, if $\star = pw$;
   (b) $\tilde{\beta}|_{cont(K)} : \rho_K \simeq \rho_{K'}$, if $\star = pom$;
4. The same as item 3, but the roles of E and E' are reversed.

Two formulas E and E' are $\star$*-ST-bisimulation equivalent*, $\star \in$ \{*interleaving, partial word, pomset*\}, denoted by $E \underline{\leftrightarrow}_{\star ST} E'$, $\star \in \{i, pw, pom\}$, iff $\exists \mathcal{R} : E \underline{\leftrightarrow}_{\star ST} E'$.

5.3.4. History preserving bisimulations

Let $\mathcal{R} \subseteq PSF(E) \times PSF(E') \times \mathcal{B}$, where

$$\mathcal{B} = \{\beta \mid \beta : cont(G) \to cont(G'),\ G \in PSF(E),\ G' \in PSF(E')\}.$$

$\mathcal{R}$ is a *pomset history preserving bisimulation* between E and E', denoted by $\mathcal{R} : E \underline{\leftrightarrow}_{pomh} E'$, iff:

1. $(\nu, \nu, \emptyset) \in \mathcal{R}$;
2. $(G, G', \beta) \in \mathcal{R} \Rightarrow \beta : \rho_G \simeq \rho_{G'}$;

3. $(G, G', \beta) \in \mathcal{R},\ G \to \tilde{G} \ \Rightarrow\ \exists \tilde{\beta},\ \tilde{G}' : G' \to \tilde{G}'$, $\tilde{\beta}|_{cont(G)} = \beta,\ (\tilde{G}, \tilde{G}', \tilde{\beta}) \in \mathcal{R}$;
4. The same as item 3, but the roles of E and E' are reversed.

Two formulas E and E' are *pomset history preserving bisimulation equivalent*, denoted by $E \underline{\leftrightarrow}_{pomh} E'$, iff $\exists \mathcal{R} : E \underline{\leftrightarrow}_{pomh} E'$.

5.4. Conflict preserving equivalences

Let E be a formula of $AFLP_2$ and $F = \vee_{i=1}^{n} F_i \in canon(E)$. On the basis of F we can construct a labelled event structure (LES) $\xi_F = \langle cont^+(F), \prec_F, \#_F, l_F|_{cont^+(F)} \rangle$, where

- $e \prec_F f \ \Leftrightarrow\ \exists i\ (1 \le i \le n)\ (e; f)$ is a subformula of F_i;
- $e \#_F f \ \Leftrightarrow\ \forall i\ (1 \le i \le n)$ e and f do not occur in F_i together.

$\mathcal{E}(E)$ denotes an isomorphism class of ξ_F for $F \in canon(E)$. Two formulas E and E' are *prime event structure (PES-) equivalent*, denoted by $E \equiv_{pes} E'$, if $\mathcal{E}(E) = \mathcal{E}(E')$.

5.5. The interrelations of the net equivalences with their analogs in $AFLP_2$

Theorem 7. *Let E (E') be a formula of $AFLP_2$ corresponding to the finite weakly labelled A-net N (N') and $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}\}$, $\star \in \{i, s, pw, pom, iST, pwST, pomST, pomh, pes\}$. Then $N \leftrightarrow_\star N' \ \Leftrightarrow\ E \leftrightarrow_\star E'$.*

Clearly, the interrelations of the equivalences of $AFLP_2$ and analogs of the net equivalences are depicted by a graph in Figure 4 without process equivalences (since they are unexpressible in terms of process algebras).

The question arises after defining the analogs of the net equivalences on formulas of $AFLP_2$, whether some of these equivalences are congruences w.r.t. operations of the algebra. Let us consider the following example.

Example 2. Let $E = e \bigtriangledown f$ and $E' = (e \bigtriangledown f) \| e \| f$, where $lab(e) = a$, $lab(f) = b$, $lab(g) = c$. We have $E \approx_{FL2+} E'$, but $E; g \not\equiv_i E'; g$, since $PSF(E; g) = \{\nu, e, f, (e; g), (f; g)\}$, whereas $PSF(E'; g) = \{\nu, e, f\}$. Therefore $SeqTraces(E; g) = \{a, b, ac, bc\}$, whereas $SeqTraces(E'; g) = \{a, b\}$.

Note that formulas $E; g$ and $E'; g$ are associated with nets N and N' in Figure 5. We proved an accordance of the net equivalences with their analogs in $AFLP_2$. Hence, the fact $E; g \not\equiv_i E'; g$ can be derived from the consideration of N and N', for which $N \not\equiv_i N'$, since only in N' the action c can never happen.

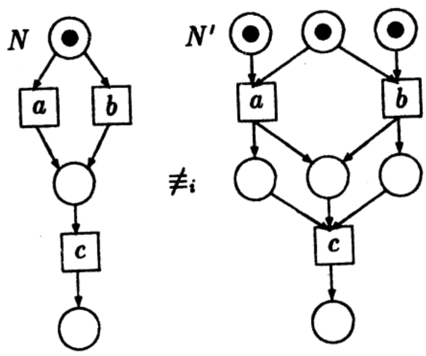

Figure 5. A-nets from example of congruence

Consequently, none of the considered equivalences on formulas of $AFLP_2$ is a congruence except $\approx_{FL2}$, i.e. $\approx_{FL2}$ is the weakest equivalence which is a congruence.

6. Conclusion

In this paper the new calculus $AFLP_2$ was presented for the description and analysis of labelled nondeterministic processes. The interrelations of the net equivalences and equivalences of the algebra were established. Analogs of the net equivalences were introduced on formulas of $AFLP_2$. It gives a possibility to consider the processes specified by formulas of the algebra at different levels of abstraction without refering to their net representations. Hence, algebra $AFLP_2$ possesses rather powerful tools of dealing with nondeterministic finite processes.

Acknowledgements. I would like to thank my scientific supervisor Dr. I.B. Virbitskaite for useful discussions which have played an important role in the research.

References

[1] C. Autant, Ph. Schnoebelen, *Place bisimulations in Petri nets*, Lect. Notes . Comput. Sci., **616**, 1992, 45–61.

[2] G. Boudol, I. Castellani, *On the semantics of concurrency: partial orders and transition systems*, Lect. Notes Comput. Sci., **249**, 1987, 123–137.

[3] L.A. Cherkasova, *Posets with non-actions: a model for concurrent nondeterministic processes*, Arbeitspapiere der GMD, Bonn, Germany, **403**, 1989, 68 p.

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

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

[6] V.E. Kotov, *Petri Nets*, Moscow, Nauka, 1984, 160 p. (in Russian).

[7] M. Nielsen, P.S. Thiagarajan, *Degrees of non-determinizm and concurrency: a Petri net view*, Lecture Notes in Computer Science, **181**, 1984, 89–117.

[8] D.M.R. Park, *Concurrency and automata on infinite sequences*, Lecture Notes in Computer Science, **104**, 1981, 167–183.

[9] L. Pomèllo, *Some equivalence notions for concurrent systems. An overview*, Lecture Notes in Computer Science, **222**, 1986, 381–400.

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

[11] I.V. Tarasyuk, *An investigation of equivalence notions on some subclasses of Petri nets*, Bulletin of Novosibirsk Computing Center, **3**, 1995, 89–101.

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

[13] I.V. Tarasyuk, *Algebra $AFLP_2$: a calculus of labelled nondeterministic processes*, Hildesheimer Informatik-Bericht, Institut für Informatik, Universität Hildesheim, Hildesheim, Germany, **4/96**, part 2, January 1996, 18 p.

[14] W. Vogler, *Bisimulation and action refinement*, Lecture Notes in Computer Science, **480**, 1991, 309–321.