Algebra $AFLP_2$: a calculus of labelled nondeterministic processes *

Igor V. Tarasyuk

Institute of Informatics Systems,
Siberian Division of the Russian Academy of Sciences,
6, Lavrentieva ave.,
630090, Novosibirsk, Russia
e-mail: itar@iis.nsk.su

November 3, 2003

Abstract

Algebra $AFLP_2$ is proposed which is an extension of algebra AFP_2 by labelling function. Denotational and operational semantics are presented. Interrelation of the net equivalences from [19, 20, 21] with equivalences of the algebra is considered. Analogs of the net equivalences are defined on formulas of $AFLP_2$, and the accordance of these equivalences with their prototypes is established.

Keywords & phrases: process algebras, labelling, denotational semantics, operational semantics, Petri nets, A-nets, behavioural equivalences, bisimulation, congruence.

1 Introduction

The importance of a proper understanding of the basic issues concerning the behaviour of systems with independent (concurrent or distributed) execution of components became obvious over the last decades. For specification of concurrent systems and processes and investigation of their behavioural properties a number of formal models were proposed. In algebraic calculi, which are one of such models, a process is specified by an algebraic formula, and the verification of process properties is accomplished by means of equivalences, axioms and inference rules. As mentioned in [22], the advantages of process algebras are: their modularity (by definition), well-developed equivalence notions, algebraic laws and complete proof systems.

In [5, 6] a number of algebras of concurrent nondeterministic processes (AFP_0, AFP_1, AFP_2) were proposed. Descriptive and analytical algebra AFP_2 (Algebra of Finite Processes) with semantics based on posets with non-actions and deadlocked actions combines mechanisms both for specification of processes and for the derivation of their behavioural properties. The algebra is close to such calculi as $TCSP$ [4] and CCS [13].

It has three basic operations (*alternative, concurrency, precedence*) and three auxilary ones (*disjunction, "not occur", "mistaken not occur"*). Comparing with CCS, one can see that CCS does not contain the auxilary operations of AFP_2. In addition, alternative and precedence operations in AFP_2 are more flexible than nondeterministic choice and prefix in CCS respectively.

Formulas of AFP_2 are combined by the operations from symbols of three alphabets (*actions, non-actions, deadlocked actions*). Non-actions with disjunction and "not occur" operations are used to preserve information about nondeterminism in sequential components of a process. Deadlocked actions with operation "mistaken not occur" are used to represent some contradictions in a process specification.

Unlike AFP_2, CCS does not contain non-actions and deadlocked actions, but it has co-actions which are used for *binary* synchronization. An advantage of AFP_2 is a *not binary* mechanism of action synchronization by names which is close to the net one. In accordance with the mechanism all equally named actions are synchronized, and the only event is considered to correspond to these actions. It permits us to specify the processes which cannot be represented (or it is not trivial to do) by formulas of other algebras (for example, CCS or algebra of event structures [2, 3]), where the unique event is associated with each action occurrence in a formula [5]. But it is impossible to specify within AFP_2 the process with two concurrent actions having the same name.

We introduce an algebra $AFLP_2$ (Algebra of Finite Labelled Processes) on the basis of AFP_2 by imposing the global labelling to event symbols which are combined into formulas. Hence, the formulas of $AFLP_2$ specify *labelled*

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

nondeterministic processes where some different events may be equally labelled, unlike AFP_2-formulas. Thus, using $AFLP_2$ we can specify a much wider class of processes than in AFP_2.

In $AFLP_2$, denotational and operational semantics are introduced on the basis of labelled posets (lposets) with non-events and deadlocked events, and their coincidence is established. The semantical equivalence of $AFLP_2$-formulas is defined, and sound and complete axiom set corresponding to the equivalence is presented.

It is demonstrated that by means of $AFLP_2$ one can analyze the behaviour of weakly labelled A-nets (i.e. A-nets [10, 12] which may have noninjective labelling function). The net equivalences considered in [19, 20, 21] are treated on this subclass of Petri nets. Semantical equivalences (usual and observational) of $AFLP_2$ are transferred to weakly labelled A-nets, and their interrelation with the net equivalences is examined.

Analogs of the net equivalences are introduced on $AFLP_2$-formulas, and their accordance with original net equivalences is proved. So, we can add simple definitions of the basic net equivalences on formulas of $AFLP_2$ to the advantages of the algebra.

At last the fact is established that semantical equivalence of $AFLP_2$ is the only one which is a congruence w.r.t. operations of the algebra.

The paper is organized as follows. In Section 2 algebra $AFLP_2$ is presented. In Subsection 2.1 a syntax of the algebra is introduced. Subsection 2.2 is devoted to denotational semantics of the algebra. Axiomatization of equivalence based on denotational semantics is proposed in Subsection 2.3. Completeness of the axiom system is proved using the notion of canonical form of $AFLP_2$-formulas, which is defined in Subsection 2.4. Operational semantics is presented in Subsection 2.5. Equivalences from [19, 20, 21] are treated on weakly labelled A-nets in Section 3. Interrelation of these net equivalences and semantical equivalences of $AFLP_2$ which have been transferred to nets is studied in Section 4. Analogs of the net equivalences are defined on $AFLP_2$-formulas in Section 5. In Subsection 5.1 process subformulas are introduced. In Subsection 5.2 trace, in Subsection 5.3 bisimulation, and in Subsection 5.4 conflict respecting equivalences are defined. Subsection 5.4 is devoted to the interrelation of the net equivalences with their analogs in $AFLP_2$. Section 6 is a conclusion which contains a review of the results obtained and some directions of further research.

Let us note that the 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 [19, 20].

2 Algebra $AFLP_2$

2.1 Syntax

Let $Ev = \{e, f, \ldots\}$ be an alphabet of symbols of *(ordinary) 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 $\widehat{Ev} = Ev \cup \overline{Ev} \cup \Delta_{Ev}$. Symbols of $\widehat{Ev}$ are combined into formulas by operations ; (*precedence*), $\bigtriangledown$ (*exclusive or, alternative*), $\|$ (*concurrency*), $\vee$ (*disjunction, union*), $\rceil\rceil$ (*"not occur"*), $\tilde{\rceil\rceil}$ (*"mistaken not occur"*). We introduce $Act = \{a, b, \ldots\}$, an alphabet of *action* symbols *(labels)*. A *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 a basis $\widehat{Ev}$ is defined by the following production system.

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

Here $e \in Ev, \; \bar{e} \in \overline{Ev}, \; \delta_e \in \Delta_{Ev}$ are *elementary formulas.* We denote by $\mathbf{AFLP_2}$ *a set of all formulas* of $AFLP_2$.

Let E be a formula of $AFLP_2$. A set $Ev(E)$ is defined as follows.

1. $Ev(e) = Ev(\bar{e}) = Ev(\delta_e) = e;$
2. $Ev(\neg E) = Ev(E), \; \neg \in \{\rceil\rceil, \tilde{\rceil\rceil}\};$
3. $Ev(E \circ F) = Ev(E) \cup Ev(F), \; \circ \in \{;, \|, \bigtriangledown, \vee\}.$

Let us introduce also $\overline{Ev}(E) = \{\bar{e} \mid e \in Ev(E)\}$, $\Delta_{Ev}(E) = \{\delta_e \mid e \in Ev(E)\}$ and $\widehat{Ev}(E) = Ev(E) \cup \overline{Ev}(E) \cup \Delta_{Ev}(E)$.

One can associate with every formula E of $AFLP_2$ a *local labelling function* $l_E = lab\lceil_{Ev(E)}$, which labels event symbols of the formula.

Let us define a *contents* of E, $cont(E)$, as follows.

1. $cont(e) = e, \; cont(\bar{e}) = \bar{e}, \; cont(\delta_e) = \delta_e;$
2. $cont(\neg E) = cont(E), \; \neg \in \{\rceil\rceil, \tilde{\rceil\rceil}\};$

3. $cont(E \circ F) = cont(E) \cup cont(F),\ \circ \in \{;, \|, \bigtriangledown, \vee\}$.

We introduce also $cont^+(E) = cont(E) \cap Ev$ — a *set of events* of E, $cont^-(E) = cont(E) \cap \overline{Ev}$ — a *set of non-events* of E, $\Delta_{cont}(E) = cont(E) \cap \Delta_{Ev}$ — a *set of deadlocked events* of E.

Let E and E' be formulas of $AFLP_2$. E and E' are *isomorphic*, notation $E \simeq E'$, if these formulas coincide up to associativity rules w.r.t. $;, \|, \vee, \bigtriangledown$ and commutativity rules w.r.t. $\|, \vee, \bigtriangledown$.

Example 1 $(e\|f\|\bar{g}) \vee (g\|\bar{e}\|\bar{f}) \simeq (\bar{e}\|\bar{f}\|g) \vee (f\|e\|\bar{g})$.

2.2 Denotational semantics

A *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*.

Let us note that $Ev(X) = \{e \mid (e \in X) \vee (\bar{e} \in X) \vee (\delta_e \in X)\}$. We define also $\overline{Ev}(X) = \{\bar{e} \mid e \in Ev(X)\}$, $\Delta_{Ev}(X) = \{\delta_e \mid e \in Ev(X)\}$ and $\widehat{Ev}(X) = Ev(X) \cup \overline{Ev}(X) \cup \Delta_{Ev}(X)$. We denote by $X^+ = X \cap Ev$ — a *subset of events* of X, $X^- = X \cap \overline{Ev}$ — a *subset of non-events* of X, $\Delta_X = X \cap \Delta_{Ev}$ — a *subset of deadlocked events* of X.

Since now we will consider lposets which satisfy the following conditions.

1. e, $\bar{e}$ and δ_e do not occur in X together, i.e. e occurs in X, or $\bar{e}$, or δ_e;
2. partial order $\prec$ is irreflexive;
3. $\forall x, y \in X^- \cup \Delta_X\ (x \not\prec y)\&(y \not\prec x)$, i.e. all elements of $X^- \cup \Delta_X$ are incomparable;
4. $\forall x \in X^+\ \forall y \in X^- \cup \Delta_X\ (x \not\prec y)\&(y \not\prec x)$, i.e. all elements of X^+ and $X^- \cup \Delta_X$ are incomparable.

We write $\rho \lhd \rho'$ when ρ is a *strict prefix* of ρ' (in usual sense) and $\rho \unlhd \rho'$ when ρ is a *prefix* of ρ', i.e. $\rho \lhd \rho'$ or $\rho = \rho'$. The *modified union* of lposets is defined as follows.

$$\rho \tilde{\cup} \rho' = \begin{cases} \rho, & \rho' \unlhd \rho; \\ \rho', & \rho \unlhd \rho'; \\ \{\rho, \rho'\}, & \text{otherwise.} \end{cases}$$

The modified union absorbs the computations which can be continued in another behaviour (deterministic subprocess) of nondeterministic process, and equal computations.

For defining denotational semantics of $AFLP_2$ the following operations over lposets are introduced: ; (*precedence*), $\|$ (*concurrency*), $\bigtriangledown$ (*alternative*), $\rceil\!\rceil$ (*not occur*), $\tilde{\rceil\!\rceil}$ (*mistaken not occur*). If lposet ρ, constructed by means of these operations, does not satisfy the conditions 1-4 mentioned above, we "correct" it using new auxilary *regularization* operation $[\rho]$. This operation singles out the maximal prefix of ρ "before" some contradictions in process specification arise. All the events specified in this process behaviour occuring "after" these contradictions, are announced as the deadlocked events.

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

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

Then $[\rho] = \langle D, \emptyset, l\lceil_{Ev(D)}\rangle \cup \langle Y, \prec \cap (Y \times Y), l\lceil_{Ev(Y)}\rangle$, where $Y = X \setminus \widehat{Ev}(D)$. It is easy to verify that if lposet ρ satisfies the conditions 1-4, then $[\rho] = \rho$.

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

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

Mistaken not occur $\tilde{\rceil\!\rceil}\rho = \langle \Delta_{Ev}(X), \emptyset, l\rangle$.

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

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

Alternative $\rho \bigtriangledown \rho' = [\langle X \cup \overline{Ev}(X'), \prec, l \cup l'\rangle]\tilde{\cup}[\langle \overline{Ev}(X) \cup X', \prec', l \cup l'\rangle]$. It should be noted that $\rho \bigtriangledown \rho'$ is not lposet but a set of two lposets describing alternative behaviours of nondeterministic process, i.e. if ρ occurs, then ρ' does not occur, and vice versa.

We extend the operations introduced above to sets of lposets in the natural way. 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 $\neg \in \{\rceil, \tilde{\rceil}\}$ and $\mathcal{P} \circ \mathcal{P}' = \tilde{\cup}_{i=1}^{n}(\tilde{\cup}_{j=1}^{m}\rho_i \circ \rho'_j)$, where $\circ \in \{;, \|, \bigtriangledown\}$.

A nondeterministic concurrent process is characterized by the set of lposets, associated with all its possible alternative behaviours. *Denotational semantics* of $AFLP_2$ is a mapping $\mathcal{D}_{FL2}$ from $\mathbf{AFLP_2}$ into set of lposets, defined as follows.

1. $\mathcal{D}_{FL2}[e] = \langle\{e\}, \emptyset, l_e\rangle,\ \mathcal{D}_{FL2}[\bar{e}] = \langle\{\bar{e}\}, \emptyset, l_e\rangle,\ \mathcal{D}_{FL2}[\delta_e] = \langle\{\delta_e\}, \emptyset, l_e\rangle$, where $l_e = (e, lab(e))$;
2. $\mathcal{D}_{FL2}[\neg E] = \neg\mathcal{D}_{FL2}[E],\ \neg \in \{\rceil, \tilde{\rceil}\}$;
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 $AFLP_2$-formulas E and E' are *equivalent w.r.t. denotational semantics* $\mathcal{D}_{FL2}$, notation $E \approx_{\mathcal{D}_{FL2}} E'$ iff $\mathcal{D}_{FL2}[E] = \mathcal{D}_{FL2}[E']$.

If $\rho = \langle X, \prec, l\rangle$ is an lposet, then $\rho^+ = \langle X^+, \prec, l\lceil_{X^+}\rangle$ is the lposet, corresponding to the "observable" part of ρ over Ev. For every formula E of $AFLP_2$ $\mathcal{D}_{FL2}[E] = \cup_{i=1}^{n}\rho_i$ is a set of lposets, which characterize a labelled nondeterministic prosess specified by the formula. *"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 w.r.t. denotational semantics* $\mathcal{D}_{FL2}$, notation $E \approx_{\mathcal{D}^+_{FL2}} E'$ iff $\mathcal{D}^+_{FL2}[E] = \mathcal{D}^+_{FL2}[E']$.

A *context* $\mathcal{C}$ is an expression which is a formula of $AFLP_2$, where zero or more subformulas are replaced by "holes" to be filled by other $AFLP_2$-formulas [6]. $\mathcal{C}[E]$ means putting of the formula E in each such "hole".

Proposition 1 *For any formulas E and E' of $AFLP_2$ $E \approx_{\mathcal{D}_{FL2}} E' \ \Leftrightarrow\ \forall\mathcal{C}\ \mathcal{C}[E] \approx_{\mathcal{D}_{FL2}} \mathcal{C}[E']$.*

Proof. As Lemma 5.1 in [6]. □

Thus, $\approx_{\mathcal{D}_{FL2}}$ is a congruence w.r.t. operations of $AFLP_2$. Let us note that $\approx_{\mathcal{D}^+_{FL2}}$ is not a congruence. It is demonstrated by the following example.

Example 2 *Let $E = e \bigtriangledown f$, $E' = (e \bigtriangledown f)\|e\|f$ and $lab(e) = a,\ lab(f) = b,\ lab(g) = c$. Then $\mathcal{D}^+_{FL2}[E] = \mathcal{D}^+_{FL2}[E'] = \{\langle\{e\}, \emptyset, l_1\rangle, \langle\{f\}, \emptyset, l_2\rangle$, where $l_1(e) = a,\ l_2(f) = b$ and $E \approx_{\mathcal{D}^+_{FL2}} E'$. But $\mathcal{D}^+_{FL2}[E; g] = \{\langle\{e, g\}, \prec_1, l_1\rangle, \langle\{f, g\}, \prec_2, l_2\rangle\}$, whereas $\mathcal{D}^+_{FL2}[E'; g] = \{\langle\{e\}, \emptyset, l_3\rangle, \langle\{f\}, \emptyset, l_4\rangle\}$, where $e \prec_1 g,\ f \prec_2 g,\ l_1(e) = l_3(e) = a,\ l_2(f) = l_4(f) = b,\ l_1(g) = l_2(g) = c$, and $E; g \not\approx_{\mathcal{D}^+_{FL2}} E'; g$. Let us note that in the process specified by the formula $E'; g$ an action c can never happen unlike $E; g$.*

2.3 Axiomatization

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

The axiom system Θ_{FL2} is *sound* for $\approx_{\mathcal{D}_{FL2}}$, i.e. if $E = E'$ is an axiom of Θ_{FL2}, then $E \approx_{\mathcal{D}_{FL2}} E'$. The proof consists in determining the semantics of E and E' and comparing them. It can be done directly by the definitions of operations over lposets.

In order to prove that Θ_{FL2} is *complete* for $\approx_{\mathcal{D}_{FL2}}$, we introduce a canonical form of $AFLP_2$-formula.

2.4 Canonical form of formulas

Let us introduce the concepts associated with the structure of $AFLP_2$-formulas.

Precedence is a formula $E_1; \ldots; E_n =;_{i=1}^{n} E_i,\ E_i \in \widehat{Ev}\ (1 \le i \le n)$;

Conjunction is a formula $E_1\| \ldots \|E_n = \|_{i=1}^{n}E_i$, where E_i are precedences $(1 \le i \le n)$.

Disjunction is a formula $E = E_1 \vee \ldots \vee E_n = \vee_{i=1}^{n}E_i$, where $E_i\ (1 \le i \le n)$ are conjunctions.

Normal conjunction is a conjunction $E = \|_{i=1}^{n}E_i$, for which the following requirements are valid.

1. Every formula $E_i\ (1 \le i \le n)$ has one of the forms:
   (a) elementary formula $e\ (e \in Ev),\ \bar{e}\ (\bar{e} \in \overline{Ev}),\ \delta_e\ (\delta_e \in \Delta_{Ev})$;
   (b) *elementary precedence* $(e; f)$, where $e, f \in Ev$ and $e \ne f$;

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

Table 1: Axiom system Θ_{FL2}

2. If there is a formula E_i $(1 \leq i \leq n)$ δ_e $(\delta_e \in \Delta_{Ev})$, then there is not another one E_j $(1 \leq j \leq n)$ s.t. $E_j = \bar{f}$ $(\bar{f} \in \overline{Ev})$;
3. For any formulas E_i and E_j $(1 \leq i \neq j \leq n)$ s.t. $Ev(E_i) \cap Ev(E_j) \neq \emptyset$, E_i and E_j have a form of different elementary precedences;
4. For any pair $E_i = (e; f)$ and $E_j = (f; g)$ $(1 \leq i \neq j \leq n)$ there exists a formula $E_k = (e; g)$ $(1 \leq k \leq n)$ describing the transitive closure of the precedence relation for events e, f and g.

Let E and F be normal conjunctions. A formula E is a *strict prefix* of F, notation $E \lhd F$, if the following requirements are satisfied.

1. $cont^+(E) \subset cont^+(F)$;
2. elementary precedence $(e; f)$ is a conjunctive member of F and $f \in cont^+(E)$ iff $(e; f)$ is a conjunctive member of E;

A formula E is a *prefix* of F, notation $E \unlhd F$, if $E \lhd F$ or $E \simeq F$.

Example 3 *In the formula* $(e\|g\|\bar{f}\|\bar{h}\|\bar{k}) \vee (g\|\delta_e\|\delta_f\|\delta_h\|\delta_k) \vee (e\|\delta_f\|\delta_g\|\delta_h\|\delta_k) \vee ((f; h)\|(f; k)\|\bar{e}\|\bar{g})$ *the second and third conjunctions are strict prefixes of the first one.*

A formula E is in *canonical form*, if it is a disjunction $E = \vee_{i=1}^{n} E_i$ and the following conditions are satisfied.

1. E_i $(1 \leq i \leq n)$ is a normal conjunction;
2. for any E_i and E_j $(1 \leq i \neq j \leq n)$ $E_i \not\simeq E_j$;
3. for any E_i and E_j $(1 \leq i \neq j \leq n)$ $\neg(E_i \lhd E_j \vee E_j \lhd E_i)$.

Each disjunctive member of canonical form characterizes one of the possible alternative behaviours of the nondeterministic process specified by the formula and has a special form practically coinciding with lposet corresponding to this behaviour.

Example 4 *The formula* $(e\|g\|\bar{f}\|\bar{h}\|\bar{k}) \vee ((f;h)\|(f;k)\|\bar{e}\|\bar{g})$ *is in canonical form which is the representation of two lposets corresponding to the deterministic (sub)processes of the nondeterministic process specified by the formula.*

A notation $E =_{\Theta_{FL2}} E'$ means that the equation may be proved using the axiom system Θ_{FL2}.
The following theorems present the required completeness result for Θ_{FL2}.

Theorem 1 *Every formula of* $AFLP_2$ *may be proved equal to unique up to isomorphism canonical form using* Θ_{FL2}.

Proof. As Theorem 6.1 in [6]. □

We will denote a set of all canonical forms of formula E by $canon(E)$. Canonical forms from $canon(E)$ coicide up to associativity and commutativity rules for $\vee$ and $\|$.

Theorem 2 *For any formulas* E *and* E' *of* $AFLP_2$ *the following statement is valid:* $E \approx_{\mathcal{D}_{FL2}} E' \Leftrightarrow E =_{\Theta_{FL2}} E'$.

Proof. As Theorem 6.2 in [6]. □

Hence, we can find whether any two formulas E and E' of $AFLP_2$ equivalent w.r.t. denotational semantics. To do this, it is sufficient to reduce them to their canonical forms F and F' and check them by isomorphism.

The author proposed in [18] the term rewriting system RWS_2 and wrote program CANON based on it to automatically transform any AFP_2-formula into canonical form. We can use CANON also in $AFLP_2$ to check automatically formulas of the algebra by equivalence $\approx_{\mathcal{D}_{FL2}}$ using their canonical forms which may be obtained as outputs of CANON.

2.5 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*.

The transition $(s, a, \tilde{s})$ will be denoted by $s \xrightarrow{a} \tilde{s}$. We will consider only *finite* transition systems, i.e. systems having finite sets of states.

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

- A set of states, $\mathbf{AFLP_2} \cup \{\nu\}$, consists of $AFLP_2$-formulas supplemented by special symbol of “empty” formula ν, denoting the process which does nothing and successfully terminates. For any $AFLP_2$-formula E the folowing equations are supposed: $E\|\nu = \nu\|E = E$ and $cont(\nu) = \emptyset$.
- A set of labels consists of conjunctions of $AFLP_2$ over alphabet Ev. Each such conjunction G is a representation of lposet $\rho_G = \langle cont(G), \prec^\star_G, l_G\rangle$, where $e \prec_G f \ \Leftrightarrow \ (e; f)$ is a conjunctive member of G, and $\prec^\star_G$ 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.
- An 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 $\frac{E \xrightarrow{G} \tilde{E}}{E \| F \xrightarrow{G} \tilde{E} \| F}$ $cont(G) \cap cont(F) = \emptyset$

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

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 a *set of 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*, and G is a *maximal formula* of $TS(F)$. It is easy to see that $\tilde{F}$ is either ν or conjunction of symbols from $\overline{Ev}$ or Δ_{Ev} and $cont(G) \subseteq Ev$. A terminal formula of $TS(F)$ contains the information about events which cannot happen in present behaviour of the nondeterministic process specified by F. Let us denote by $\mathbf{TS_{max}}(F)$ a *set of maximal formulas* of $TS(F)$.

An *operational semantics* of $AFLP_2$ is a mapping $\mathcal{O}_{FL2}$ from $\mathbf{AFLP_2}$ into set of lposets which is defined as follows. Let E be a formula of $AFLP_2$ and $F \in canon(E)$. Then $\mathcal{O}_{FL2}[E] = \{\rho_{G\|\tilde{F}} \mid G \in \mathbf{TS_{max}}(F)\ \&\ F \xrightarrow{G} \tilde{F}\}$. For any formula E of $AFLP_2$ $\mathcal{O}_{FL2}[E] = \cup_{i=1}^{n} \rho_i$ is a set of lposets which characterize the labelled process specified by the formula. Therefore, the definition of operational semantics does not depend on concrete canonical form F of formula E.

"Observable" part of the set is defined as follows: $\mathcal{O}^{+}_{FL2}[E] = \cup_{i=1}^{n} \rho_i^{+}$.

Given a normal conjunction E of $AFLP_2$. E^+ denotes a formula which is a result of removing the symbols of $\overline{Ev} \cup \Delta_{Ev}$ from E. Formally, E^+ is defined as follows.

1. $e^+ = e,\ \bar{e}^+ = \delta_e^+ = \nu,$
2. $(e; f)^+ = e; f,$
3. $(E \circ F)^+ = E^+ \circ F^+,\ \circ \in \{\|, \vee\}.$

The following proposition is devoted to the interrelation between maximal formulas of $TS(F)$ and disjunctive members of F.

Proposition 2 *Let $F = \vee_{i=1}^{n} F_i$ be canonical form. Then:*

1. *For any $G \in \mathbf{TS_{max}}(F)$ and terminal formula $\tilde{F}$ with $F \xrightarrow{G} \tilde{F}$ there exists a disjunctive member F_j $(1 \le j \le n)$ of F s.t. $G \| \tilde{F} \simeq F_j$.*
2. *For any disjunctive member F_j of F there exist $G \in \mathbf{TS_{max}}(F)$ and terminal formula $\tilde{F}$ with $F \xrightarrow{G} \tilde{F}$ s.t. $G \| \tilde{F} \simeq F_j$.*

Proof.

1. We have $\vee_{i=1}^{n} F_i \xrightarrow{G} \tilde{F}$. Since $\tilde{F}$ does not contain disjunction operations, rules 4.1 and 4.2 were applied several times to $\vee_{i=1}^{n} F_i$. Consequently, $\exists j$ $(1 \le j \le n)$ $F_j \xrightarrow{G} \tilde{F}$. Since $cont(G) \subseteq Ev$ and $cont(\tilde{F}) \subseteq \overline{Ev} \cup \Delta_{Ev}$, $F_j \simeq F_j^+ \| \tilde{F}$ and $F_j^+ \| \tilde{F} \xrightarrow{G} \tilde{F}$. By rule 3.1 we have $F_j^+ \xrightarrow{G} \nu$. It is easy to prove with induction by structure of formulas that for some normal conjunction E, $E \xrightarrow{G} \nu$ implies $E = G$. In our case we have $F_j^+ = G$. Thus, $F_j \simeq F_j^+ \| \tilde{F} = G \| \tilde{F}$.

2. Obviously, $F_j^+ \xrightarrow{F_j^+} \nu$. For conjunction $\tilde{F}$ of symbols from $cont^-(F_j)$ or $\Delta_{cont}(F_j)$ (since symbols from $\overline{Ev}$ and Δ_{Ev} may not occur in F_j together) we have $F_j \simeq F_j^+ \| \tilde{F}$. By rule 3.1 $F_j^+ \| \tilde{F} \xrightarrow{F_j^+} \tilde{F}$. Consequently, $F_j \xrightarrow{F_j^+} \tilde{F}$. By rules 4.1, 4.2 $\vee_{i=1}^{n} F_i \xrightarrow{F_j^+} \tilde{F}$, since for disjunctive members of canonical form the following is valid: $cont(F_k) \not\subseteq cont(F_l)$ $(1 \le k \ne l \le n)$. Therefore, $F_j^+ = G \in \mathbf{TS_{max}}(F)$. □

The following proposition says about interrelation between observable part of $\mathcal{O}_{FL2}[E]$ and set of maximal formulas of $TS(F)$ for $F \in canon(E)$.

Proposition 3 *Let E be a formula of $AFLP_2$ and $F \in canon(E)$. Then $\mathcal{O}^+_{FL2}[E] = \{\rho_G \mid G \in \mathbf{TS_{max}}(F)\}$.*

Proof. Let $\rho_{G\|\tilde{F}} \in \mathcal{O}_{FL2}[E]$. Since $cont(G) \subseteq Ev$ and $cont(\tilde{F}) \subseteq \overline{Ev} \cup \Delta_{Ev}$, we have $(G\|\tilde{F})^+ = G$. Consequently, $\rho^+_{G\|\tilde{F}} = \rho_{(G\|\tilde{F})^+} = \rho_G$. □

Now we can present the main result concerning the interrelation between denotational and operational semantics of $AFLP_2$.

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

Proof. Let $F = \vee^n_{i=1} F_i \in canon(E)$. By definition of canonical form the following equation takes place: $\mathcal{D}_{FL2}[E] = \cup^n_{i=1} \rho_{F_i}$. Let $\rho_{G\|\tilde{F}} \in \mathcal{O}_{FL2}[E]$. By Proposition 2 there exists a disjunctive member F_j $(1 \leq j \leq n)$ of F s.t. $G\|\tilde{F} \simeq F_j$. Hence, $\rho_{G\|\tilde{F}} = \rho_{F_j}$, and we have $\mathcal{O}_{FL2}[E] \subseteq \mathcal{D}_{FL2}[E]$. The backward inclusion is proved analogously. □

3 Equivalences on weakly labelled A-nets

Algebra AFP_0 is dual to AFP_2 descriptive calculus [5]. Its formulas specify finite A-nets which can form a semantic domain for a subclass of "structured" formulas of AFP_2 (i.e. formulas over Ev with operations $\bigtriangledown, \|, ;$, in our terminology). The labelling on AFP_0-formulas may be introduced and a new algebra $AFLP_0$ may be obtained as a result. Then formulas of $AFLP_0$ will specify finite *weakly labelled A-nets* (i.e. A-nets having labelling function which may be noninjective).

Formally, *A-net* [12, 10] is an acyclic ordinary strictly labelled net $N = \langle P_N, T_N, F_N, l_N, M_N \rangle$ with the following properties.

1. $\forall p \in P_N$ $(^\bullet p \neq \emptyset) \vee (p^\bullet \neq \emptyset)$, i.e. there are no isolated places;
2. $\forall p, q \in P_N$ $(^\bullet p = {}^\bullet q) \& (p^\bullet = q^\bullet) \Rightarrow p = q$, i.e. there are no "superfluous" places;
3. $\forall t \in T_N$ $(^\bullet t \neq \emptyset) \& (t^\bullet \neq \emptyset)$, i.e. all transitions have input and output places;
4. $\forall x \in P_N \cup T_N$ $|\{y \mid y \prec_N x\}| < \infty$, i.e. the set of causes is finite (here $\prec_N = F^\star_N$ is a transitive closure of F_N);
5. $\forall p \in P_N$ $\forall t, u \in T_N$ $t, u \in {}^\bullet p \Rightarrow t$ **al** u, i.e. transitions with common output place are alternative;
6. $M_N = \{p \in P_N \mid {}^\bullet p = \emptyset\}$, i.e. an initial marking is a set of input places of the net.

The *alternative* relation, denoted by **al**, is defined as follows. Let $t, u \in T_N$ for A-net N. t **al** u, if the following requirements are valid.

1. $(t \not\prec_N u) \& (u \not\prec_N t)$;
2. $(^\bullet t \cap {}^\bullet u \neq \emptyset) \vee (\exists p \in {}^\bullet t \; \forall t' \in {}^\bullet p \; t'$ **al** $u) \vee (\exists q \in {}^\bullet u \; \forall u' \in {}^\bullet q \; t$ **al** $u') \vee (t = u)$.

Let us note that in original definition [12] A-nets are considered as nonlabelled, that corresponds to the requirement of strict labelling in present definition (i.e. no two different transitions have the same label). Since we will consider nets having only finite processes, item 4 of A-nets definition may be ignored. Items 5 and 6 of the definition imply a safeness of A-nets.

Let us define a mapping $\Psi_L : \mathbf{AFLP_0} \rightarrow \mathbf{AFLP_2}$ 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. $\Psi_L(E \bigtriangledown_{FL0} F) = E \bigtriangledown_{FL2} F$.

Symbol "$FL0$" marks the operations of $AFLP_0$, and symbol "$FL2$" is used for $AFLP_2$ ones. Denotational semantics of $AFLP_0$ is a mapping $\mathcal{D}_{FL0}$, which associates with every formula E of the algebra a set of maximal C-subnets (O-subnets, in terms of [5]) of finite A-net N, specified by the formula. Let us note that with every causal net $C = \langle P_C, T_C, F_C, l_C \rangle$ we can associate lposet $\rho_C = \langle T_C, \prec_C \cap (T_C \times T_C), l_C \rangle$.

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 $\{\rho_C \mid C \in \mathcal{D}_{FL0}[E]\} = \mathcal{D}^+_{FL2}[F]$.*

Proof. As Theorem 4.3 in [5], taking into account the information about labelling of E and F. □

Hence, with every formula E of $AFLP_0$ which specifies finite weakly labelled A-net N, we can associate the formula F of $AFLP_2$ s.t. the set of lposets of maximal C-subnets of N coincides with the set of lposets of maximal deterministic (sub)processes of the nondeterministic process specified by F. Let us note that the result of the theorem is valid for any (not only maximal) initial C-subnets of N and for any deterministic processes specified by F. In such a case initial deterministic processes will correspond to initial C-subnets.

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

Example 5 *Let us consider $AFLP_2$-formulas E and E' which are associated with nets N and N' in Figures 1 and 2. Let $lab(e) = lab(e_i) = a,\ lab(f) = lab(f_i) = b,\ lab(g) = lab(g_i) = c,\ lab(h) = lab(h_i) = d\ (1 \le i \le 3)$.*

- *In Figure 1(a) $E = e\|f,\ E' = (e_1; f_1) \bigtriangledown (e_2; f_2)$.*
- *In Figure 1(b) $E = (e_1; f) \bigtriangledown e_2,\ E' = e; f$.*
- *In Figure 1(c) $E = (e; f_1)\|(f_1 \bigtriangledown f_2),\ E' = e\|f$.*
- *In Figure 1(d) $E = (e; f)\|e,\ E' = e; f$.*
- *In Figure 1(e) $E = (e; f)\|(g; h),\ E' = (e; (f_1 \bigtriangledown f_2))\|(e; (f_2 \bigtriangledown h_1))\|(g; (f_2 \bigtriangledown h_1))\|(g; (h_1 \bigtriangledown h_2))\|(f_1 \bigtriangledown h_2)$.*
- *In Figure 2(a) $E = ((e_1 \bigtriangledown e_2); f_1)\|(f_1 \bigtriangledown f_2)\|e_1\|e_2\|f_2,\ E' = ((e_1; f_1) \bigtriangledown (e_2; f_3))\|(f_1 \bigtriangledown f_2)\|(e_2 \bigtriangledown f_2)\|e_1\|f_3$.*
- *In Figure 2(b) $E = (e; f; h)\|(e; g_2)\|(g_1 \bigtriangledown g_2)\|f\|g_1,\ E' = (e; (f_1 \bigtriangledown f_2); h)\|(e; g_2)\|(f_2 \bigtriangledown g_1)\|(g_1 \bigtriangledown g_2)\|f_1$.*
- *In Figure 2(c) $E = e,\ E' = e_1 \bigtriangledown e_2$.*
- *In Figure 2(d) $E = (e \bigtriangledown f)\|e\|f,\ E' = (e \bigtriangledown f)\|(e; g)\|(f; g)$.*

In [19, 20, 21] a wide set of equivalences (considered in the literature as well as proposed by the author) was examined on nets. These equivalences may be partitioned as follows. *Trace equivalences*: interleaving (denoted by $\equiv_i$) [9], step ($\equiv_s$) [16], partial word ($\equiv_{pw}$) [19], pomset ($\equiv_{pom}$) [8] and process ($\equiv_{pr}$) [19]. *Bisimulation equivalences*: interleaving ($\underline{\leftrightarrow}_i$) [15], step ($\underline{\leftrightarrow}_s$) [14], partial word ($\underline{\leftrightarrow}_{pw}$) [23], pomset ($\underline{\leftrightarrow}_{pom}$) [2] and process ($\underline{\leftrightarrow}_{pr}$) [1]. *ST-bisimulation equivalences*: interleaving ($\underline{\leftrightarrow}_{iST}$) [8], partial word ($\underline{\leftrightarrow}_{pwST}$) [23], pomset ($\underline{\leftrightarrow}_{pomST}$)[23] and process ($\underline{\leftrightarrow}_{prST}$) [19]. *History preserving bisimulation equivalences*: partial word ($\underline{\leftrightarrow}_{pwh}$) [19], pomset ($\underline{\leftrightarrow}_{pomh}$) [17] and process ($\underline{\leftrightarrow}_{prh}$) [19].

Since then we considered the following equivalence notions. *Conflict respecting equivalences*: prime event structure (PES) ($\equiv_{pes}$) and occurrence ($\equiv_{occ}$) [8]. *Isomorphism* ($\simeq$) is a coincidence of nets up to renaming of places and transitions. The author proved that correlation of all the equivalences is depicted by graph in Figure 4 without $\approx_{\mathcal{D}_{FL2}}$ and $\approx_{\mathcal{D}^+_{FL2}}$. No additional nontrivial arrow may be added in the graph.

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

Theorem 5 *Let N and N' be weakly labelled A-nets and $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}, \simeq\},\ \star, \star\star \in \{i, s, pw, pom, pr, iST, pwST, pomST, prST, pwh, pomh, prh, pes, occ\}$. 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 $\approx_{\mathcal{D}_{FL2}}$ and $\approx_{\mathcal{D}^+_{FL2}}$).*

Proof. $\Leftarrow$ By Theorem 1 in [19, 20].

$\Rightarrow$ 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 can be executed concurrently.
- In Figure 1(e) $N \underline{\leftrightarrow}_{iST} N'$, but $N \not\equiv_{pw} N'$, since the net N corresponds to pomset s.t. even less sequential pomset cannot be executed in N'.
- In Figure 1(c) $N \underline{\leftrightarrow}_{pwh} N'$, but $N \not\equiv_{pom} N'$, since only in N 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 additional output place.

- In Figure 1(b) $N \equiv_{pr} N'$, but $N \not\leftrightarrow_{i} N'$, since only in N action a can happen so that b cannot happen after it.
- In Figure 2(a) $N \underline{\leftrightarrow}_{pr} N'$, but $N \not\leftrightarrow_{iST} N'$, since only in N' action a can begin working so that no b can start unless a finishes.
- In Figure 2(b) $N \underline{\leftrightarrow}_{prST} N'$, but $N \not\leftrightarrow_{pwh} N'$, 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 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). □

The following example concludes this section.

Example 6 *Let us consider the net N' in Figure 1(e). The corresponding $AFLP_2$-formula is $E' = (e;(f_1 \bigtriangledown f_2))\| (e;(f_2 \bigtriangledown h_1))\|(g;(f_2 \bigtriangledown h_1))\|(g;(h_1 \bigtriangledown h_2))\|(f_1 \bigtriangledown h_2)$, $lab(e) = a$, $lab(f_1) = lab(f_2) = b$, $lab(g) = c$, $lab(h_1) = lab(h_2) = d$. 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 which are presented in Figure 3. In this figure labels of events are in parentheses, and partial order is depicted by arrows.*

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

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

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 $TS(F')$ 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 information that in present behaviour of the labelled nondeterministic process, specified by E', events f_2 and h_2 did not happen since some alternative with them events (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 $e \prec_1 f_1$, $e \prec_1 h_1$, $g \prec_1 h_1$, $e \prec_2 f_2$, $g \prec_2 f_2$, $g \prec_2 h_2$, $l(e) = l_1(e) = l_2(e) = a$, $l(f_1) = l(f_2) = l_1(f_1) = l_2(f_2) = b$, $l(g) = l_1(g) = l_2(g) = c$, $l(h_1) = l(h_2) = l_1(h_1) = l_2(h_2) = d$.

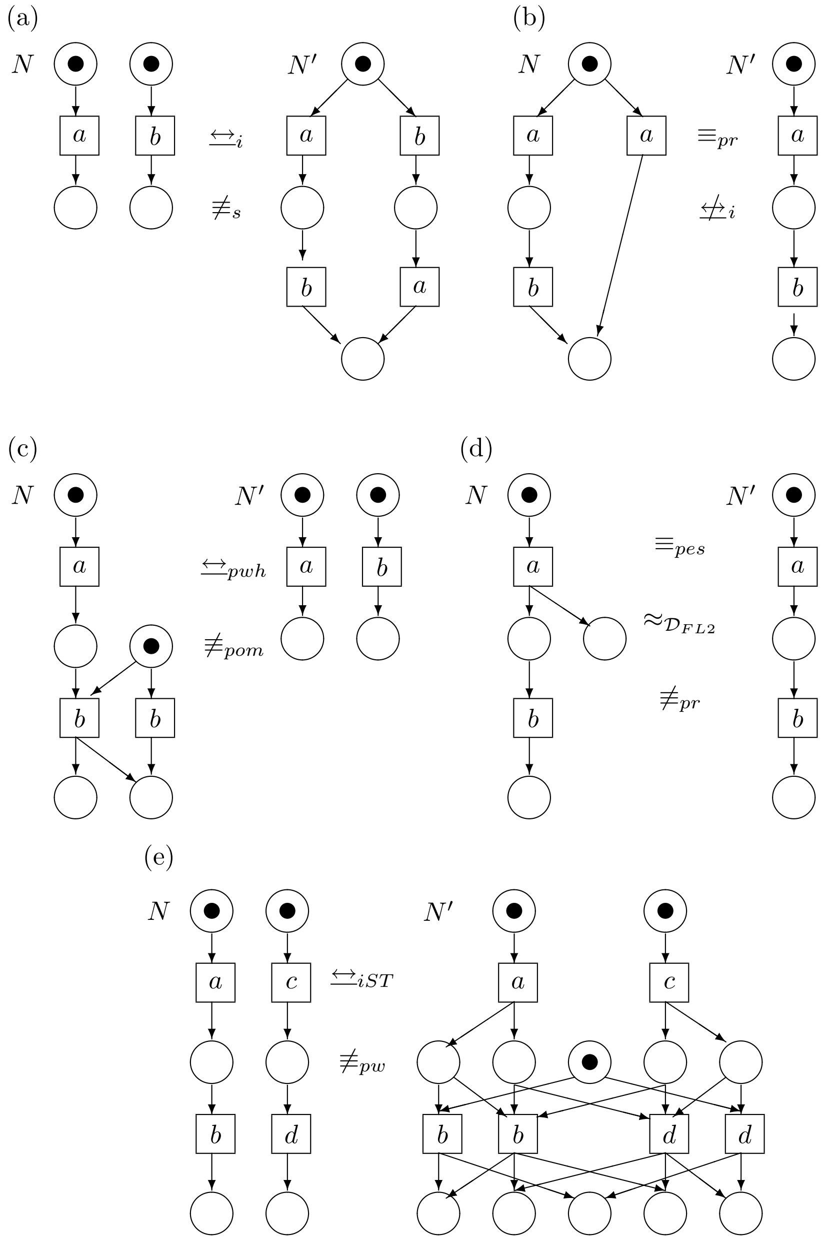

Figure 1: Examples of weakly labelled A-nets

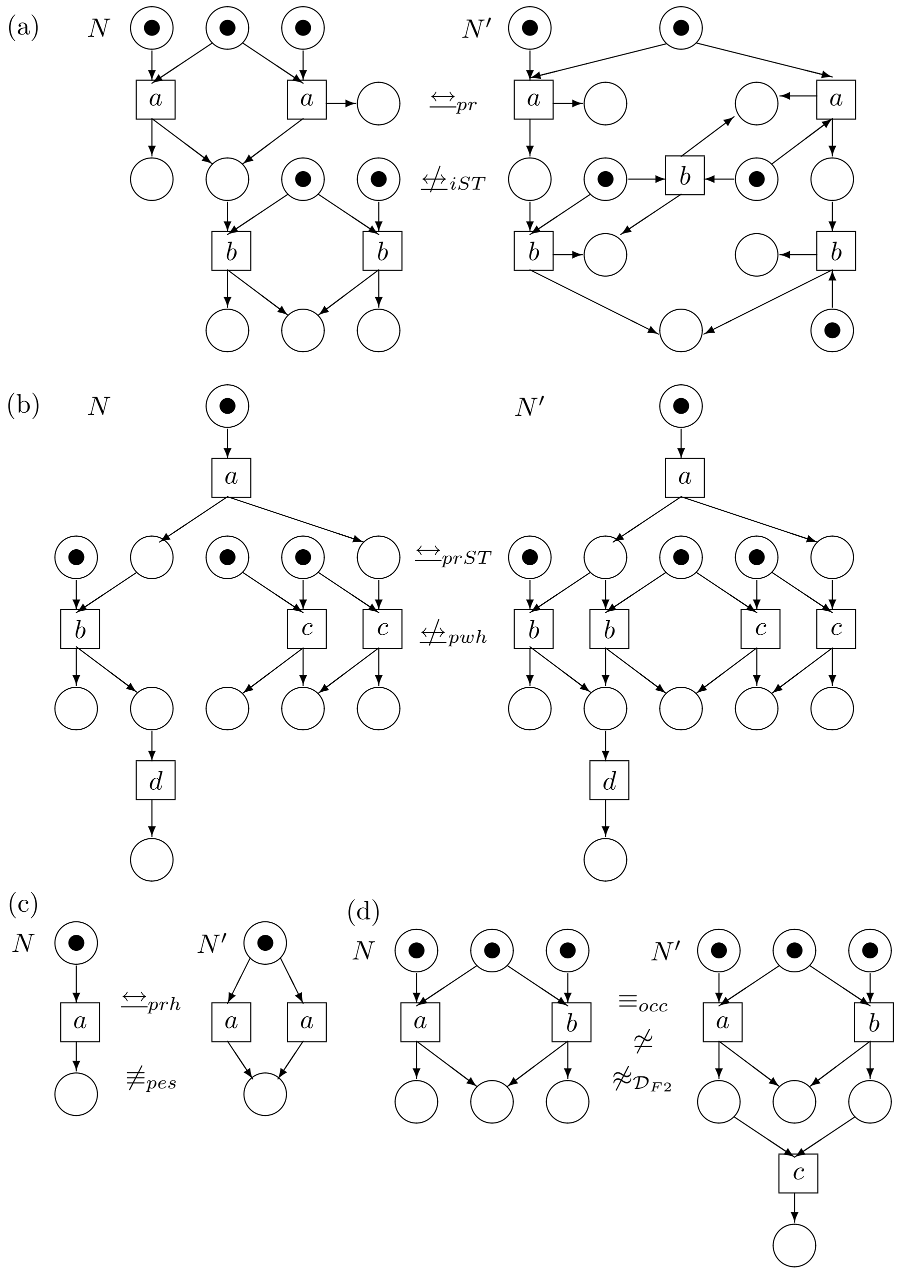

Figure 2: Examples of weakly labelled A-nets (continued)

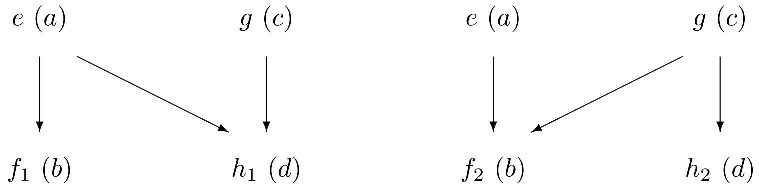

Figure 3: Set of lposets of the labelled nondeterministic process

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

Any finite A-net, as it was proved in [11], can be represented by AFP_0-formula using regularization algorithm. Therefore, any finite weakly labelled A-net can be represented by $AFLP_0$-formula with the use of the analogous algorithm. In the previous section the mapping Ψ_L was defined which associates $AFLP_2$-formula with every $AFLP_0$-formula and preserves the sets of lposets. Hence, one can associate $AFLP_2$-formula E with every finite weakly labelled A-net N s.t. the set of lposets of initial C-subnets of N coincides with the set of lposets of deterministic processes specified by E.

In such a case it is clear that the concepts of formula equivalences of $AFLP_2$ may be extended to nets. Given some formula equivalence, we will consider two nets to be equivalent iff the formulas are equivalent which are associated with these nets.

Let us consider the interrelation of the net and formula equivalences.

Theorem 6 *Let N and N' be weakly labelled A-nets and $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}, \simeq, \approx\}$, $\star, \star\star \in \{i, s, pw, pom, pr, iST, pwST, pomST, prST, pwh, pomh, prh, pes, occ, \mathcal{D}_{FL2}, \mathcal{D}^{+}_{FL2}\}$. 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.*

Proof. $\Leftarrow$ Using Theorem 5 and the following notes.

- $\approx^{+}_{\mathcal{D}_{FL2}}$ implies $\equiv_{pes}$. It is proved as follows. Let $N \approx_{\mathcal{D}^{+}_{FL2}} N'$, E and E' are the formulas which corresponds to the nets N and N' respectively. We have $\mathcal{D}^{+}_{FL2}[E] = \mathcal{D}^{+}_{FL2}[E'] = \cup_{i=1}^{n}\rho_i$, $\rho_i = \langle X_i, \prec_i, l_i\rangle$ $(1 \le i \le n)$. On the basis of this set of lposets we can uniquely construct LES $\xi = \langle \cup_{i=1}^{n} X_i, \cup_{i=1}^{n} \prec_i, \#, \cup_{i=1}^{n} l_i\rangle$, where $x\#y \;\Leftrightarrow\; \forall i\ (1 \le i \le n)\ (x \notin X_i) \vee (y \notin X_i)$. It is easy to see that $\mathcal{E}(N) = \mathcal{E}(N')$ is an isomorphism class of ξ. Consequently, $N \equiv_{pes} N'$.
- $\approx_{\mathcal{D}^{+}_{FL2}}$ is a consequence of $\approx_{\mathcal{D}_{FL2}}$, since $\approx_{\mathcal{D}^{+}_{FL2}}$ does not respect the symbols of $\overline{Ev} \cup \Delta_{Ev}$.

$\Rightarrow$ Using Theorem 5 and the following examples of weakly labelled A-nets.

- A-nets N and N' in Figure 1(d) are associated with $AFLP_2$-formulas $E = (e; f)\|e$ and $E' = e; f$, $lab(e) = a$, $lab(f) = b$. Since $\mathcal{D}_{FL2}[E] = \mathcal{D}_{FL2}[E'] = \langle \{e, f\}, \prec, l\rangle$, where $e \prec f$, $l(e) = a$, $l(f) = b$, we have $N \approx_{\mathcal{D}_{FL2}} N'$, but $N \not\equiv_{pr} N'$.
- Let us consider some weakly labelled A-nets N and N' which differ only by transition names. We have $N \simeq N'$, but $N \not\approx_{\mathcal{D}^{+}_{FL2}} N'$, since $\approx_{\mathcal{D}^{+}_{FL2}}$ respects transition names (events).
- A-nets N and N' in Figure 2(d) are associated with $AFLP_2$-formulas $E = (e \bigtriangledown f)\|e\|f$ and $E' = (e \bigtriangledown f)\|(e; g)\|(f; g)$, $lab(e) = a$, $lab(f) = b$. $N \approx_{\mathcal{D}^{+}_{FL2}} N'$, but $N \not\approx_{\mathcal{D}_{FL2}} N'$, since $\mathcal{D}_{FL2}[E] = \{\langle \{e, \delta_f\}, \emptyset, l\rangle, \langle \{f, \delta_e\}, \emptyset, l\rangle\}$, $l(e) = a$, $l(f) = b$, whereas $\mathcal{D}_{FL2}[E'] = \{\langle \{e, \delta_f, \delta_g\}, \emptyset, l'\rangle, \langle \{f, \delta_e, \delta_g\}, \emptyset, l'\rangle\}$, $l'(e) = a$, $l'(f) = b$, $l'(g) = c$. □

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

In this section we introduce equivalences on formulas of $AFLP_2$ which correspond to the net ones.

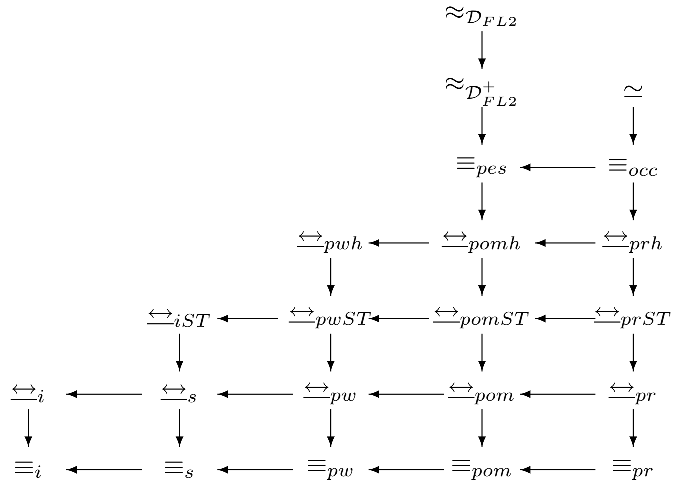

Figure 4: Correlation of the net equivalences and equivalences of $AFLP_2$

5.1 Process subformulas

Let E be $AFLP_2$-formula and $F \in canon(E)$. A *set of process subformulas* of E is defined as follows: $PSF(E) = \{G \mid G \in canon(H)\ \&\ H \in \mathbf{TS}(F)\} \cup \{\nu\}$. One can see that this definition does not depend on concrete canonical form F of formula E, since PSF(E) contains *all* possible transpositions of conjunctive members of normal conjunctions based on each formula $H \in \mathbf{TS}(F)$. By definition, a process subformula is either ν or normal conjunction which is a disjunctive member of F or prefix of such a member. We consider process subformulas up to isomorphism. Since process subformulas are normal conjunctions, isomorphism on such formulas is a coincidence up to transposition of conjunctive members. Let lposet $\rho_\nu = \langle \emptyset, \emptyset, \emptyset \rangle$ correspond to empty formula ν.

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

Let $G \in PSF(E)$. Then G is a *maximal process subformula* of E, if it can be extended by no process subformula. A *set of all maximal process subformulas* of E is denoted by $PSF_{max}(E)$.

Example 7 *For the formula E', corresponding to the net N' in Figure 1(e), $PSF_{max}(E') = \{(e; f_1)\|(e; h_1)\|(g; h_1), (e; f_2)\|(g; f_2)\|(g; h_2)\}$. Let us note that each of 2 process subformulas in $PSF_{max}(E')$ represents an isomorphism class consisting of 6 formulas which are different transpositions of conjunctive members. Since we consider process subformulas up to isomorphism, we write only 2 formulas instead of 12.*

5.2 Trace equivalences

An *interleaving trace* of a formula E is a sequence $a_1 \cdots a_n \in Act^*$ s.t. $\nu \xrightarrow{a_1} G_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} G_n$, where $G_i \in PSF(E)$ $(1 \le i \le n)$. Let us denote a *set of all interleaving traces* of E by $SeqTraces(E)$. Two formulas E and E' are *interleaving trace equivalent*, notation $E \equiv_i E'$, 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. $\nu \xrightarrow{A_1} G_1 \xrightarrow{A_2} \dots \xrightarrow{A_n} G_n$, where $G_i \in PSF(E)$ $(1 \le i \le n)$. Let us denote a *set of all step traces* of E by $StepTraces(E)$. Two formulas E and E' are *step trace equivalent*, notation $E \equiv_s E'$, 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'. Let us denote by $Pomsets(E)$ a *set of all pomset traces* of E. Two formulas E and E' are *partial word trace equivalent*, notation $E \equiv_{pw} E'$, 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*, notation $E \equiv_{pom} E'$, iff $Pomsets(E) = Pomsets(E')$.

5.3 Bisimulation equivalences

A notation $\mathcal{R} : E\underline{\leftrightarrow}_\star E'$ means that $\mathcal{R}$ is a bisimulation of type $\star$ ($\star$-bisimulation) between formulas E and E'. E and E' are $\star$-*bisimulation equivalent*, notation $E\underline{\leftrightarrow}_\star E'$, iff $\mathcal{R} : E\underline{\leftrightarrow}_\star E'$ for some $\star$-bisimulation $\mathcal{R}$.

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*\}, notation $\mathcal{R} : E\underline{\leftrightarrow}_\star E'$, $\star \in \{i, s, pw, pom\}$, iff:

1. $(\nu, \nu) \in \mathcal{R}$;
2. $(G, G') \in \mathcal{R},\ G \stackrel{\hat{G}}{\longrightarrow} \tilde{G}$,
   (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. As previous item but the roles of E and E' are reversed.

5.3.2 ST-process subformulas

An *ST-process subformula* of a formula E is a pair (G, H) s.t. $G, H \in PSF(E),\ H \stackrel{K}{\longrightarrow} G$ 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 has started. The process subformula H corresponds to that part of G, which has finished, and K — 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.

(ν, ν) is 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*\}, notation $\mathcal{R} : E\underline{\leftrightarrow}_{\star ST} E'$, $\star \in \{i, pw, pom\}$, 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. $((G, H), (G', H'), \beta) \in \mathcal{R},\ (G, H) \to (\tilde{G}, \tilde{H})\ \Rightarrow\ \exists \tilde{\beta},\ (\tilde{G}', \tilde{H}') :\ (G', H') \to (\tilde{G}', \tilde{H}'),\ \tilde{\beta}\lceil_{cont(G)} = \beta$, $((\tilde{G}, \tilde{H}), (\tilde{G}', \tilde{H}'), \tilde{\beta}) \in \mathcal{R}$, and if $H \stackrel{K}{\longrightarrow} \tilde{G},\ H' \stackrel{K'}{\longrightarrow} \tilde{G}'$ then:
   (a) $(\tilde{\beta}\lceil_{cont(K)})^{-1} : \rho_{K'} \sqsubseteq \rho_K$, if $\star = pw$;
   (b) $\tilde{\beta}\lceil_{cont(K)}: \rho_K \simeq \rho_{K'}$, if $\star = pom$;
4. As previous item but the roles of E and E' are reversed.

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 $\star$-*history preserving bisimulation* between E and E', $\star \in$\{ *partial word, pomset*\}, notation $\mathcal{R} : E\underline{\leftrightarrow}_{\star h} E'$, $\star \in \{pw, pom\}$, iff:

1. $(\nu, \nu, \emptyset) \in \mathcal{R}$;
2. $(G, G', \beta) \in \mathcal{R}\ \Rightarrow\ \beta : \rho_G \approx \rho_{G'}$;
3. $(G, G', \beta) \in \mathcal{R},\ G \to \tilde{G}\ \Rightarrow\ \exists \tilde{\beta},\ \tilde{G}' : G' \to \tilde{G}',\ \tilde{\beta}\lceil_{cont(G)} = \beta,\ (\tilde{G}, \tilde{G}', \tilde{\beta}) \in \mathcal{R}$ and

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

(b) $\tilde{\beta} : \rho_{\tilde{G}} \simeq \rho_{\tilde{G}'}$, if $\star = pom$;

4. As previous item but the roles of E and E' are reversed.

5.4 Conflict respecting 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 LES $\xi_F = \langle cont^+(F), \prec_F, \#_F, l_F\lceil_{cont^+(F)}\rangle$, 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.

Let us denote by $\mathcal{E}(E)$ a PES which is an isomorphism class of ξ_F for $F \in canon(E)$. Obviously, the definition of $\mathcal{E}(E)$ does not depend on concrete canonical form F of formula E. Formulas E and E' are *prime event structure (PES-) equivalent*, notation $E \equiv_{pes} E'$, if $\mathcal{E}(E) = \mathcal{E}(E')$.

5.5 Interrelation of the net equivalences with their analogs in $AFLP_2$

Let E be a formula of $AFLP_2$ which corresponds to finite weakly labelled A-net N. In Section 3 the set of lposets of initial C-subnets of N was established to coincide with set of lposets of deterministic processes specified by E. The following proposition says about the interrelation of lposets of processes of N (from set of all processes $\Pi(N)$ of N) and lposets of process subformulas of E.

Proposition 4 *Let E be a formula of $AFLP_2$ corresponding to finite weakly labelled A-net N. Then $\{\rho_C \mid \pi = (C, id) \in \Pi(N)\} = \{\rho_G \mid G \in PSF(E)\}$.*

Proof.

1. As it was mentioned in [5], a set of maximal C-subnets of finite A-net forms a set of its maximal processes. Obviously, a set of initial C-subnets forms a set of all (not only maximal) processes of A-net. The similar fact is valid for weakly labelled A-nets. Hence, we may consider a set of all processes of N, $\Pi(N)$ as consisting (up to isomorphism of processes) of processes having the form $\pi = (C, id)$, where id is an identity mapping over $P_C \cup T_C$. A lposet $\rho_C = \langle T_C, \prec_C \cap (T_C \times T_C), l_C\rangle$ may be associated with each such a process.

2. On the other side, with each disjunctive member F_j $(1 \le j \le n)$ of $F = \vee_{i=1}^{n} F_i \in canon(E)$ lposet of one of the maximal deterministic processes specified by E, $\rho^+_{F_j} = \langle cont^+(F_j), \prec^\star_{F_j}, l_{F_j}\rangle$, may be associated. Hence, with disjunctive members of F and their prefixes lposets of all (not only maximal) deterministic processes specified by E may be associated. Let us note that for any disjunctive member F_j (of its prefix) of F there exists a process subformula $G \in PSF(E)$ s.t. $G \simeq F_j^+$ and $\rho^+_{F_j} = \rho_{F_j^+} = \rho_G$. □

The following proposition establishes a bijection between the set of processes of N and the set of process subformulas of E which preserves lposets.

Proposition 5 *Let E be a formula of $AFLP_2$ corresponding to finite weakly labelled A-net N. Then there exists a bijection $\chi : \Pi(N) \to PSF(E)$ s.t. for $\pi \in \Pi(N)$, $\pi = (C, id)$ and $G \in PSF(E)$ with $\chi(\pi) = G$ we have $\rho_C = \rho_G$.*

Proof. Let us demonstrate that lposets define up to isomorphism both processes of N and process subformulas of E.

1. We define a mapping χ_1 from $\Pi(N)$ into set of lposets as follows. If $\pi = (C, id) \in \Pi(N)$ then $\chi_1(\pi) = \rho_C$. Obviously, each process is associated with the only lposet. Consequently, χ_1 is a function. It is a surjection by definition. In addition, each process $\pi = (C, id) \in \Pi(N)$ is determined uniquely by its causal net C. A net C is an initial C-subnet of N, and, consequently, it is uniquely determined by its transition set T_C. Therefore, no two different processes of $\Pi(N)$ are associated with the same lposet, because otherwise transition sets of causal nets of the processes would coincide. Hence, χ_1 is a bijection.

2. We define a mapping χ_2 from $PSF(E)$ into set of lposets as follows. If $G \in PSF(E)$ then $\chi_2(G) = \rho_G$. Obviously, each process subformula is associated with the only lposet. Consequently, χ_2 is a function. It is a surjection by definition. In addition, no two different (not isomorphic) process subformulas are associated with one lposet, since process subformulas are, essentially, representations of lposets. Hence, χ_2 is a bijection.

If $\chi = \chi_2^{-1} \circ \chi_1$ then $\chi : \Pi(N) \to PSF(E)$ is a bijection which preserves lposets, i.e. if $\chi(\pi) = G$, $\pi = (C, id)$ then $\rho_C = \rho_G$. □

Now we will prove the result concerning extension rules for processes and process subformulas.

Proposition 6 *Let E be a formula of $AFLP_2$ corresponding to finite weakly labelled A-net N. Then $\forall \pi, \pi' \in \Pi(N)$ $\pi \xrightarrow{\hat{\pi}} \tilde{\pi} \;\Leftrightarrow\; \chi(\pi) \xrightarrow{\chi(\hat{\pi})} \chi(\tilde{\pi})$.*

Proof. It is sufficient to remark that the definitions of process and process subformula extensions are based on the following extension rule for lposets. Let $\rho = \langle X, \prec, l\rangle$, $\tilde{\rho} = \langle \tilde{X}, \tilde{\prec}, \tilde{l}\rangle$, $\hat{\rho}$ are lposets. $\tilde{\rho}$ is an *extension* of ρ *by* $\hat{\rho}$, notation $\rho \xrightarrow{\hat{\rho}} \tilde{\rho}$ iff $\rho \lhd \tilde{\rho}$ and $\hat{\rho} = \tilde{\rho}\lceil_{\tilde{X}\setminus X}$. □

Now we can present the main result of this section concerning interrelation of the net equivalences and their analogs on formulas.

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

Proof. $\Rightarrow$ Any trace of the net N [19, 20] is a trace of E. To prove it is sufficient to replace each $\pi \in \Pi(N)$ in definition of trace of N by process subformula $G \in PSF(E)$ s.t. $\chi(\pi) = G$. The fact that any trace of E is a trace of N is proved analogously. Therefore, the sets of traces of N and E coincide as well as sets of traces of N' and E'. Consequently, $N \equiv_\star N' \Leftrightarrow E \equiv_\star E'$, $\star \in \{i, s, pw, pom\}$.

Using Proposition 6, we may assert the following. Let $\star \in \{i, s, pw, pom, iST, pwST, pomST, pwh, pomh\}$, then $\mathcal{R} : N \underline{\leftrightarrow}_\star N' \;\Leftrightarrow\; \mathcal{S} : E \underline{\leftrightarrow}_\star E'$, where $\mathcal{S}$ is defined as follows.

Usual bisimulations $(\pi, \pi') \in \mathcal{R} \;\Leftrightarrow\; (\chi(\pi), \chi'(\pi')) \in \mathcal{S}$;

ST-bisimulations $((\pi_E, \pi_P), (\pi'_E, \pi'_P), \beta) \in \mathcal{R} \;\Leftrightarrow\; ((\chi(\pi_E), \chi(\pi_P)), (\chi'(\pi'_E), \chi'(\pi'_P)), \chi' \circ \beta \circ \chi^{-1}) \in \mathcal{S}$.

History preserving bisimulations $(\pi, \pi', \beta) \in \mathcal{R} \;\Leftrightarrow\; (\chi(\pi), \chi'(\pi'), \chi' \circ \beta \circ \chi^{-1}) \in \mathcal{S}$.

Formula E specifies nondeterministic process which is a maximal O-process (process based on occurrence instead of causal net, branching process in terms of [7]) of N. Consequently, PES based on occurrence net of such a process of N, notation $\mathcal{E}(N)$, coincides with $\mathcal{E}(E)$. We also have $\mathcal{E}(N') = \mathcal{E}(E')$. Consequently, $N \equiv_{pes} N' \Leftrightarrow E \equiv_{pes} E'$.

$\Leftarrow$ As previous item but using χ^{-1} and $(\chi')^{-1}$ instead of χ and χ' respectively. □

Clearly, correlation of formula equivalences and analogs of the net equivalences in $AFLP_2$ is depicted by graph in Figure 4, where process equivalences are removed (since they are unexpressible in terms of process algebras).

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

Example 8 *Let $E = e \bigtriangledown f$ and $E' = (e \bigtriangledown f)\|e\|f$, where $lab(e) = a$, $lab(f) = b$, $lab(g) = c$. We have $E \approx_{\mathcal{D}^+_{FL2}} E'$, 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 $SeqTraces(E'; g) = \{a, b\}$.*

Let us 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 considering N and N', for which $N \not\equiv_i N'$, since only in N' an action c can never happen.

Consequently, none of the considered equivalences on $AFLP_2$-formulas is a congruence, with the exception of $\approx_{\mathcal{D}_{FL2}}$, i.e. $\approx_{\mathcal{D}_{FL2}}$ is the weakest equivalence which is a congruence.

6 Conclusion

In the paper a new algebra $AFLP_2$ was presented for description and analysis of properties of labelled nondeterministic processes. Denotational and operational semantics and formula equivalences on their basis were proposed. A correlation of the net and formula equivalences was established on finite weakly labelled A-nets. Analogs of the net equivalences were introduced on $AFLP_2$-formulas which are in accordance with the initial equivalences on Petri nets. Hence, algebra $AFLP_2$ possesses rather powerful tools to deal with nondeterministic finite processes.

Further development of the theme may consist in introducing a recursion operator in $AFLP_2$ (as it was suggested in [5] for AFP_2) to specify not only finite but infinite processes as well. Now, the author develops algebra ALP_2 which is an extension of $AFLP_2$ by recursion.

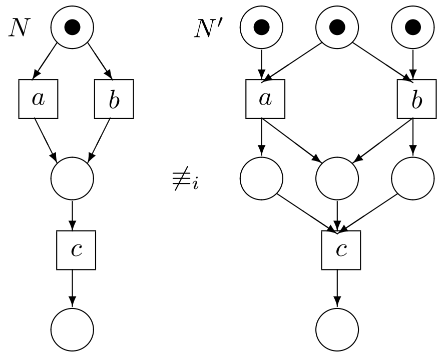

Figure 5: A-nets from example of congruence

Acknowlegements I would like to thank my scientific supervisor Irina B. Virbitskaite for advice to introduce the net equivalences on formulas of AFP_2, which iniciated the author to propose new algebra $AFLP_2$, and for many useful discussions which played an important role in the research.

This work was done while the author's visit to Institute of Informatics of Hildesheim. I thank the head of the institute Prof. Dr. Eike Best for providing a good working atmosphere and for helpful advices, remarks and questions.

References

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

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

[3] Boudol G., Castellani I. *Concurrency and atomicity.* TCS 59, p.25-84, 1988.

[4] Brookes S.D., Hoare C.A.R., Roscoe A.D. *A theory of communicating sequential processes.* Journal of the ACM 31(3), p.560–599, 1984.

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

[6] Cherkasova L.A. *Algebra AFP_2 for concurrent nondeterministic processes: Fully abstract model and complete axiomatization.* Reihe Informatic 72, 28p., July 1990.

[7] Engelfriet J. *Branching processes of Petri nets.* Acta Informatica 28(6), p.575–591, 1991.

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

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

[10] Kotov V.E., Cherkasova L.A. *On structural properties of generalized processes.* LNCS 188, p.288–306, 1985.

[11] Kotov V.E. *An algebra for parallelism based on Petri nets.* LNCS 64, p.39–55, 1978.

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

[13] Milner R.A.J. *A calculus of communicating systems.* LNCS 92, p.172–180, 1980.

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

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

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

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

[18] Tarasyuk I.V. *Rewriting terms of algebra AFP_2 into canonical form.* Problems of Theoretical and Experimental Programming, p.47–59, Institute of Informatics Systems, Novosibirsk, 1993 (in Russian).

[19] Tarasyuk I.V. *Equivalences on Petri nets.* Specification, Verification and Net Models of Concurrent Systems, p.33–54, Institute of Informatics Systems, Novosibirsk, 1994.

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

[21] Tarasyuk I.V. *An investigation of net equivalences.* Proceedings of 4^{th} International Conference on Applied Logics, p.74–75, Irkutsk, June 1995 (In Russian).

[22] Vogler W. *Failures semantics based on interval semiwords is a congruence for refinement.* LNCS 415, p.285–297, 1990.

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