Joint NCC & IIS Bull., Comp. Science, 13 (2000), 76–87

Algebraic characterization of behavioural equivalences over event structures*

A. Votintseva

We consider the process algebra BPA^* proposed by Bergstra, Bethke, and Ponse, since it nicely defines a class of infinite processes. Investigation of representation of event structures for this class of processes is presented in this article. We extend the algebra BPA^* by a parallel composition and modify its sequential operation. For the obtained algebra, named $PBPA^*$, we get a correspondence between an algebraic bisimulation defined using the transition systems over $PBPA^*$-processes and a behavioural one defined over event structures. This gives us better understanding of the place of event structures among other models of parallelism.

1. Introduction

The development of methods for the design of concurrent/distributed systems and investigation of their properties are carried out by means of different formal models (Petri nets, trace languages, transition systems, event structures, process algebras, etc.) varying accordingly to the class of systems, the level of abstraction for structures and behaviours, and the kind of problems under consideration. When verifying different properties of processes and establishing a transition from one abstract model to another, one can demand the subclasses of systems with equivalent behaviours to be specified. At the time, there have been designed a lot of equivalence notions for different models of concurrent and distributed systems. With the aim to classify the variety of their semantic representations, it is necessary to choose a common model of processes and establish its correspondence to other ones (e.g., in [7]).

Event structures are a well-known formalism of "true concurrency" which provides a very detailed model for concurrent and distributed systems. All the main issues attendant the concurrent computations are presented therein. The notion of event structures was proposed by Nielsen, Plotkin and Winskel in [10] to establish the correspondence between occurrence nets (a class of Petri nets) and Scott domains (a class of partial orders). An event structure is a partially ordered set of event occurrences together with a symmetric conflict relation. The ordering relation models causality, whereas the conflict relation expresses alternative choices between events. Two event occurrences that are neither causally comparable nor in conflict may occur concurrently. In this sense, event structures provide explicit and distinct representations of causality, choice, and concurrency. Computations in an event structure are modelled by conflict-free and left-closed sets of event occurrences.

The notion of a bisimulation equivalence was introduced in [14]. The importance of bisimulations in the concurrent systems theory is widely acknowledged. A bisimilarity of two systems means that they can model the behaviours of each other in the branching-time semantics, i.e., starting with equivalent states, the bisimilar systems must be able to perform the same moves, which leads to the next pair of equivalent states. Initially, the bisimulation notion was introduced over transition systems, and later it was extended to other formal models such as event structures, Petri nets, process algebras, and others. For finite state automata, it was shown that a bisimulation equivalence is decidable with the time complexity $O(m \log n)$, where m is the number of transitions and n is the number of states. In [5] the variants of bisimulations over event structures were investigated, namely, interleaving, step, pomset, and history-preserving ones. In forth-and-back variants of bisimulations ([9]), two systems model the behaviour of each other not only in the future but also in the past. The forth-and-back bisimulations are interesting because of their correspondence to equivalences induced by temporal and

*This work is supported by RFBR (grant No 00-01-00898), DAAD Post-Doc grant and the youth grant 2000 of SB RAS.

modal logics with past operators. In [11] a number of bisimulations explicitly reflecting conflict and concurrency have been proposed, and all their interrelations have been established.

Correspondences between bisimulation notions defined over different domains attract a lot of scientific interests. A possible approach to study of behavioural equivalences is to characterize them by means of process algebras. Several results in decidability and (full or partial) axiomatization of equivalences have been obtained for algebraic systems. As an example, decidability of weak bisimilarities between BPA (Basic Process Algebra), BPP (Basic Parallel Processes) and finite-state processes has been investigated in [6], and axiomatization and its completeness have been established for a bisimulation over BPA^* (BPA enriched with an iteration) in [2, 15]. In this paper, we extend the algebra BPA^* to its parallel variant $PBPA^*$ by adding a new operator and adapting another one, which allows us to relate it to prime event structures. We establish the correspondence between an interleaving bisimulation earlier introduced over event structures and "algebraic" one, i.e. that defined over the transition systems of $PBPA^*$-terms specified by transition rules.

The paper is organized as follows. In Section 2, we remind the basic definitions from the event structures theory. In Section 3, an extension $PBPA^*$ of the process algebra BPA^* and its operational semantics are presented. Section 4 proposes the event structure semantics for $PBPA^*$-terms and establishes the correspondence between the interleaving bisimulation over event structures and algebraic one. Conclusion resumes the main achievements and gives some prospects for further research.

2. Basic notions of event structures

A prime event structure (event structure for brevity) consists of a set of event occurrences partially ordered by a causality relation. In addition, the structure contains a conflict relation between the events. Two events that are neither causally related nor in conflict are called concurrent.

Definition 2.1. A *(labeled) event structure* over an alphabet Act is a quadruple $\mathcal{E} = (E, \leq, \#, l)$, where

- E is a countable set of events;
- $\leq \subseteq E \times E$ is a partial order (the *causality* relation) satisfying the *principle of finite causes*: $\forall e \in E \diamond \{d \in E \mid d \leq e\}$ is finite;
- $\# \subseteq E \times E$ is a symmetric and irreflexive relation (the *conflict* relation) satisfying the *principle of conflict heredity*: $\forall e_1, e_2, e_3 \in E \diamond e_1 \leq e_2 \ \&\ e_1 \# e_3 \Rightarrow e_2 \# e_3$;
- $l : E \to Act$ is a labeling function. □

The components of an event structure $\mathcal{E}$ are denoted as $E_{\mathcal{E}}$, $\#_{\mathcal{E}}$, $\leq_{\mathcal{E}}$ and $l_{\mathcal{E}}$. The index $\mathcal{E}$ can be omitted if clear from the context. For $\mathcal{E} = (E, \leq, \#, l)$, we denote: $id = \{(e, e) \mid e \in E\}$; $< = \leq \setminus id$; $\leq^2 \subseteq \leq$ (transitivity); $\lessdot = < \setminus <^2$ (immediate causal dependency); $\smile = (E \times E) \setminus (\leq \cup \geq \cup \#)$ (concurrency); $e \#_m d \iff e \# d \ \&\ \forall e_1, d_1 \in E \diamond (e_1 \leq e \ \&\ d_1 \leq d \ \&\ e_1 \# d_1) \Rightarrow (e_1 = e \ \&\ d_1 = d)$ (minimal conflict).

In the grafic representations of an event structure, only minimal conflicts (not the inherited ones) are pictured. The immediate causal dependencies are represented by directed arcs, omitting those derivable by transitivity. A trivial example of an event structure is shown in Fig. 2.1, where $e_1 \smile e_2$ and $e_2 \smile e_3$.

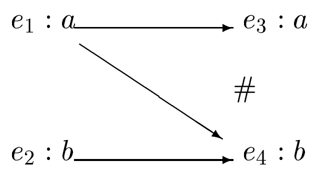


Figure 2.1.

An event structure $\mathcal{E}$ is called *empty* if $E_\mathcal{E} = \emptyset$; *finite* if $E_\mathcal{E}$ is finite; *conflict-free* if $\#_\mathcal{E} = \emptyset$; a *substructure* of an event structure $\mathcal{F}$ ($\mathcal{E} \sqsubseteq \mathcal{F}$) if $E_\mathcal{E} \subseteq E_\mathcal{F}$, $\leq_\mathcal{E} \subseteq \leq_\mathcal{F}$, $\#_\mathcal{E} \subseteq \#_\mathcal{F}$ and $l_\mathcal{E} = l_\mathcal{F} \mid_{E_\mathcal{E}}$. Two event structures $\mathcal{E}$ and $\mathcal{F}$ are called *isomorphic* ($\mathcal{E} \cong \mathcal{F}$) if there is a bijection between the sets $E_\mathcal{E}$ and $E_\mathcal{F}$ preserving the relations $\leq$, $\#$ and labeling.

The states of an event structure are called configurations. A configuration defines the set of events occurred at a point of time. An event can occur in a configuration if all preceded events have already occurred in it. Two events related by a conflict can not occur in the same configuration.

Definition 2.2. A *configuration* of an event structure $\mathcal{E}$ is a subset $C \subseteq E_\mathcal{E}$ such that
(i) $\forall e, e' \in C \diamond \neg(e \#_\mathcal{E} e')$ (conflict-freeness);
(ii) $\forall e, e' \in E_\mathcal{E} \diamond e \in C \ \&\ e' \leq_\mathcal{E} e \Rightarrow e' \in C$ (left-closedness).

By $\mathcal{C}(\mathcal{E})$ we denote the set of all configurations in $\mathcal{E}$. □

A configuration $C \in \mathcal{C}(\mathcal{E})$ is called *maximal* if the following holds: $C' \in \mathcal{C}(\mathcal{E})\ \&\ C \subseteq C' \Rightarrow C = C'$, i.e. C is maximal w.r.t. the set inclusion.

The set of configurations for the event structure shown on Fig. 2.1 includes the following elements: $\emptyset$, $\{e_1\}$, $\{e_2\}$, $\{e_1,\ e_3\}$, $\{e_1,\ e_2\}$, $\{e_1,\ e_2,\ e_3\}$, $\{e_1,\ e_2,\ e_4\}$.

Let $C' \subseteq C \in \mathcal{C}(\mathcal{E})$. Then C' is a *step* if $\forall e_1,\ e_2 \in C' \diamond \neg(e_1 <_\mathcal{E} e_2)$; the *restriction* of $\mathcal{E}$ to C' is defined as $\mathcal{E} \lceil C' = (C',\ \leq_\mathcal{E} \cap (C' \times C'),\ \#_\mathcal{E} \cap (C' \times C'),\ l_\mathcal{E} \mid_{C'})$; we use $pom_\mathcal{E}(C) = \{(\mathcal{E} \lceil (C'' \setminus C))/_\simeq \mid\ C \subseteq C'' \in \mathcal{C}(\mathcal{E})\}$ to denote the set of *pomsets* of C. We denote by C' not only the set itself, but also the labeled partial order it induces by restricting $\leq_\mathcal{E}$ and $l_\mathcal{E}$ to C'. It will, hopefully, be clear from the context what is meant. In addition, we define causal relations over the set of configurations as follows. Let $C,\ C' \in \mathcal{C}(\mathcal{E})$. Then $C \to_\mathcal{E} C'$ iff $C \subseteq C'$; $C \xrightarrow{p}_\mathcal{E} C'$ iff $C \to_\mathcal{E} C'$ and $C' \setminus C = p$, where $p \in pom_\mathcal{E}(C)$. We use $\mapsto_\mathcal{E} = \{C \xrightarrow{a}_\mathcal{E} C' \in \to_\mathcal{E} \mid\ a \in Act,\ C, C' \in \mathcal{C}(\mathcal{E})\}$ to denote the immediate causality relation between configurations.

We introduce a behavioural bisimulation equivalence defined over the sets of configurations of event structures.

Definition 2.3. Let $\mathcal{E}$ and $\mathcal{F}$ be event structures, $\mathcal{B} \subseteq \mathcal{C}(\mathcal{E}) \times \mathcal{C}(\mathcal{F})$. Then $\mathcal{B}$ is an *interleaving bisimulation* between $\mathcal{E}$ and $\mathcal{F}$ iff $(\emptyset, \emptyset) \in \mathcal{B}$ and for any $(C, D) \in \mathcal{B}$ the following holds:

- if $C \xrightarrow{a}_\mathcal{E} C'$ such that $a \in Act$ and $C' \in \mathcal{C}(\mathcal{E})$, then there is $D' \in \mathcal{C}(\mathcal{F})$ such that $D \xrightarrow{a}_\mathcal{F} D'$ and $(C', D') \in \mathcal{B}$;
- if $D \xrightarrow{a}_\mathcal{F} D'$ such that $a \in Act$ and $D' \in \mathcal{C}(\mathcal{F})$, then there is $C' \in \mathcal{C}(\mathcal{E})$ such that $C \xrightarrow{a}_\mathcal{E} C'$ and $(C', D') \in \mathcal{B}$.

$\mathcal{E}$ and $\mathcal{F}$ are *interleaving bisimilar* (denoted by $\mathcal{E} \approx_i \mathcal{F}$) if there exists an interleaving bisimulation between $\mathcal{E}$ and $\mathcal{F}$. □

3. Operational semantics of algebra $PBPA^*$

In this section we consider an extension of a well-known algebra BPA^* (standing for Basic Process Algebra with the binary Kleene star operator, due to [15]) with a parallel operator. Moreover, we modify the operator of sequential composition. We take the process algebra BPA^* as a starting point, because it is capable to represent infinite processes in a very natural way and, after being extended with additional operations, it seems to be fine to fit the event structure model. The extended algebra $PBPA^*$ here considered reflects all basic relations between the processes: causality, concurrency and choice.

We now define the syntax of $PBPA^*$ over a fixed alphabet Act:

$$PBPA^*_{cf}(Act): \ r = a|(p||q)|(p;q)$$

is the set of conflict-free terms, where $a \in Act$ and $p, q \in PBPA^*_{cf}(Act)$;

$$PBPA^*(Act): \ s = a|(p||q)|(p+q)|(r;q)|(r*q)$$

is the set of all $PBPA^*$-terms, where $a \in Act$, $p, q \in PBPA^*(Act)$, and $r \in PBPA^*_{cf}(Act)$.

The semantics of the process algebras are often given using the notion of a labeled transition system. A *(labeled) transition system* over an alphabet Act is a triple $Tr = (V, \rightarrow, s)$, where V is a set of states; $\rightarrow \subseteq V \times Act \times V$ is a transition relation and $s \in V$ is the initial state. Two transition systems $Tr_1 = (V_1, \rightarrow_1, s_1)$ and $Tr_2 = (V_2, \rightarrow_2, s_2)$ are called *isomorphic* if there is a bijection $f : V_1 \rightarrow V_2$ such that $f(s_1) = s_2$ and f preserves the transition relation, i.e. for all $v, v' \in V_1$ and $a \in Act$ the following holds: $v \xrightarrow{a}_1 v' \Leftrightarrow f(v) \xrightarrow{a}_2 f(v')$.

We present the operational semantics of $PBPA^*$ by means of a transition system associated with each process represented by a $PBPA^*$-term. Over the set $PBPA^*(Act)$ we define a transition relation $\longrightarrow_{PBPA^*} \subseteq PBPA^*(Act) \times Act \times (PBPA^*(Act) \cup \{\surd\})$, where $\surd \notin PBPA^*(Act)$ is used to denote a successful termination. We write $p \xrightarrow{a}_{PBPA^*} q$ to denote the transition from the process (represented by the term) p to the process q, when performing the action $a \in Act$ given by the transition rules shown in Table 3.1.

Table 3.1.

(Ax) If $a \in Act$, then $a \xrightarrow{a}_{PBPA^*} \surd$

$x \xrightarrow{a}_{PBPA^*} \surd$	$x \xrightarrow{a}_{PBPA^*} x'$
(A1) $x + y \xrightarrow{a}_{PBPA^*} \surd$	(B1) $x + y \xrightarrow{a}_{PBPA^*} x'$
(A2) $y + x \xrightarrow{a}_{PBPA^*} \surd$	(B2) $y + x \xrightarrow{a}_{PBPA^*} x'$
(A3) $x;y \xrightarrow{a}_{PBPA^*} y$	(B3) $x;y \xrightarrow{a}_{PBPA^*} x';y$
(A4) $x\|\|y \xrightarrow{a}_{PBPA^*} y$	(B4) $x\|\|y \xrightarrow{a}_{PBPA^*} x'\|\|y$
(A5) $y\|\|x \xrightarrow{a}_{PBPA^*} y$	(B5) $y\|\|x \xrightarrow{a}_{PBPA^*} y\|\|x'$
(A6) $x * y \xrightarrow{a}_{PBPA^*} x * y$	(B6) $x * y \xrightarrow{a}_{PBPA^*} x';(x * y)$
(A7) $y * x \xrightarrow{a}_{PBPA^*} \surd$	(B7) $y * x \xrightarrow{a}_{PBPA^*} x'$

We define a bisimulation equivalence over the obtained algebraic system $(PBPA^*(Act), \longrightarrow_{PBPA^*})$ of transitions.

Definition 3.1. An *algebraic bisimulation* is a relation $\mathcal{R} \subseteq PBPA^*(Act) \times PBPA^*(Act)$ such that:

— $p\mathcal{R}q$ and $p \xrightarrow{a}_{PBPA^*} p' \in PBPA^*(Act) \Rightarrow \exists q' \in PBPA^*(Act) \diamond q \xrightarrow{a}_{PBPA^*} q'$ and $p'\mathcal{R}q'$;

— $p\mathcal{R}q$ and $q \xrightarrow{a}_{PBPA^*} q' \in PBPA^*(Act) \Rightarrow \exists p' \in PBPA^*(Act) \diamond p \xrightarrow{a}_{PBPA^*} p'$ and $p'\mathcal{R}q'$;

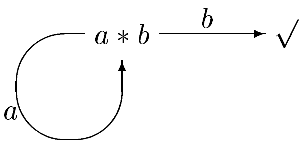
— $p\mathcal{R}q \Rightarrow (p \xrightarrow{a}_{PBPA^*} \surd \Leftrightarrow q \xrightarrow{a}_{PBPA^*} \surd)$. □

We call two $PBPA^*$-terms p and q equivalent ($p \underline{\leftrightarrow} q$) if there is an algebraic bisimulation $\mathcal{R}$ such that $p\mathcal{R}q$.

As an example, one can observe that the following term equivalences hold:

$$
\begin{aligned}
(x;y);z &\underline{\leftrightarrow} x;(y;z)\\
x*y &\underline{\leftrightarrow} x;(x*y)+y\\
x+y &\underline{\leftrightarrow} y+x\\
x||y &\underline{\leftrightarrow} y||x\\
(x+y)+z &\underline{\leftrightarrow} x+(y+z)\\
(x||y)||z &\underline{\leftrightarrow} x||(y||z)\\
x+x &\underline{\leftrightarrow} x
\end{aligned}
$$

Hence, the process $(a*b)$ for actions a and b can be depicted by:



4. Event structure semantics for $PBPA^*$

Now we wish to give the event structure semantics of $PBPA^*$-terms, where each $PBPA^*$-term defines an event structure up to isomorphism. For a given term $p \in PBPA^*(Act)$ we construct the event structure $\mathcal{E}_{PBPA^*}(p) = (E, \leq, \#, l)$ by induction on the structure of p as follows:

1. Let $p = a \in Act$. Then $\mathcal{E} = (\{e\}, \emptyset, \emptyset, \{(e,a)\})$.
2. Let $p = p_1||p_2$, $\mathcal{E}_1 = \mathcal{E}_{PBPA^*(Act)}(p_1)$ and $\mathcal{E}_2 = \mathcal{E}_{PBPA^*(Act)}(p_2)$ such that $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$. Then $\mathcal{E} = (E_{\mathcal{E}_1} \cup E_{\mathcal{E}_2}, \leq_{\mathcal{E}_1} \cup \leq_{\mathcal{E}_2}, \#_{\mathcal{E}_1} \cup \#_{\mathcal{E}_2}, l_{\mathcal{E}_1} \cup l_{\mathcal{E}_2})$.
3. Let $p = p_1 + p_2$, $\mathcal{E}_1 = \mathcal{E}_{PBPA^*(Act)}(p_1)$ and $\mathcal{E}_2 = \mathcal{E}_{PBPA^*(Act)}(p_2)$ such that $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$. Then $\mathcal{E} = (E_{\mathcal{E}_1} \cup E_{\mathcal{E}_2}, \leq_{\mathcal{E}_1} \cup \leq_{\mathcal{E}_2}, \#_{\mathcal{E}_1} \cup \#_{\mathcal{E}_2} \cup \{(e_1,e_2),(e_2,e_1) | e_1 \in E_{\mathcal{E}_1}, e_2 \in E_{\mathcal{E}_2}\}, l_{\mathcal{E}_1} \cup l_{\mathcal{E}_2})$.
4. Let $p = p_1;p_2$, $\mathcal{E}_1 = \mathcal{E}_{PBPA^*(Act)}(p_1)$ and $\mathcal{E}_2 = \mathcal{E}_{PBPA^*(Act)}(p_2)$ such that $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$. Then $\mathcal{E} = (E_{\mathcal{E}_1} \cup E_{\mathcal{E}_2}, \leq_{\mathcal{E}_1} \cup \leq_{\mathcal{E}_2} \cup \{(e_1,e_2) | e_1 \in E_{\mathcal{E}_1}, e_2 \in E_{\mathcal{E}_2}\}, \#_{\mathcal{E}_1} \cup \#_{\mathcal{E}_2}, l_{\mathcal{E}_1} \cup l_{\mathcal{E}_2})$.
5. Let $p = p_1 * p_2$. We assume $p^{(0)} = p_1 + p_2$ and $p^{(i+1)} = p_1;p^{(i)} + p_2$ for all $i \geq 0$. Then $\mathcal{E}$ is defined as the minimal structure such that $\mathcal{E}_{PBPA^*(Act)}(p^{(n)}) \sqsubseteq \mathcal{E}$ for all $n \in \mathbf{N}$.

Here event structures present iteration as unfolding. By construction, $\mathcal{E}_{PBPA^*}(p)$ is a prime event structure for all $p \in PBPA^*$.

Before establishing the correspondence between a bisimulation defined over event structures and an algebraic one, we introduce for each term $p \in PBPA^*(Act)$ the notion of a *process structure* which is a quadruple $Pr(p) = (\mathcal{P}, \to_p, s, l)$, where $\mathcal{P}$ is the *set of subprocesses* of the initial process $s \in \mathcal{P}$, $\to_p$ is a transition relation between processes over the alphabet Act and $l : \mathcal{P} \to PBPA^*(Act) \cup \{\surd\}$ is a labeling function. The process structure $Pr(p)$ is constructed according to the following rules:

($TR1$) Let $p = a \in Act$. Then $\mathcal{P} = \{v, s\}$, $\to_p = \{s \xrightarrow{a}_p v\}$, $l(s) = a$ and $l(v) = \surd$.

($TR2$) Let $p = p_1;p_2$ and $Pr(p_1) = (\mathcal{P}_1, \to_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \to_{p_2}, s_2, l_2)$ have been constructed so that $\mathcal{P}_1 \cap \mathcal{P}_2 = \{s_2\}$ and $l_1(s_2) = \surd$. Then $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$; $\to_p = \to_{p_1} \cup \to_{p_2}$; $s = s_1$; $l(v) = l_2(v)$ for all $v \in \mathcal{P}_2$, and $l(v) = l_1(v);p_2$ for all $v \in \mathcal{P}_1 \setminus \{s_2\}$.

($TR3$) Let $p = p_1 + p_2$ and $Pr(p_1) = (\mathcal{P}_1, \to_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \to_{p_2}, s_2, l_2)$ have been constructed so that $\mathcal{P}_1 \cap \mathcal{P}_2 = \{s_1\} = \{s_2\}$. Then $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$; $\to_p = \to_{p_1} \cup \to_{p_2}$; $s = s_1$ (and $s = s_2$); $l(v) = l_1(v)$ for all $v \in \mathcal{P}_1 \setminus \{s\}$, $l(v) = l_2(v)$ for all $v \in \mathcal{P}_2 \setminus \{s\}$ and $l(s) = l_1(s) + l_2(s)$.

($TR4$) Let $p = p_1||p_2$ and $Pr(p_1) = (\mathcal{P}_1, \to_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \to_{p_2}, s_2, l_2)$ have been already constructed. Then $\mathcal{P} = \mathcal{P}_1 \times \mathcal{P}_2$; $(v_1, v_2) \xrightarrow{a}_p (v_1', v_2') \Leftrightarrow ((v_1 \xrightarrow{a}_{p_1} v_1'$ & $v_2 = v_2') \vee (v_2 \xrightarrow{a}_{p_2} v_2'$ &

$v_1 = v_1')); \; s = (s_1, s_2);$

$$l(v,v') = \begin{cases} l_2(v') & \text{if } l_1(v) = \surd, \\ l_1(v) & \text{if } l_2(v') = \surd, \\ l_1(v)||l_2(v') & \text{otherwise.} \end{cases}$$

$(TR5)$ Let $p = p_1 * p_2$. For $0 \leq i$ we assume $Pr^i(p_1) = (\mathcal{P}_1^i, \rightarrow_{p_1}^i, s_1^i, l_1^i)$ and $Pr^i(p_2) = (\mathcal{P}_2^i, \rightarrow_{p_2}^i, s_2^i, l_2^i)$ to be process structures such that $(\mathcal{P}_1^i, \rightarrow_{p_1}^i, s_1^i) \cong (\mathcal{P}_1^j, \rightarrow_{p_1}^j, s_1^j)$ and $(\mathcal{P}_2^i, \rightarrow_{p_2}^i, s_2^i) \cong (\mathcal{P}_2^j, \rightarrow_{p_2}^j, s_2^j)$ for all i and j, and the following holds: $s_1^i = s_2^i (= s^i)$, $\mathcal{P}_1^i \cap \mathcal{P}_2^i = \{s^i\}$, $\mathcal{P}_1^i \cap \mathcal{P}_1^{i+1} = \{s^{i+1}\} = \mathcal{P}_1^i \cap \mathcal{P}_2^{i+1}$, where $l_1^i(s^{i+1}) = \surd$; $\mathcal{P}_2^i \cap \mathcal{P}_2^j = \emptyset$, with $i \neq j$; $\mathcal{P}_1^i \cap \mathcal{P}_1^j = \emptyset$, with $i \leq j-1$ and $j+1 \leq i$; $\mathcal{P}_1^i \cap \mathcal{P}_2^j = \emptyset$, with $i \neq j$ and $i \neq j-1$. Then $\mathcal{P} = \bigcup_{i \geq 0} (\mathcal{P}_1^i \cup \mathcal{P}_2^i)$; $s = s^0$; $\rightarrow_p = \bigcup_{i \geq 0} (\rightarrow_{p_1}^i \cup \rightarrow_{p_2}^i)$;

$$l(v) = \begin{cases} l_2^i(v) & \text{if } v \in \mathcal{P}_2^i \setminus \{s^i\} \text{ for } 0 \leq i, \\ l_1^i(v);(p_1 * p_2) & \text{if } v \in \mathcal{P}_1^i \setminus \{s^i, s^{i+1}\} \text{ for } 0 \leq i, \\ l_1^i(v) * l_2^i(v) & \text{if } v = s^i \text{ for } 0 \leq i. \end{cases}$$

In a process structure, two different processes can be labeled by the same $PBPA^*$-term, which means that these processes behave in the same way while occurring in different possible computations of the modeled system. Fig. 4.1 (a) shows the transition system of $PBPA^*(Act)$ taking the process $p = (b;a) * (a+b)$ as the initial states, whereas Fig. 4.1 (b) reflects the finite fragment of the corresponding process structure $Pr(p)$.

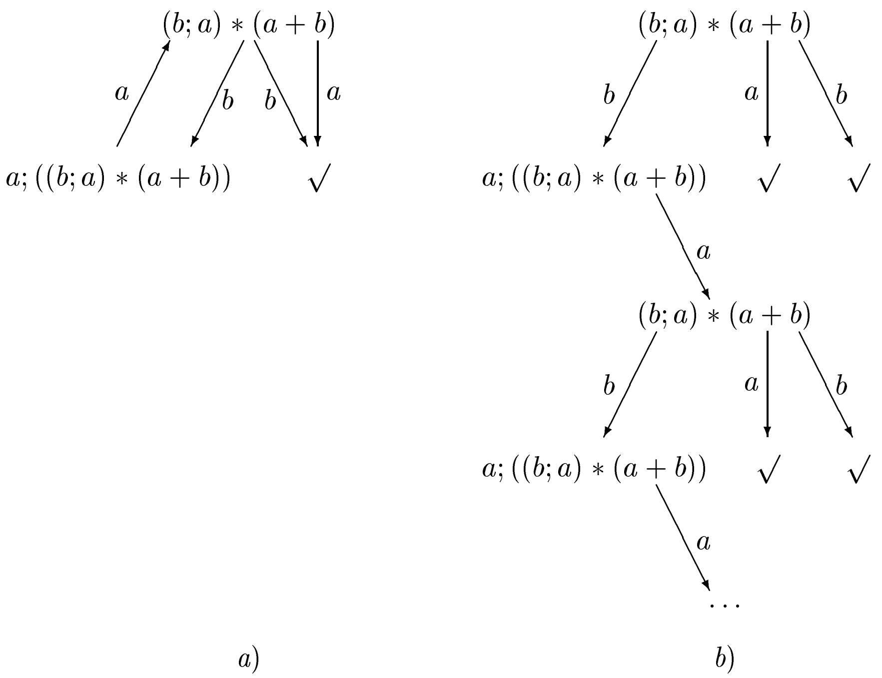


Figure 4.1.

Lemma 4.1. Let $Pr(p) = (\mathcal{P}, \rightarrow_p, s, l)$ be a process structure for $p \in PBPA^*(Act)$. (TR1)-(TR5). Then
(i) $l(s) = p$;
(ii) if $p \in PBPA_{cf}^*(Act)$, then $|\{v \in \mathcal{P} | l(v) = \surd\}| = 1$;
(iii) $l(v) = \surd \Rightarrow \{v' \in \mathcal{P} | \exists a \in Act : v \xrightarrow{a}_p v'\} = \emptyset$.

Proof. We prove **(i)** by induction on the structure of the term p.

- $p = a \in Act$. According to (TR1), we have $l(s) = a = p$.

- $p = p_1||p_2$. Let, for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \rightarrow_{p_2}, s_2, l_2)$, it be proved that $l_1(s_1) = p_1$ and $l_2(s_2) = p_2$. According to (TR4), we have $s = (s_1, s_2)$ and $l(s_1, s_2) = l_1(s_1)||l_2(s_2) = p_1||p_2 = p$, since $l_1(s_1) \neq \surd$ and $l_2(s_2) \neq \surd$.
- $p = p_1 + p_2$. Let, for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \rightarrow_{p_2}, s_2, l_2)$, it be proved that $l_1(s_1) = p_1$ and $l_2(s_2) = p_2$. According to (TR3), we have $s = s_1 = s_2$ and $l(s) = l_1(s) + l_2(s) = l_1(s_1) + l_2(s_2) = p_1 + p_2 = p$.
- $p = p_1;p_2$. Let, for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$, it be proved that $l_1(s_1) = p_1$. According to (TR2), we have $s = s_1$ and $l_1(s_2) = \surd$, where s_2 is the initial state in $Pr(p_2)$. Hence, $s \in \mathcal{P} \setminus \{s_2\}$ and, according to (TR2), we get $l(s) = l_1(s);p_2 = l_1(s_1);p_2 = p_1;p_2 = p$.
- $p = p_1 * p_2$. Let, for $Pr^0(p_1) = (\mathcal{P}_1^0, \rightarrow_{p_1}^0, s_1^0, l_1^0)$ and $Pr^0(p_2) = (\mathcal{P}_2^0, \rightarrow_{p_2}^0, s_2^0, l_2^0)$, it be proved that $l_1^0(s_1^0) = p_1$ and $l_2^0(s_2^0) = p_2$. According to (TR4), we have $s = s^0 = s_1^0 = s_2^0)$ and $l(s) = l_1^0(s^0) * l_2^0(s^0) = p_1 * p_2 = p$.

We prove **(ii)** by induction on the structure of the term p.

- $p = a \in Act$. Obvious, due to (TR1).
- $p = p_1;p_2$. Let, for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \rightarrow_{p_2}, s_2, l_2)$, it be proved that there are the only node $v \in \mathcal{P}_1$ and the only node $v' \in \mathcal{P}_2$ labeled by $\surd$. According to (TR2), we have $v = s_2$ and $l(v) = l_2(s_2) = p_2 \in PBPA^*$; $\forall\, \widetilde{v} \in \mathcal{P}_1 \setminus \{v\} \diamond l(\widetilde{v}) = l_1(\widetilde{v});p_2 \in PBPA^*$ and $\forall\, \widetilde{v} \in \mathcal{P}_2 \diamond l(\widetilde{v}) = l_2(\widetilde{v})$. It is obvious that there is the only node v' in $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$ labeled by $\surd$.
- $p = p_1||p_2$. Let, for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \rightarrow_{p_2}, s_2, l_2)$, it be proved that there are the only node $v \in \mathcal{P}_1$ and the only node $v' \in \mathcal{P}_2$ such that $l_1(v) = \surd$ and $l_2(v') = \surd$. According to (TR4), we have $l(v_1, v_2) = \surd \Leftrightarrow (l_1(v_1) = \surd$ and $l_2(v_2) = \surd)$. This is possible only if $v_1 = v$ and $v_2 = v'$. So, there is the only node $(v, v') \in \mathcal{P}$ such that $l(v, v') = \surd$.

We prove **(iii)** by induction on the structure of the term p.

- $p = a \in Act$. Obvious, according to (TR1).
- $p = p_1;p_2$. By the point (ii) of the lemma, we have that there is the only node $v_0 \in \mathcal{P}_1$ such that $l_1(v_0) = \surd$, since $p_1 \in PBPA^*_{cf}$. According to (TR2), we have $v_0 = s_2 \in \mathcal{P}_2$ and $\forall v' \in \mathcal{P}_1 \setminus \{v_0\} \diamond l(v') = l_1(v');p_2 \in PBPA^*$, since $l_1(v') \neq \surd$. For $v' \in \mathcal{P}_2$ we have $l(v) = l_2(v)$. Hence, $l(v) = \surd \Leftrightarrow (v \in \mathcal{P}_2 \ \&\ l_2(v) = \surd)$. Let $\widetilde{v} \in \mathcal{P}_2$ be such that $l_2(\widetilde{v}) = \surd$. Then, by the induction hypothesis, we get $\{v' \in \mathcal{P}_2 | \exists a \in Act \diamond \widetilde{v} \xrightarrow{a}_{p_2} v'\} = \emptyset$. Since $\rightarrow_{p_1} \cap \rightarrow_{p_2} = \emptyset$ implied from (TR2), we have $\forall v \in \mathcal{P}_2 \diamond \{v' \in \mathcal{P}_2 | \exists a \in Act \diamond v \xrightarrow{a}_{p_2} v'\} = \{v' \in \mathcal{P} | \exists a \in Act \diamond \widetilde{v} \xrightarrow{a}_p v'\}$. Therefore, $\{v' \in \mathcal{P} | \exists a \in Act \diamond \widetilde{v} \xrightarrow{a}_p v'\} = \emptyset$.
- $p = p_1 + p_2$. Let $v \in \mathcal{P}$ be such that $l(v) = \surd$. We assume $v \in \mathcal{P}_1$ (the case "$v \in \mathcal{P}_2$" can be proved in the similar way). Since $\rightarrow_{p_1} \cap \rightarrow_{p_2} = \emptyset$ implied from (TR3), we have $\{v' \in \mathcal{P} | \exists a \in Act \diamond v \xrightarrow{a}_p v'\} = \{v' \in \mathcal{P}_1 | \exists a \in Act \diamond v \xrightarrow{a}_{p_1} v'\} = \emptyset$, due to the induction hypothesis.
- $p = p_1||p_2$. Let $(v_1, v_2) \in \mathcal{P}$ be such that $l(v_1, v_2) = \surd$. According to (TR4), this is possible if $l_1(v_1) = \surd$ and $l_2(v_2) = \surd$. Let us consider $(\widetilde{v}_1, \widetilde{v}_2) \in \mathcal{P}$ such that $l_1(\widetilde{v}_1) = \surd$ and $l_2 = (\widetilde{v}_2) = \surd$. Then, according to (TR4), we get $\{(v_1, v_2) \in \mathcal{P} | \exists a \in Act \diamond (\widetilde{v}_1, \widetilde{v}_2) \xrightarrow{a}_p (v_1, v_2)\}$ $= \{(\widetilde{v}_1, v_2) \in \mathcal{P} | \exists a \in Act \diamond (\widetilde{v}_1, \widetilde{v}_2) \xrightarrow{a}_p (\widetilde{v}_1, v_2)\} \cup \{(v_1, \widetilde{v}_2) \in \mathcal{P} | \exists a \in Act \diamond (\widetilde{v}_1, \widetilde{v}_2) \xrightarrow{a}_p (v_1, \widetilde{v}_2)\}$ $= \{(\widetilde{v}_1, v_2) \in \mathcal{P} | \exists a \in Act \diamond \widetilde{v}_2 \xrightarrow{a}_{p_2} v_2\} \cup \{(v_1, \widetilde{v}_2) \in \mathcal{P} | \exists a \in Act \diamond \widetilde{v}_1 \xrightarrow{a}_{p_1} v_1\} = \emptyset \cup \emptyset = \emptyset$.
- $p = p_1 * p_2$. Let $Pr^i(p_1) = (\mathcal{P}_1^i, \rightarrow_{p_1}^i, s_1^i, l_1^i)$ and $Pr^i(p_2) = (\mathcal{P}_2^i, \rightarrow_{p_2}^i, s_2^i, l_2^i)$ be the process structures constructed according to (TR5) for $0 \leq i$. We consider $v_0 \in \mathcal{P}$ such that $l(v_0) = \surd$. Then $v_0 \notin \mathcal{P}_1^i$ for all $i \geq 0$, since for $v \in \mathcal{P}_1^i \setminus \{s^i, s^{i+1}\}$ the following holds: $l(v) = l_1^i(v);(p_1 * p_2) \in PBPA^*$ and $l(s^i) = l(s^{i+1}) = p_1 * p_2$ for $0 \leq i$, due to the point (i) of the lemma. Hence, $v_0 \in \mathcal{P}_2^i$ for some $0 \leq i$. According to (TR5), we have that $\rightarrow_{p_j}^i$ are disjoint for $j = 1, 2$ and $0 \leq i$ and, so, $\{v \in \mathcal{P} | \exists a \in Act \diamond v \xrightarrow{a}_p v_0\} = \{v \in \mathcal{P}_2^i | \exists a \in Act \diamond v \xrightarrow{a\ i}_{p_2} v_0\} = \emptyset$ by the induction hypothesis. $\square$

The following proposition shows that the process structures adequately present the transition system of the algebra $PBPA^*$, i.e. that defined by the rules (Ax), (A1)-(A7) and (B1)-(B7).

Proposition 4.1. Let $p \in PBPA^*(Act)$ and $Pr(p) = (\mathcal{P}, \rightarrow_p, s, l)$ be the process structure for p. Then, for any $v_1, v_2 \in \mathcal{P}$, it holds that $v_1 \xrightarrow{a}_p v_2 \Rightarrow l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$.

Proof. We prove by induction on the structure of the term p.

1. $p = a \in Act$. If $v_1 \xrightarrow{a}_p v_2$, then $l(v_1) = a$ and $l(v_2) = \surd$, according to (TR1). And $a \xrightarrow{a}_{PBPA^*} \surd$ is a transition in $PBPA^*(Act)$, due to (Ax).
2. $p = p_1;p_2$. Let the proposition be proved for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \rightarrow_{p_2}, s_2, l_2)$. According to (TR2), we have $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$. We assume that $v_1 \xrightarrow{a}_p v_2$. Since $\xrightarrow{a}_p = \xrightarrow{a}_{p_1} \cup \xrightarrow{a}_{p_2}$, it is easy to see that $\{v_1, v_2\} \subseteq \mathcal{P}_1$ or $\{v_1, v_2\} \subseteq \mathcal{P}_2$. If $\{v_1, v_2\} \subseteq \mathcal{P}_2$, then $l(v_1) = l_2(v_1)$, $l(v_2) = l_2(v_2)$ and, by the induction hypothesis, $l_2(v_1) \xrightarrow{a}_{PBPA^*} l_2(v_2)$. If $\{v_1, v_2\} \subseteq \mathcal{P}_1$, then $l(v_1) = l_1(v_1);p_2$,
   $$l(v_2) = \begin{cases} p_2 & \text{if } v_2 = s_2 \text{ ;} \\ l_1(v_2);p_2 & \text{otherwise.} \end{cases}$$
   Let $v_2 = s_2$. Then $l_1(v_2) = \surd$, due to Lemma 4.1 (i) and according to (TR2). By the induction hypothesis, we have $l_1(v_1) \xrightarrow{a}_{PBPA^*} \surd$. According to (A3), we get $l_2(v_1);p_2 \xrightarrow{a}_{PBPA^*} p_2$. We assume that $v_2 \neq s_2$. Then $l_1(v_2) \in PBPA^*(Act)$. By the induction hypothesis, $l_1(v_1) \xrightarrow{a}_{PBPA^*} l_1(v_2)$. Then, according to (B3), we have $l_1(v_1);p_2 \xrightarrow{a}_{PBPA^*} l_1(v_2);p_2$.
3. $p = p_1 + p_2$. Let the proposition be proved for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \rightarrow_{p_2}, s_2, l_2)$. Let us consider $v_1, v_2 \in \mathcal{P}$ such that $v_1 \xrightarrow{a}_p v_2$. Since $\xrightarrow{a}_p = \xrightarrow{a}_{p_1} \cup \xrightarrow{a}_{p_2}$, it is easy to see that $\{v_1, v_2\} \subseteq \mathcal{P}_1$ or $\{v_1, v_2\} \subseteq \mathcal{P}_2$. We assume that $\{v_1, v_2\} \subseteq \mathcal{P}_1$ (the case $\{v_1, v_2\} \subseteq \mathcal{P}_2$ is proved in the similar way). Let us consider two possible cases. We suppose that $v_1 \in \mathcal{P}_1 \setminus \{s_1\}$, then $l(v_1) = l_1(v_1)$ and $l(v_2) = l_1(v_2)$. By the induction hypothesis, $l_1(v_1) \xrightarrow{a}_{PBPA^*} l_1(v_2)$ and, hence, $l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$. Assume that $v_1 = s_1$. Then $l_1(v_1) = p_1$, due to Lemma 3.1 (i), and $l(v_1) = p_1 + p_2$, according to (TR3). By the induction hypothesis, we have $p_1 \xrightarrow{a}_{PBPA^*} l_1(v_2)$. Then, according to (B2) (or (A2) if $l_1(v_2) = \surd$), we have $p_1 + p_2 \xrightarrow{a}_{PBPA^*} l_1(v_2)$. Therefore, $l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$.
4. $p = p_1||p_2$. Let the proposition be proved for $Pr(p_1) = (\mathcal{P}_1, \rightarrow_{p_1}, s_1, l_1)$ and $Pr(p_2) = (\mathcal{P}_2, \rightarrow_{p_2}, s_2, l_2)$. According to (TR4), we have $\mathcal{P} = \mathcal{P}_1 \times \mathcal{P}_2$. Let us consider $(v_1, v_1'), (v_2, v_2') \in \mathcal{P}$ such that $(v_1, v_1') \xrightarrow{a}_p (v_2, v_2')$. According to (TR4), we suppose that $v_1' = v_2'$ and $v_1 \xrightarrow{a}_{p_1} v_2$ (the case of $v_1 = v_2$ and $v_1' \xrightarrow{a}_{p_1} v_2'$ is proved in the similar way). By the induction hypothesis, we have $l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$, $l_2(v_1') = l_2(v_2') = q \in PBPA^* \cup \{\surd\}$. According to (TR4), we get $l(v_1, v_1') = l_1(v_1)||q$ and $l(v_2, v_2') = l_1(v_2)||q$ if $q \neq \surd$, and $l(v_1, v_1') = l_1(v_1)$, $l(v_2, v_2') = l_1(v_2)$ if $q = \surd$. Then $q = \surd$, and we get $l(v_1, v_1') = l_1(v_1) \xrightarrow{a}_{PBPA^*} l(v_2) = l(v_2, v_2')$. For $q \neq \surd$, according to (B4), we get $l(v_1, v_1') = l_1(v_1)||q \xrightarrow{a}_{PBPA^*} l(v_2)||q$.
5. $p = p_1 * p_2$. Let the proposition be proved for $Pr^i(p_1) = (\mathcal{P}_1^i, \rightarrow_{p_1}^i, s_1^i, l_1^i)$ and $Pr^i(p_2) = (\mathcal{P}_2^i, \rightarrow_{p_2}^i, s_2^i, l_2^i)$ (with $i \geq 0$). Let us consider $v_1, v_2 \in \mathcal{P}$ such that $v_1 \xrightarrow{a}_p v_2$. According to (TR5), we have $\xrightarrow{a}_p = \bigcup_{i \leq 0}(\xrightarrow{a}{}^i_{p_1} \cup \xrightarrow{a}{}^i_{p_2})$. Therefore, $\{v_1, v_2\} \subseteq \mathcal{P}_1^i$ or $\{v_1, v_2\} \subseteq \mathcal{P}_2^i$ for some $i \geq 0$. Two cases are worth to be considered:

   1) $\{v_1, v_2\} \subseteq \mathcal{P}_1^i$. Four cases are possible:

   - Let $v_1 = s^i$ (with $s^i = s_1^i = s_2^i$) and $v_2 = s^{i+1}$. This means that $\mathcal{P}_1^i = \{v_1, v_2\}$. According to (TR5), it implies that $p_1 = a \in Act$, since $v_1 \xrightarrow{a}_{p_1} v_2$. We have $l(v_1) = p_1 * p_2 = a * p_2$ and $l(v_2) = p_1 * p_2 = a * p_2$, since $p_1 = a \xrightarrow{a}_{PBPA^*} \surd$, due to the axiom (Ax). According to (A6), we get $l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$.

- Let $v_1 = s^i$ and $v_2 \neq s^{i+1}$. By the induction hypothesis, $p_1 = l_1^i(v_1) \xrightarrow{a}_{PBPA^*} l_1^i(v_2)$. According to (TR5), we get $l(v_1) = p_1 * p_2$ and $l(v_2) = l_1^i(v_2);(p_1 * p_2)$. Then, according to (B6), we get $l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$.
- Let $v_1 \neq s^i$ and $v_2 = s^{i+1}$. According to (TR5), we have $l(v_1) = l_1^i(v_2);(p_1 * p_2)$ and $l(v_1) = (p_1 * p_2)$, due to Lemma 4.1. By the induction hypothesis, we have $l(v_1^i) \xrightarrow{a}_{PBPA^*} \surd$. Then, according to (A3), we get $l^i(v_1);(p_1 * p_2) = l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2) = (p_1 * p_2)$.
- Let $v_1 \neq s^i$ and $v_2 \neq s^{i+1}$. Then $l(v_1) = l^i(v_1);(p_1 * p_2)$ and $l(v_2) = l^i(v_2);(p_1 * p_2)$. By the induction hypothesis, $l^i(v_1) \xrightarrow{a}_{PBPA^*} l^i(v_2)$ and, according to (B3), we get $l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$.

2) $\{v_1, v_2\} \subseteq \mathcal{P}_2^i$. According to (TR5), we get $v_2 \neq s^i$. Therefore, $l(v_2) = l_2^i(v_2)$. Let us consider two possible cases:

- Let $v_1 = s^i$ (with $s^i = s_1^i = s_2^i$). According to (TR5) and Lemma 4.1 (i), we have $l(v_1) = p_1 * p_2$ and $l_2^i(v_1) = p_2$. By the induction hypothesis, $l^i(v_1) \xrightarrow{a}_{PBPA^*} l^i(v_2)$, i.e. $p_2 \xrightarrow{a}_{PBPA^*} l^i(v_2)$. Then, according to (B7) (or (A7) if $l_2^i(v_2) = \surd$), we have $l(v_1) = (p_1 * p_2) \xrightarrow{a}_{PBPA^*} l(v_2)$.
- Let $v_1 \neq s^i$. Then $l(v_1) = l_2^i(v_1)$. By the induction hypothesis, $l_2^i(v_1) \xrightarrow{a}_{PBPA^*} l_2^i(v_2)$, which means that $l(v_1) \xrightarrow{a}_{PBPA^*} l(v_2)$. □

The following theorem establishes the correspondence between the transition system for a process defined by a $PBPA^*(Act)$-term and the transition system defined over the set of configurations of the event structure constructed for the $PBPA^*(Act)$-term.

Theorem 4.1. For $p \in PBPA^*(Act)$, let $Pr(p) = (\mathcal{P}, \rightarrow_p, s, l)$ be the process structure and $\mathcal{E} = \mathcal{E}_{PBPA^*}(p)$. Then $(\mathcal{C}(\mathcal{E}), \mapsto_{\mathcal{E}}, \emptyset) \cong (\mathcal{P}, \rightarrow_p, s)$.

Proof. We prove the theorem by induction on the structure of p.

1. $p = a \in Act$. Then $\mathcal{C}(\mathcal{E}) = \{\emptyset, \{e\}\}$, where $l_{\mathcal{E}}(e) = a$; $\mapsto_{\mathcal{E}} = \{\emptyset \xrightarrow{a}_{\mathcal{E}} \{e\}\}$. According to (TR1), we have $\mathcal{P} = \{v_1, v_2\}$, $\rightarrow_p = \{v_1 \xrightarrow{a}_p v_2\}, s = v_1$. Let us consider the mapping $f : C(\mathcal{E}) \rightarrow \mathcal{P}$ such that $f(\emptyset) = v_1$ and $f(\{e\}) = v_2$. It is easy to see that f is an isomorphism between $(\mathcal{C}(\mathcal{E}), \mapsto_{\mathcal{E}}, \emptyset)$ and $(\mathcal{P}, \rightarrow_p, s)$.

2. $p = p_1;p_2$. Let us consider $\mathcal{E}_1 = \mathcal{E}_{PBPA^*}(p_1)$ and $\mathcal{E}_2 = \mathcal{E}_{PBPA^*}(p_2)$ such that $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$. Let $Pr(p_i) = (\mathcal{P}_i, \rightarrow_{p_i}, s_i, l_i)$ be the process structure for p_i, with $i = 1, 2$. By the induction hypothesis, there are isomorphisms $f_i : \mathcal{C}(\mathcal{E}_i) \rightarrow \mathcal{P}_i$, with $i = 1, 2$. By construction of $\mathcal{E}_{PBPA^*}(p)$ and by definition of a $PBPA^*(Act)$-term, it is easy to see that $\mathcal{E}_1$ is a conflict-free event structure. Therefore, $E_{\mathcal{E}_1} \in \mathcal{C}(\mathcal{E}_1)$ is the maximal configuration in $\mathcal{E}_1$. By construction of $\mathcal{E}_{PBPA^*}(p)$, we get $\mathcal{C}(\mathcal{E}) = \mathcal{C}(\mathcal{E}_1) \cup \{E_{\mathcal{E}_1} \cup C' | C' \in \mathcal{C}(\mathcal{E}_2)\}$. Since f_1 preserves the transition relation, due to Lemma 4.1 (ii,iii) we get $l_1(f_1(E_{\mathcal{E}_1})) = \surd$ and $\forall v \in \mathcal{P}_1 \diamond l_1(v) = \surd \Rightarrow v = f_1(E_{\mathcal{E}_1})$. According to (TR2), we have $f_1(E_{\mathcal{E}_1}) = s_2$. Then the mapping $f : \mathcal{C}(\mathcal{E}) \rightarrow \mathcal{P}$, such that $f(C) = f_1(C)$ for $C \in \mathcal{C}(\mathcal{E}_1)$ and $f(C) = f_2(C \setminus E_{\mathcal{E}_1})$ for $C \in \{E_{\mathcal{E}_1} \cup C' | C' \in \mathcal{C}(\mathcal{E}_2)\}$, is bijective.

We assume that $C \xrightarrow{a}_{\mathcal{E}} C'$. Two cases are worth to be considered.

- Let $C' \in \mathcal{C}(\mathcal{E}_1)$. Then, obviously, $C \in \mathcal{C}(\mathcal{E}_1)$. So, $f(C) = f(C_1)$ and $f(C') = f_1(C')$. Since f_1 preserves the transition relation, $f(C) \xrightarrow{a}_p f(C')$.
- Let $C' \in \{E_{\mathcal{E}_1} \cup \overline{C} | \overline{C} \in \mathcal{C}(\mathcal{E}_2)\} \setminus \mathcal{C}(\mathcal{E}_1)$. Then $C' \neq E_{\mathcal{E}_1}$, which means that $C' = E_{\mathcal{E}_1} \cup \overline{C}$ and $\overline{C} \in \mathcal{C}(\mathcal{E}_2) \setminus \{\emptyset\}$. By definition of the relation $\xrightarrow{a}_{\mathcal{E}}$, we have $C = C' \setminus \{e\}$ for some $e \in E_{\mathcal{E}}$ such that $l_{\mathcal{E}} = a$. Hence, $f(C) = f_2(C \setminus E_{\mathcal{E}_1})$, $f(C') = f_2(C' \setminus E_{\mathcal{E}_1})$, and $C \setminus E_{\mathcal{E}_1} \xrightarrow{a}_{\mathcal{E}_2} C' \setminus E_{\mathcal{E}_1}$. Since f_2 preserves the transition relation, $f(C) \xrightarrow{a}_p f(C')$.

We assume that $f(C) \xrightarrow{a}_p f(C')$. If $f(C') \in \mathcal{P}_1$ then, according to (TR2), we get $f(C) \in \mathcal{P}_1$. Hence, $f(C) \xrightarrow{a}_{p_1} f(C')$, $f(C) = f_1(C)$ and $f(C') = f_1(C')$. Since f_1 preserves the transition relation,

$C \xrightarrow{a}_{\mathcal{E}_1} C'$. Since $\rightarrow_{\mathcal{E}} = \rightarrow_{\mathcal{E}_1} \cup \rightarrow_{\mathcal{E}_2}$, $C \xrightarrow{a}_{\mathcal{E}} C'$. If $f(C') \in \mathcal{P}_2 \setminus \mathcal{P}_1$ then, according to (TR2), we have $f(C) \in \mathcal{P}_2$. Hence, $f(C) \xrightarrow{a}_{p_2} f(C')$ and $f(C) = f(C \setminus E_{\mathcal{E}_1})$ and $f(C) = f_2(C' \setminus E_{\mathcal{E}_1})$. Since f_2 preserves the transition relation, $C \setminus E_{\mathcal{E}_1} \xrightarrow{a}_{\mathcal{E}_2} C' \setminus E_{\mathcal{E}_1}$, by definition of the relation $\xrightarrow{a}_{\mathcal{E}_2}$. This means that $C = C' \setminus \{e\}$ for some $e \in E_{\mathcal{E}_2}$ such that $l_{\mathcal{E}_2}(e) = a$. Since $E_{\mathcal{E}} = E_{\mathcal{E}_1} \cup E_{\mathcal{E}_2}$ and $l_{\mathcal{E}} = l_{\mathcal{E}_1} \cup l_{\mathcal{E}_2}$, $e \in E_{\mathcal{E}}$ and $l_{\mathcal{E}}(e) = a$. Therefore, $C \xrightarrow{a}_{\mathcal{E}} C'$.

3. $p = p_1 + p_2$. Let us consider $\mathcal{E}_1 = \mathcal{E}_{PBPA^*}(p_1)$ and $\mathcal{E}_2 = \mathcal{E}_{PBPA^*}(p_2)$ such that $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$. Let $Pr(p_i) = (\mathcal{P}_i, \xrightarrow{a}_{p_i}, s_i, l_i)$ be the process structures for p_i, with $i = 1, 2$. By the induction hypothesis, there are isomorphisms $f_i : \mathcal{C}(\mathcal{E}) = \mathcal{C}(\mathcal{E}_1) \cup \mathcal{C}(\mathcal{E}_2)$ and $\mathcal{C}(\mathcal{E}_1) \cap \mathcal{C}(\mathcal{E}_2) = \{\emptyset\}$. According to (TR3), we have $s_1 = s_2$. Therefore, $f_1(\emptyset) = f_2(\emptyset)$. Hence, the mapping $f = f_1 \cup f_2 : \mathcal{C}(\mathcal{E}) \to \mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$, such that $f(C) = f_1(C)$ with $C \in \mathcal{C}(\mathcal{E}_1)$ and $f(C) = f_2(C)$ with $C \in \mathcal{C}(\mathcal{E}_2)$, is a bijection.

We assume that $C \xrightarrow{a}_{\mathcal{E}} C'$ and $C' \in \mathcal{C}(\mathcal{E}_1)$ (the case $C' \in \mathcal{C}(\mathcal{E}_2)$ is proved in a similar way). Then, obviously, $C \in \mathcal{C}(\mathcal{E}_1)$ and, hence, $C \xrightarrow{a}_{\mathcal{E}_1} C'$. Since f_1 preserves the transition relation, $f_1(C) \xrightarrow{a}_{p_1} f_1(C')$. Since $\xrightarrow{a}_p = \xrightarrow{a}_{p_1} \cup \xrightarrow{a}_{p_2}$ according to $(TR3)$, $f_1(C) \xrightarrow{a}_p f_1(C')$.

We now assume that $f_1(C) \xrightarrow{a}_p f_1(C')$. We have to show $C \xrightarrow{a}_{\mathcal{E}} C'$. According to $(TR3)$, we have $f(C) \xrightarrow{a}_p f(C)' \iff (f(C) \xrightarrow{a}_{p_1} f(C') \vee f(C) \xrightarrow{a}_{p_2} f(C'))$. We suppose that $f(C) \xrightarrow{a}_{p_1} f(C')$ (the remained case is proved analogously), then $f(C), f(C') \in \mathcal{P}_1$. Since f is a bijective function, C and $C' \in \mathcal{C}(\mathcal{E}_1)$. Moreover, $f(C) = f_1(C)$ and $f(C') = f_1(C')$. Since f_1 preserves the transition relation, $C \xrightarrow{a}_{\mathcal{E}_1} C'$ and, hence, $C \xrightarrow{a}_{\mathcal{E}} C'$ (since $\xrightarrow{a}_{\mathcal{E}} = \xrightarrow{a}_{\mathcal{E}_1} \cup \xrightarrow{a}_{\mathcal{E}_2}$), by construction of $\mathcal{E}_{PBPA^*}(p)$.

4. $p = p_1 || p_2$. Let us consider $\mathcal{E}_1 = \mathcal{E}_{PBPA*}(p_1)$ and $\mathcal{E}_2 = \mathcal{E}_{PBPA*}(p_2)$ such that $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$. We assume $Pr(p_i) = (\mathcal{P}_i, \xrightarrow{a}_{p_i}, s_i, l_i)$ to be the process structures for p_i, with $i = 1, 2$. By the induction hypothesis, there are isomorphisms $f_i : \mathcal{C}(\mathcal{E}_i) \to \mathcal{P}_i$, with $i = 1, 2$. By the construction of $\mathcal{E}_{PBPA*}(p)$, it is easy to see that $\mathcal{C}(\mathcal{E}) = \{C \cup C' | C \in \mathcal{C}(\mathcal{E}_1), C' \in \mathcal{C}(\mathcal{E}_2)\}$. Since $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$, each configuration $C \in \mathcal{C}$ can be represented as $C = C_1 \cup C_2$ in the only way, where $C \in \mathcal{C}(\mathcal{E}_1)$ and $C \in \mathcal{C}(\mathcal{E}_2)$. Therefore, one can take $\mathcal{C}(\mathcal{E}) = \{[C_1, C_2] | C_1 \in \mathcal{C}(\mathcal{E}_1), C_2 \in \mathcal{C}(\mathcal{E}_2)\} = \mathcal{C}(\mathcal{E}_1) \times \mathcal{C}(\mathcal{E}_2)$. Let us consider the mapping $f : \mathcal{C}(\mathcal{E}) \to \mathcal{P} = \mathcal{P}_1 \times \mathcal{P}_2$ such that $f(C_1, C_2) = (f_1(C_1), f_2(C_2))$. We need to show that f is an isomorphism. Since f_1 and f_2 are surjective and injective, it is obvious that $f = f_1 \times f_2$ is also surjective and injective.

We suppose that $C \xrightarrow{a}_{\mathcal{E}} C'$. We have to show $f(C) \xrightarrow{a}_p f(C')$. By definition of the relation $\xrightarrow{a}_{\mathcal{E}}$, we have $C' \setminus C = \{e\}$ and $l_{\mathcal{E}}(e) = a$. We assume that $C' = (C_1', C_2')$ and $e \in C_1'$ (the case $e \in C_2'$ is proved in a similar way). Then $C_2' = C_2$, where $C = (C_1, C_2)$. Hence, $C_1' \setminus C_1 = \{e\}$, where $l_{\mathcal{E}_1}(e) = a$, $f(C') = (f_1(C_1'), f_2(C_2))$, $f(C) = (f_1(C_1), f_2(C_2))$ and $C_1 \xrightarrow{a}_{\mathcal{E}_1} C_1'$. Since f_1 preserves the transition relation, $f_1(C_1) \xrightarrow{a}_{p_1} f_1(C_1')$. According to $(TR4)$ we have $(f_1(C_1), f_2(C_2)) \xrightarrow{a}_p (f_1(C_1'), f_2(C_2))$, i.e. $f(C) \xrightarrow{a}_p f(C')$.

We now assume that $f(C) \xrightarrow{a}_p f(C')$. We need to show that $C \xrightarrow{a}_{\mathcal{E}} C'$. We suppose $f(C') = (v_1', v_2') \in \mathcal{P}$ and $f(C) = (v_1, v_2) \in \mathcal{P}$. According to $(TR4)$, we get $(v_1 \xrightarrow{a}_{p_1} v_1'$ & $v_2 = v_2')$ or $(v_2 \xrightarrow{a}_{p_2} v_2'$ & $v_1 = v_1')$. Let us consider the case $v_2 = v_2'$ & $v_1 \xrightarrow{a}_{p_1} v_1'$ (the remained case is proved in the similar way). Suppose that $C = (C_1, C_2)$ and $C' = (C_1', C_2')$. Since f_2 is an isomorphism, $C_2 = C_2'$. Since f_1 preserves the transition relation, $(C_1 \xrightarrow{a}_{\mathcal{E}_1} C_1')$, i.e. $C_1' = C_1 \cup \{e\}$ and $l_{\mathcal{E}_1}(e) = a$. Then $C' = C_1' \cup C_2' = (C_1 \cup \{e\}) \cup C_2 = C \cup \{e\}$ and $l_{\mathcal{E}}(e) = a$, which means that $C \xrightarrow{a}_{\mathcal{E}} C'$.

5. $p = p_1 * p_2$. Let us take a countable set of event structures $\mathcal{E}_1^0$, $\mathcal{E}_2^0$, ..., $\mathcal{E}_1^i$, $\mathcal{E}_2^i$, ... such that $\mathcal{E}_1^i = \mathcal{E}_{PBPA^*}(p_1)$, $\mathcal{E}_2^i = \mathcal{E}_{PBPA^*}(p_2)$, and $(E^i_{\mathcal{E}_1} \bigcap E^j_{\mathcal{E}_1}) \bigcup (E^i_{\mathcal{E}_2} \bigcap E^j_{\mathcal{E}_2}) \bigcup (E^k_{\mathcal{E}_1} \bigcap E^l_{\mathcal{E}_2}) = \emptyset$ with $i \neq j$ and $\forall k, l \in \mathbf{N}$.

Let us consider $E = \bigcup_{i \geq 0} (E_{\mathcal{E}_1^i} \cup E_{\mathcal{E}_2^i})$, $\leq = \bigcup_{i \geq 0} (\leq_{\mathcal{E}_1^i} \cup \leq_{\mathcal{E}_2^i}) \cup \{(e, e') | e \in E_{\mathcal{E}_1^i}$ & $e' \in \bigcup_{j > i} (E_{\mathcal{E}_1^j} \cup E_{\mathcal{E}_2^j})$, with $i \geq j\}$, $\# = \bigcup_{i \geq 0} \#_{\mathcal{E}_2^i} \cup \{(e, e'), (e', e) | e \in E_{\mathcal{E}_2^i}$ & $e' \in E_{\mathcal{E}_1^i} \bigcup_{j > i} (E_{\mathcal{E}_1^j} \cup E_{\mathcal{E}_2^j})\}$, $l = \bigcup_{i \geq 0} (l_{\mathcal{E}_1^i} \cup l_{\mathcal{E}_2^i})$. Then $\mathcal{E} = (E, \leq, \#, l) = \mathcal{E}_{PBPA*}(p)$.

We denote $\widehat{C}(0) = \emptyset$, $\widehat{C}(n) = \widehat{C}(n-1) \cup E_{\mathcal{E}_1^n}$, for $n \geq 1$, and $[n, \mathcal{C}(\mathcal{E}_i)] = \widehat{C}(n) \cup \{C_j | C_j \in \mathcal{C}(\mathcal{E}_i^{n+1})\}$, for $i = 1, 2$. Then $\mathcal{C}(\mathcal{E}) = \bigcup_{i \geq 0} ([i, \mathcal{C}(\mathcal{E}_1)] \cup [i, \mathcal{C}(\mathcal{E}_2)])$. By the induction hypothesis for p_i and process structures $Pr(p_i) = (\mathcal{P}_i, \xrightarrow{a}_{p_i}, s_i, l_i)$, with $i = 1, 2$, there are isomorphisms $f_1 : \mathcal{C}(\mathcal{E}_{PBPA*}(p_1)) \to \mathcal{P}_1$ and $f_2 : \mathcal{C}(\mathcal{E}_{PBPA*}(p_2)) \to \mathcal{P}_2$. Let us consider the mappings $f_1^i : [i, \mathcal{C}(\mathcal{E}_1)] \to \mathcal{P}_1^i$ and $f_2^i : [i, \mathcal{C}(\mathcal{E}_2)] \to \mathcal{P}_2^i$ such

that $f_1^i(C) = f_1(C) \setminus \widehat{C}(i)$ and $f_2^i(C) = f_2(C) \setminus \widehat{C}(i)$, with $i \geq 0$. It is easy to see that f_1^i and f_2^i are isomorphisms for $i \geq 0$. Since $E_{\mathcal{E}_1} \cap E_{\mathcal{E}_2} = \emptyset$, $\mathcal{C}(\mathcal{E}_i^j) \cap \mathcal{C}(\mathcal{E}_k^l) = \{\emptyset\}$, if $i \neq k$ or $j \neq l$. Therefore, we get the following: $[i, \mathcal{C}(\mathcal{E}_1)] \cap [i, \mathcal{C}(\mathcal{E}_2)] = \widehat{C}(i)$,
$[i, \mathcal{C}(\mathcal{E}_1)] \cap [i+1, \mathcal{C}(\mathcal{E}_1)] = \widehat{C}(i+1) = [i, \mathcal{C}(\mathcal{E}_1)] \cap [i+1, \mathcal{C}(\mathcal{E}_2)]$ for $i \geq 0$;
$[i, \mathcal{C}(\mathcal{E}_2)] \cap [j, \mathcal{C}(\mathcal{E}_2)] = \emptyset$ for $i \neq j$;
$[i, \mathcal{C}(\mathcal{E}_1)] \cap [j, \mathcal{C}(\mathcal{E}_1)] = \emptyset$ for $i < j-1$ or $i > j+1$;
$[i, \mathcal{C}(\mathcal{E}_1)] \cap [j, \mathcal{C}(\mathcal{E}_2)] \neq \emptyset$ for $i \neq j$ and $i \neq j-1$.
We construct the mapping $f : \mathcal{C}(\mathcal{E}) \to \mathcal{P}$ as follows. We set $f(C) = f_i^j(C)$ if $C \in [j, \mathcal{C}(\mathcal{E}_i)]$ for $i = 1, 2$ and $j \geq 0$. According to $TR5$ and the construction of $\mathcal{E}$, we see that f is an isomorphism. □

Since $Pr(p) = (\mathcal{P}_p, \to_p, s_p, l_p)$ can be viewed as a transition system for each $p \in PBPA^*(Act)$, we can apply the bisimulation notion to it. So, we call two process structures $Pr(p)$ and $Pr(q)$ *bisimilar* (denoted by $Pr(p) \underline{\leftrightarrow} Pr(q)$) if there is a bisimulation $\mathcal{B}$ between $(\mathcal{P}_p, \to_p, s_p)$ and $(\mathcal{P}_q, \to_q, s_q)$ such that $s_p \mathcal{B} s_q$.

The following proposition shows that, although the procedure of constructing a process structure changes the structure of a transition system for a $PBPA^*$-term, the bisimulation between the process structures remains to be corresponding to the algebraic one.

Proposition 4.2. Let $p, q \in PBPA^*(Act)$. Then $Pr(p) \underline{\leftrightarrow} Pr(q) \Leftrightarrow p \underline{\leftrightarrow}\, q$.

Proof. $(\Rightarrow)$. It follows directly from Proposition 4.1.
$(\Leftarrow)$. Assume that $p \underline{\leftrightarrow}\, q$, and $\mathcal{R} \subseteq PBPA^*(Act) \times PBPA^*(Act)$ to be an algebraic bisimulation such that $p\mathcal{R}q$. We construct a relation $\mathcal{B} \subseteq \mathcal{P}_p \times \mathcal{P}_q$ as follows. For $v \in \mathcal{P}_p$ and $v' \in \mathcal{P}_q$, we set $(v, v') \in \mathcal{B}$ if $l_p(v) \mathcal{R} l_q(v')$. We have to show that $\mathcal{B}$ is a bisimulation.

Let $(v_1, v_1') \in \mathcal{B}$ and $v_1 \xrightarrow{a}_p v_2$. By Proposition 4.1, we have $l_p(v_1) \xrightarrow{a}_{PBPA^*} l_p(v_2)$. By construction of $\mathcal{B}$, we get $l_p(v_1) \mathcal{R} l_q(v_1')$. Then, by definition of the relation $\mathcal{R}$, we get $\exists r \in PBPA^*(Act) \diamond l_q(v_1') \xrightarrow{a}_{PBPA^*} r$. By definition of the transition relation in $PBPA^*(Act)$, we have that the transition $l_q(v_1') \xrightarrow{a}_{PBPA^*} r$ is obtained from one of the rules (Ax),(A1)-(A7) or (B1)-(B7), which depends on the structure of the term $l_q(v_1')$. Obviously, $q = l_q(s_q) = l_q(\overline{v}_1) \xrightarrow{a_1} \ldots \xrightarrow{a_{n-1}}_{PBPA^*} l_q(\overline{v}_n) = r$, where $l_q(\overline{v}_{(n-1)}) = l_q(v_1')$. Then, by construction of $Pr(q)$, we get $v_2' = \overline{v}_n$ and $v_1' \xrightarrow{a}_p v_2'$, where $v_1' = \overline{v}_{(n-1)}$.

The case $v_1' \xrightarrow{a}_q v_2'$ can be considered similarly to the previous one.
By construction of $\mathcal{B}$, it is obvious that $(s_p, s_q) \in \mathcal{B}$, since $l_p(s_p) = p$, $l_q(s_q) = q$, and $p\mathcal{R}q$. Therefore, $Pr(p) \underline{\leftrightarrow} Pr(q)$. □

Now we can establish the main result of the paper.

Theorem 4.2. Let $p, q \in PBPA^*(Act)$. Then $\mathcal{E}_{PBPA^*}(p) \approx_i \mathcal{E}_{PBPA^*}(q) \Leftrightarrow p \underline{\leftrightarrow} q$.

Proof. We have from Theorem 4.1. and Proposition 4.2, $\mathcal{E}_{PBPA^*}(p) \approx_i \mathcal{E}_{PBPA^*}(q) \Leftrightarrow Pr(p) \underline{\leftrightarrow} Pr(q)$, since the corresponding transition systems are isomorphic. Due to Proposition 4.2, this means that $\mathcal{E}_{PBPA^*}(p) \approx_i \mathcal{E}_{PBPA^*}(q) \Leftrightarrow p \underline{\leftrightarrow} q$. □

5. Conclusion

In this paper we have investigated an algebraic specification of a behavioural equivalence. We have introduced a process algebra with iteration and operations corresponding to all relations between events in a structure. By proposing the event structure semantics to algebraic terms, we have established the correspondence between algebraic and behavioural bisimulations. We have considered the process algebra BPA^* as a starting point, since it seems to be nice to specify a class of event structures with finite representations, which is needed for investigation of decidability of bisimulation notions defined over event structures. The decidability problem is an important question in the study of an equivalence notion. As an example, it is easy to notice that for the class of finite event structures bisimulations are decidable, and in the general case of infinite event structures it is obviously undecidable, whether two structures are bisimilar or not. The aim of our further research is to obtain nontrivial results for classes of event structures.

References

[1] N.A. Anisimov, *A Disabling of Event Structure*, PARLE'93, Lect. Notes Comput. Sci., **694**, 1993, 724–727.

[2] J.A. Bergstra, I. Bethke, A. Ponse, *Process algebra with iteration and nesting*, The Comput. J., **37**, No 4, 1994.

[3] G. Boudol, I. Castellani, *Concurrency and atomicity*, Theor. Comput. Sci., **59**, 1988, 25–84.

[4] R. van Glabbeek, U. Goltz, *Equivalence notions for concurrent systems and refinement of actions*, Lect. Notes Comput. Sci., **379**, 1989, 237–248.

[5] R. van Glabbeek, U. Goltz, *Equivalences and refinement*, Lect. Notes Comput. Sci., **469**, 1990, 309–333.

[6] A. Kucera, R. Mayr, *Weak bisimilarity with infinite-state systems can be decided in polynomial time*, Lect. Notes Comput. Sci., **1664**, 1999, 368–382.

[7] R. Loogen, U. Goltz, *Modelling Nondeterministic Concurrent Processes with Event Structures*, Fundamenta Informaticae, **14**, 1991, 39–73.

[8] M. Mukund, P.S. Thiagarajan, *An Axiomatization of Event Structures*, Foundations of Software Technology and Theoretical Computer Science, Lect. Notes Comput. Sci., **405**, 1989, 143–160.

[9] R. deNicola, U. Montanari, F. Vaandrager, *Back and Forth Bisimulations*, CONCUR'90, Lect. Notes Comput. Sci., **458**, 1990, 152–165.

[10] M. Nielsen, G. Plotkin, G. Winskel, *Petri nets, event structures and domains*, Theor. Comput. Sci., **13**, No 1, 1981, 85–108.

[11] I.B. Virbitskaite, A.V. Votintseva, E. Best, *Investigating Equivalence Notions for Nondetermenistic Processes*, Proc. of A.P. Ershov Second International Memorial Conference on Perspectives of System Informatics, Novosibirsk, June 25–28, 1996, 220–226.

[12] J. Winkowski, *Event structure representation of the behaviour of place/transition systems*, Fundamenta Informaticae, **11**, 1988, 405–432.

[13] G. Winskel, *Event structures*, Lect. Notes Comput. Sci., **255**, 1987, 325–392.

[14] D.M.R. Park, *Concurrency and automata on infinite sequences*, Proc. 5^{th} GI Conference, Lect. Notes Comput. Sci., **104**, 1981, 167–183.

[15] W.J. Fokkink, H. Zantema, *Basic Process Algebra with iteration: completeness of its equational axioms*, The Comput. J., **37**, No 4, 1994, 259–267.