Rule Formats for Non Interference*

Simone Tini

Dipartimento di Scienze CC.FF.MM., Università dell'Insubria, Via Valleggio 11, I-22100, Como, Italy
simone.tini@uninsubria.it

Abstract. We present the *SBSNNI rule format*. We prove that any Process Algebra construct whose SOS-style semantics is defined by SOS transition rules respecting such a format, preserves the well known non interference properties Persistent BNDC, SBSNNI, and SBNDC.

1 Introduction

One of the problems in computer security is the necessity to guarantee that only legitimate users can access some kind of information. To face this problem, one should take into account that malicious users could attempt to access information not only directly, but also indirectly through so called *covert channels*.

In *multilevel systems* [4], users are bound to several levels of security, and it must be guaranteed that users at any level cannot *interfere* with users at lower levels and cause different status of the system in which they operate to be perceived. This means that *information flow* from high levels to lower levels must be prevented. A drastic solution to this kind of problems is to avoid at all these possible interferences. A lot of *non interference* definitions have been proposed in the literature since [11], for several formal models of interaction between users. In most of these papers, for simplicity multilevel systems are represented by two level systems: Users are bound either to a *high level* of security, or to a *low level* of security. In [6,7,8,16,3,15] some of the non interference definitions given in the literature have been translated into the context of *Process Algebras*.

The most successful non interference definition in [6,7,8] is called *Bisimulation-based Non Deducibility on Compositions* (BNDC, for short). Intuitively, a system enforces BNDC if, by interacting with any possible high level user, the system always appears the same to low level users. Among the other non interference definitions in [6,7,8], we mention *Strong Bisimulation Strong Non-deterministic Non Interference* (SBSNNI, for short), which is stronger than BNDC, and *Strong BNDC* (SBNDC, for short), which, in turn, is stronger than SBSNNI. The mentioned properties are studied for systems specified by using the language of *Security Process Algebra* (SPA, for short), which is an extension of CCS [13] tailored to deal with two level systems. BNDC has been a successful non interference definition for systems lying in static contexts. In [9] it has been

* Research partially supported by Progetto Cofinanziato "Metodi Formali per la Sicurezza e il Tempo" (Mefisto)

P. Degano (Ed.): ESOP 2003, LNCS 2618, pp. 129–143, 2003.

shown that BNDC is too weak for systems running into a dynamic environment that can be reconfigured at run-time, or, equivalently, for systems that can migrate on the web during their computation. For this reason, the more restrictive non interference definition of *Persistent BNDC* (P_BNDC, for short) has been introduced. Intuitively, a system enforces P_BNDC if every state that can be reached by the system during its computation enforces BNDC. This means that even if the environment changes during the execution of the system, the security of the system is not compromised. P_BNDC is equivalent to SBSNNI, meaning that any system enforces P_BNDC if and only if it enforces SBSNNI (see [9]).

All the mentioned non interference properties are not, in general, *compositional*, meaning that there are constructs of SPA that do not preserve them. This is a critical issue, since one is not guaranteed that by putting a secure system into a SPA context, the obtained system is, in turn, secure. Another consequence of non compositionality is that the non interference properties cannot be checked compositionally with respect to the syntactic structure of systems [8,12].

In the present paper we argue that the non compositionality of the non interference properties depends on general *semantic properties* of SPA constructs. This implies that other Process Algebras having constructs with the same semantic properties suffer of the same problem. This is a typical situation in Process Algebras: A big amount of results depend on general semantic properties of the language constructs and do not depend on the particular language that is considered. An interesting challenge is to develop a meta theory for Process Algebras to study which semantic properties the constructs must have to preserve non interference properties. To this purpose, we recall that since the pioneering work [17], the concept of *rule format* has played a major rôle to develop meta theories for Process Algebras endowed with a *Structural Operational Semantics* [14] (SOS, for short). A rule format consists of a set of restrictions on the syntax of the SOS transition rules admitted. In particular, several rule formats have been proposed for ensuring that a given behavioral preorder (resp. equivalence) notion over processes is a precongruence (resp. congruence) (see [2] for a survey). Now, in the present paper we present the *SBSNNI rule format*, and we prove that any Process Algebra construct preserves both SBSNNI (and, therefore, P_BNDC) and SBNDC, provided that the operational semantics of such a construct is given by SOS transition rules respecting the SBSNNI format.

In Section 2 we recall SPA and the various non interference properties. In Section 3 we define our rule format. In Section 4 we prove that all constraints on SOS transition rules are needed. In Section 5 we prove that the format is correct for SBSNNI and SBNDC. Finally, in Section 6 we draw some conclusions.

2 Security Process Algebra

The *Security Process Algebra* (SPA) [6] models systems where the set *Act* of the actions that can be performed by each (sub)system is partitioned into a set of *visible input actions*, ranged over by $a, a_1, \ldots$, a set of *visible output actions*, ranged over by $\overline{a}, \overline{a_1}, \ldots$, and the *invisible action* τ, which models an internal

computation step that cannot be observed outside the system. A *complementation function* $(\text{-}) : Act \to Act$ is defined over actions such that $\overline{\overline{a}} = a$, for each $a \in Act \setminus \{\tau\}$, and $\overline{\tau} = \tau$. The intuition is that actions a and $\overline{a}$ performed by two processes running in parallel can synchronize, thus producing action τ.

To reflect two different levels of security, the set of (input and output) visible actions is partitioned into the set H of *high actions*, ranged over by $h, h_1, \ldots$, $\overline{h}, \overline{h_1}, \ldots$, and the set L of *low actions*, ranged over by $l, l_1, \ldots, \overline{l}, \overline{l_1}, \ldots$. Both sets H and L are closed under complementation.

The abstract syntax of SPA is given by the grammar below:

$$E ::= 0 \quad | \quad \mu \cdot E \quad | \quad E_1 + E_2 \quad | \quad E_1 | E_2 \quad | \quad E \setminus A \quad | \quad E[f]$$

where $E, E_1, \ldots$ are SPA process variables, μ is an action in Act, A is a set of actions in $Act \setminus \{\tau\}$ closed w.r.t. complementation, and $f : Act \to Act$ is a relabeling function over actions such that $f(\tau) = \tau$.

Process 0 does nothing. Process $\mu \cdot E$ performs action μ and then behaves as E. Process $E_1 + E_2$ can choose nondeterministically to behave like either E_1 or E_2. Process $E_1 | E_2$ is the parallel composition of E_1 and E_2, which interleave and can synchronize on complementary actions, thus producing action τ. Process $E \setminus A$ behaves as E, but it cannot perform actions in A. Finally, process $E[f]$ behaves as the process E where all actions are relabeled by function f. The SOS style semantics of SPA is given by the SOS transition rules in Table 1.

Table 1. The SOS transition rules for SPA

$$\frac{}{\mu \cdot E \xrightarrow{\mu} E} \qquad \frac{E_1 \xrightarrow{\mu} E_1'}{E_1 + E_2 \xrightarrow{\mu} E_1'} \qquad \frac{E_2 \xrightarrow{\mu} E_2'}{E_1 + E_2 \xrightarrow{\mu} E_2'}$$

$$\frac{E_1 \xrightarrow{\mu} E_1'}{E_1 | E_2 \xrightarrow{\mu} E_1' | E_2} \qquad \frac{E_2 \xrightarrow{\mu} E_2'}{E_1 | E_2 \xrightarrow{\mu} E_1 | E_2'} \qquad \frac{E_1 \xrightarrow{\mu} E_1' \; E_2 \xrightarrow{\overline{\mu}} E_2'}{E_1 | E_2 \xrightarrow{\tau} E_1' | E_2'} \; \mu \neq \tau$$

$$\frac{E \xrightarrow{\mu} E'}{E \setminus A \xrightarrow{\mu} E' \setminus A} \; \mu \notin A \qquad \frac{E \xrightarrow{\mu} E'}{E[f] \xrightarrow{f(\mu)} E'[f]}$$

As in [8], for any set of actions $A \subseteq Act$, we denote with E/A the process $E[f]$ such that $f(\mu) \equiv \begin{cases} \tau & \text{if } \mu \in A \\ \mu & \text{otherwise.} \end{cases}$

Moreover, we denote with $\mathcal{E}$ the set of all processes.

Let us recall the notion of weak bisimulation [13] over SPA processes. We need before some more notation.

Let $E \stackrel{\hat{\mu}}{\Longrightarrow} E'$ be either a shorthand for $E(\xrightarrow{\tau})^* E_1 \xrightarrow{\mu} E_2(\xrightarrow{\tau})^* E'$, if $\mu \in Act \setminus \{\tau\}$, or a shorthand for $E(\xrightarrow{\tau})^* E'$, if $\mu = \tau$. (As usual $(\xrightarrow{\tau})^*$ denotes a possibly empty sequence of τ transitions.)

Let $E \Longrightarrow E'$ denote that E' is *reachable* from E, i.e. either $E \stackrel{\hat{\tau}}{\Longrightarrow} E'$, or there is a sequence $\mu_1 \ldots \mu_n \in Act^*$ such that $E \stackrel{\hat{\mu_1}}{\Longrightarrow} \ldots \stackrel{\hat{\mu_n}}{\Longrightarrow} E'$.

Definition 1. *A relation $R \subseteq \mathcal{E} \times \mathcal{E}$ is a* weak bisimulation *if $(E,F) \in R$ implies, for all $\mu \in Act$,*

- *whenever $E \xrightarrow{\mu} E'$ for some process E', then there is a process F' such that $F \stackrel{\hat{\mu}}{\Longrightarrow} F'$ and $(E',F') \in R$*
- *whenever $F \xrightarrow{\mu} F'$ for some process F', then there is a process E' such that $E \stackrel{\hat{\mu}}{\Longrightarrow} E'$ and $(E',F') \in R$.*

Two SPA processes E, F are weakly bisimilar*, written $E \approx F$, iff there is a weak bisimulation containing the pair (E,F).*

Let us recall the notion of BNDC [6,7,8]. Let $\mathcal{E}_H$ denote the set of all SPA processes in $\mathcal{E}$ having only actions in $H \cup \{\tau\}$.

Definition 2. *A process E enforces the property of* Bisimulation-based Non Deducibility on Compositions*, written E is BNDC, iff*

$$\textit{for each process } F \in \mathcal{E}_H, \textit{ it holds that } (E|F) \setminus H \approx E/H$$

As explained in [6,7,8], E/H is what a low level observer can see of E, i.e. the part of E with which such an observer can synchronize. So, E is BNDC if, for each high level process F, a low level observer cannot distinguish E from $(E|F) \setminus H$, i.e. what the low level observer can see of E is not modified by composing any high level process F in parallel with E and by forcing synchronization on high actions between E and F.

In [9] it is shown that BNDC guarantees non interference only in static contexts. To guarantee non interference in completely dynamic hostile environments, the property of Persistent BNDC has been defined.

Definition 3. *A process E enforces the property of* Persistent BNDC*, written E is P_BNDC, iff*

$$\textit{for each process } E' \in \mathcal{E},\ E \Longrightarrow E' \textit{ implies that } E' \textit{ is BNDC}$$

P_BNDC requires that each state that is reachable from E is BNDC.

We recall also the property SBSNNI [6,7,8], which is equivalent to P_BNDC and does not require universal quantification over high level processes.

Definition 4. *A process E enforces the property of* Strong Bisimulation Strong Non-deterministic Non Interference*, written E is SBSNNI, iff*

$$\textit{for each process } E' \in \mathcal{E},\ E \Longrightarrow E' \textit{ implies that } E' \setminus H \approx E'/H$$

Finally, we recall the property SBNDC [6,7,8].

Definition 5. *A process E enforces the property of* Strong BNDC*, written E is SBNDC, iff*

$$\textit{for processes } E', E'' \in \mathcal{E},\ E \Longrightarrow E' \xrightarrow{h} E'' \textit{ implies that } E' \setminus H \approx E'' \setminus H$$

SBNDC requires that before and after each high action, the system appears to be the same, for a low level perspective.

The following results on non interference properties were proved in [6,9].

Proposition 1. *If a process is SBNDC then it is SBSNNI. A process is SBSNNI if and only if it is P_BNDC. If a process is SBSNNI then it is BNDC.*

Both SBSNNI and SBNDC are preserved by operators "|" and "\" (see [6]). Unfortunately, they are not preserved by operator "+", as it is shown below.

Example 1. Let $E \equiv h_1 \cdot l_1 \cdot 0 + l_1 \cdot 0$ and $F \equiv h_2 \cdot l_2 \cdot 0 + l_2 \cdot 0$. Both processes E and F are SBSNNI and SBNDC. Intuitively, in both processes, the high action guards a low action that can be performed also without performing the high action. The process $E + F$ is neither SBSNNI nor SBNDC. Intuitively, by performing the high action h_1, $E + F$ reaches a state in which it has no choice and it can perform only action l_1. Analogously, by performing the high action h_2, $E + F$ reaches a state in which it has no choice and it can perform only action l_2. Now, without performing any high action, $E + F$ is in a state in which it can choose between performing l_1 or l_2. So, such a state cannot be simulated by the two states reached by performing h_1 or h_2. Formally, the process E' reachable from E that violates conditions of Def. 4 is E itself. The processes E' and E'' that violate conditions of Def. 5 are E and the process reachable through h_1 (or that reachable through h_2), respectively.

3 The Format SBSNNI

In this section we present the format SBSNNI.

Let us return to Example 1. The reason for which process $E + F$ is neither SBSNNI nor SBNDC is that the high action h_1 of E forces $E + F$ to discard F (and, symmetrically, the high action h_2 of F forces $E + F$ to discard E).

We note that a quite similar reason implies another well know problem of operator +, i.e. that it does not preserve weak bisimulation (see [13]). In fact, notwithstanding $\tau \cdot a \cdot 0 \approx a \cdot 0$, it holds that $\tau \cdot a \cdot 0 + b \cdot 0 \not\approx a \cdot 0 + b \cdot 0$. Here the problem is that action τ of $\tau \cdot a \cdot 0 + b \cdot 0$ forces $\tau \cdot a \cdot 0 + b \cdot 0$ to discard $b \cdot 0$. To preserve weak bisimulation, operator + must be *patient*, meaning that, given any process $E + F$, the performance of some action τ by E (resp. F) should not imply discarding F (resp. E). To this purpose, as it has been observed in [5,18, 10], SOS transition rules of Table 1 for operator + must require that μ is not action τ, and, moreover, *patient rules* for operator + must be added as below:

$$\frac{E \xrightarrow{\tau} E'}{E + F \xrightarrow{\tau} E' + F} \qquad \frac{F \xrightarrow{\tau} F'}{E + F \xrightarrow{\tau} E + F'}$$

In order to preserve SBSNNI and SBNDC, operator + must have rules for high actions similar to the patient rules above.

Before introducing our format, we recall that, in general, the abstract syntax of a process algebra is given by a *signature* Σ, i.e. a set of *function* symbols with their *arities*. The algebra of (open) *terms* freely constructed over a set of variables `Var` (ranged over by $E, F, \ldots$) by applying function symbols in Σ is ranged over by t, s, r. Terms that do not contain variables are called *closed terms*, or *processes*, and are ranged over by p, q. A SOS *transition rule* (with only positive premises and without predicates) ρ has the form $\frac{H}{\alpha}$, where:

- H is a collection of *premises* of the form $t \xrightarrow{\mu} t'$
- α is a *conclusion* of the form $s \xrightarrow{\mu_1} s'$, where term s is called the *source* of ρ, term s' is called the *target* of ρ, and μ_1 is called the *action* of ρ.

Definition 6. *A Process Algebra having operator "·" of CCS and defined by SOS transition rules is* SBSNNI *if:*

1. *For each high action $h \in H$, the following transition rule is admitted:*

$$\frac{}{h \cdot E \xrightarrow{h} E}$$

2. *Transition rules ρ of the following form are admitted:*

$$\frac{\{E_i \xrightarrow{l_i} F_i \,|\, i \in I(\rho)\}}{f(E_1, \ldots, E_n) \xrightarrow{\mu} t}, \text{ where:}$$

   - $I(\rho) \subseteq \{1, \ldots, n\}$
   - $l_i \in L$ *for each* $i \in I(\rho)$, *and* $\mu \in L \cup \{\tau\}$
   - $E_1, F_1, \ldots, E_n, F_n$ *are the only variables occurring in* ρ, *and no variable* E_i *with* $i \in I(\rho)$ *occurs in the target* t
   - *no subterm* $h \cdot s$ *appears in* t, *for any* $h \in H$.

3. *Transition rules ρ of the following form are admitted:*

$$\frac{\{E_i \xrightarrow{h_i} F_i \,|\, i \in I(\rho)\}}{f(E_1, \ldots, E_n) \xrightarrow{\mu} f(F'_1, \ldots, F'_n)}, \text{ where:}$$

   - $I(\rho) \subseteq \{1, \ldots, n\}$ *and* $I(\rho) \neq \emptyset$
   - $h_i \in H$ *for each* $i \in I(\rho)$, *and* $\mu \in H \cup \{\tau\}$
   - *for each* $i \in \{1, \ldots, n\}$, $F'_i \equiv \begin{cases} F_i \text{ if } i \in I(\rho) \\ E_i \text{ otherwise.} \end{cases}$

4. *For all transition rules ρ, and all $i \in I(\rho)$, there is a* patient transition rule

$$\frac{E_i \xrightarrow{\tau} F_i}{f(E_1, \ldots, E_n) \xrightarrow{\tau} f(E_1, \ldots, E_{i-1}, F_i, E_{i+1}, \ldots, E_n)}$$

   and, moreover, for each action $h \in H$, there is a H-patient transition rule

$$\frac{E_i \xrightarrow{h} F_i}{f(E_1, \ldots, E_n) \xrightarrow{h} f(E_1, \ldots, E_{i-1}, F_i, E_{i+1}, \ldots, E_n)}$$

5. *No further transition rule is admitted.*

Notice that, on one hand, clause 1 above implies that high prefixing cannot preserve SBSNNI and SBNDC. On the other hand, clause 1 is reasonable and is needed to let processes perform high actions. So, we require that all operators except "·" preserve SBSNNI and SBNDC.

SPA becomes SBSNNI if we modify Table 1 as follows:

- in the transition rules for operator "+" we require that $\mu \notin H \cup \{\tau\}$, and we add the patience and H-patience transition rules for "+"
- in the transition rule for $E[f]$ we require that $f(h) \in H \cup \{\tau\}$, for each $h \in H$, and that $f(l) \in L \cup \{\tau\}$, for each $l \in L$, and we add the H-patience transition rules for $E[f]$
- in the transition rule for "$\backslash$" we require that $A \subseteq L$
- no modification for transition rules for operators "$\cdot$" and "$|$" is needed.

Let SPA$'$ be SPA with these modifications. One could ask whether Def. 2 is well defined for SPA$'$, since it considers process $(E|F)\backslash H$ and the operator $\backslash$ of SPA$'$ admits process $G \backslash A$ only if $A \subseteq L$. We have two (independent) explanations that this is not a contradiction. The first explanation is that the classic $\backslash$ used in Def. 2 is defined outside the format, and Def. 2 is valid also for languages in which the classic $\backslash$ is not defined. The idea is that, also for these languages, Def. 2 simply says that E is BNDC iff "what a low lever observer sees of E is not modified by composing any high level process F in parallel with E and by forcing synchronization on high actions between E and F", even if forcing synchronization on high actions is not admitted inside E. Here, classic $\backslash$ is simply a tool that is used to discover whether there is some information flow in systems (that are specified without such a tool). The second explanation is that we could consider SPA$'$ with the classic operator $\backslash$ and require that all operators except $\backslash$ and, obviously, $\cdot$ preserve non interference properties.

In the following, let us denote with $\oplus$ the operator $+$ with patience and H-patience transition rules, and with $+$ the classic operator defined in Table 1.

We conclude by observing that the formats in the literature that are closer to our format are *simply WB format* [5] and *de Simone format* [17]. Our format is more restrictive than simply WB format since simply WB does not distinguish between high and low actions and, therefore, it does not impose H-patience rules. As de Simone format, our format admits neither premises of the form $E \not\xrightarrow{\mu}$ (*negative premises*), nor variables appearing both in the left hand side of a premise and in the right hand side of another premise (*look ahead*), nor variables appearing in the left hand side of two premises (*double testing*), nor variables appearing both in the left hand side of a premise and in the target. Moreover, on one side, our format imposes H-patient rules, which are not considered by de Simone format, since it does not distinguish between high and low actions. On the other side, de Simone format does not admit variables to appear more than once in the target of transition rules, which is allowed by our format.

4 Necessity of Restrictions

In this section we show that all constraints of the SBSNNI format are needed. The necessity for having H-patience transition rules follows by Example 1.

First of all we show that SBSNNI format cannot admit transition rules where either high actions appear in premises and the action of the rule is low, or low actions appear in premises and the action of the rule is high.

Example 2. Let $p \equiv l_1 \cdot l_2 \cdot 0$. Process p is trivially SBSNNI and SBNDC. Let f be the function whose semantics is described by the following transition rules

$$\frac{E \xrightarrow{l_1} E'}{f(E) \xrightarrow{h} f(E')} \qquad \frac{E \xrightarrow{l_2} E'}{f(E) \xrightarrow{l_2} f(E')}$$

and by the patience and H-patience transition rules. Process $f(p)$ is isomorphic to $h \cdot l_2 \cdot 0$ and is neither SBSNNI nor SBNDC, since action h guards action l_2.

Let $p \equiv h_1 \cdot 0$ and $q \equiv h_2 \cdot 0$. Processes p and q are SBSNNI and SBNDC. Let f be the function whose semantics is described by the following transition rule

$$\frac{E \xrightarrow{h_1} E' \quad F \xrightarrow{h_2} F'}{f(E,F) \xrightarrow{l} f(E',F')}$$

and by the patience and H-patience transition rules. Process $f(p,q)$ is isomorphic to $h_1 \cdot h_2 \cdot 0 + h_2 \cdot h_1 \cdot 0 + l \cdot 0$, and it is neither SBSNNI nor SBNDC. In fact both actions h_1 and h_2 guard subprocesses that cannot perform the low action l, which can be performed in the initial state.

We show now that negative premises cannot be admitted in SBSNNI format.

Example 3. Let $p \equiv h \cdot l_1 \cdot \tau \cdot l_2 \cdot 0 \oplus l_1 \cdot l_2 \cdot 0$. Process p is isomorphic to $h \cdot (l_1 \cdot \tau \cdot l_2 \cdot 0 + l_1 \cdot l_2 \cdot 0) + l_1 \cdot l_2 \cdot 0$. It can be proved that p is SBSNNI and SBNDC. Intuitively, the reason is that the subprocess $l_1 \cdot \tau \cdot l_2 \cdot 0 + l_1 \cdot l_2 \cdot 0$ that is guarded by h is weakly bisimilar to the subprocess $l_1 \cdot l_2 \cdot 0$ that is not guarded by h. Let f, g be the functions whose semantics is described by the rules

$$\frac{E \xrightarrow{l_1} E'}{f(E) \xrightarrow{l_1} g(E')} \qquad \frac{E \xrightarrow{l_2} E'}{g(E) \xrightarrow{l_2} E'} \qquad \frac{E \not\xrightarrow{l_2}}{g(E) \xrightarrow{l_3} 0}$$

and by the patience and H-patience transition rules. Process $f(p)$ is neither SBSNNI nor SBNDC. In fact, $f(p)$ can perform l_3 only in the branch guarded by h. So, process E' violating conditions of Def. 4 is $f(p)$, and processes $E'$and E'' violating conditions of Def. 5 are $f(p)$ and that reachable from $f(p)$ through h. Note that the subprocess $l_1 \cdot \tau \cdot l_2 \cdot 0 + l_1 \cdot l_2 \cdot 0$ in p that is guarded by h is weakly bisimilar to the subprocess $l_1 \cdot l_2 \cdot 0$ that is not guarded by h since $\approx$ does not distinguish $l_1 \cdot \tau \cdot l_2 \cdot 0$ and $l_1 \cdot l_2 \cdot 0$. On the contrary, $f(l_1 \cdot \tau \cdot l_2 \cdot 0 + l_1 \cdot l_2 \cdot 0)$ and $f(l_1 \cdot l_2 \cdot 0)$ are not weakly bisimilar. In fact, the former process can perform l_1 and reach $g(\tau \cdot l_2 \cdot 0)$, whereas if the latter process performs l_1, it can reach only $g(l_2 \cdot 0)$. So, $\tau \cdot l_2 \cdot 0$ cannot perform l_2, and, therefore, $g(\tau \cdot l_2 \cdot 0)$ can perform l_3, whereas $l_2 \cdot 0$ can perform l_2, and, therefore, $g(l_2 \cdot 0)$ cannot perform l_3.

We show now that double testing cannot be admitted in SBSNNI format.

Example 4. Let $q \equiv ((l_1 \cdot l_3 \cdot 0 \oplus l_2 \cdot l_4 \cdot 0) \,|\, (\overline{l_1} \cdot 0 \oplus \overline{l_2} \cdot 0)) \setminus \{l_1, l_2, \overline{l_1}, \overline{l_2}\}$. Process q is isomorphic to $\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0$. Let $p \equiv ((h \cdot (l_3 \cdot 0 \oplus l_4 \cdot 0) \oplus l \cdot q) \,|\, \bar{l}) \setminus \{l, \bar{l}\}$. Process p is isomorphic to $h \cdot (l_3 \cdot 0 + l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)) + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$. It can be proved that p is SBSNNI and SBNDC. The reason is that the subprocess

$l_3 \cdot 0 + l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ that is guarded by h is weakly bisimilar to the subprocess $\tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ that is not guarded by h. Let f be the function whose semantics is described by the following transition rule

$$\frac{E \xrightarrow{l_3} E' \quad E \xrightarrow{l_4} E''}{f(E) \xrightarrow{l_5} 0}$$

and by patience and H-patience rules. Process $f(p)$ is neither SBSNNI nor SBNDC, since it can perform l_5 only in the branch guarded by h. As seen above, the subprocess $l_3 \cdot 0 + l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ in p guarded by h is weakly bisimilar to the subprocess $\tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ that is not guarded by h. On the contrary, $f(l_3 \cdot 0 + l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0))$ and $f(\tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0))$ are not weakly bisimilar. In fact, since $l_3 \cdot 0 + l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ can perform both l_3 and l_4, the former process performs l_5, whereas no subprocess reachable by $\tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ can perform both l_3 and l_4 and, therefore, the latter process cannot perform l_5.

We show now that look ahead cannot be admitted in SBSNNI format.

Example 5. Let $p \equiv h \cdot l_1 \cdot l_2 \cdot 0 \oplus l_1 \cdot \tau \cdot l_2 \cdot 0$. Process p is isomorphic to $h \cdot (l_1 \cdot l_2 \cdot 0 + l_1 \cdot \tau \cdot l_2 \cdot 0) + l_1 \cdot \tau \cdot l_2 \cdot 0$ and is SBSNNI and SBNDC. Intuitively, the reason is that the subprocess $l_1 \cdot l_2 \cdot 0 + l_1 \cdot \tau \cdot l_2 \cdot 0$ guarded by h is weakly bisimilar to the subprocess $l_1 \cdot \tau \cdot l_2 \cdot 0$ not guarded by h. Let f be the function whose semantics is described by the following transition rule

$$\frac{E \xrightarrow{l_1} E' \quad E' \xrightarrow{l_2} E''}{f(E) \xrightarrow{l_3} 0} \qquad \frac{E \xrightarrow{l} E'}{f(E) \xrightarrow{l} E'} \text{ for any } l \in L$$

and by patience and H-patience rules. The process $f(p)$ is neither SBSNNI nor SBNDC. In fact, $f(p)$ can perform l_3 only in the branch guarded by h. Note that the subprocess $l_1 \cdot l_2 \cdot 0 + l_1 \cdot \tau \cdot l_2 \cdot 0$ in p that is guarded by h is weakly bisimilar to the subprocess $l_1 \cdot \tau \cdot l_2 \cdot 0$ that is not guarded by h since $\approx$ does not distinguish between $l_1 \cdot l_2 \cdot 0$ and $l_1 \cdot \tau \cdot l_2 \cdot 0$. On the contrary, $f(l_1 \cdot l_2 \cdot 0 + l_1 \cdot \tau \cdot l_2 \cdot 0)$ and $f(l_1 \cdot \tau \cdot l_2 \cdot 0)$ are not weakly bisimilar. In fact, since $l_1 \cdot l_2 \cdot 0$ can perform action l_1 followed by l_2, the former process can perform l_3, whereas actions l_1 and l_2 in $l_1 \cdot \tau \cdot l_2 \cdot 0$ are separated by τ and, therefore, $f(l_1 \cdot \tau \cdot l_2 \cdot 0)$ cannot perform l_3.

Finally, we show that in SBSNNI format variables appearing in left hand side of premises cannot appear in the target of the transition rule.

Example 6. Let p be the SBSNNI and SBNDC process of Example 4. Let f be the function whose semantics is described by the following transition rule

$$\frac{E \xrightarrow{l} E'}{f(E) \xrightarrow{l} f(E)} \text{ for any } l \in L$$

and by patience and H-patience rules. The process $f(p)$ is neither SBSNNI nor SBNDC, since it can perform infinite sequences of actions l_3 and l_4 only in the branch guarded by h. As we have seen in Ex. 4, the subprocess $l_3 \cdot 0 +$

$l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ in p guarded by h is weakly bisimilar to the subprocess $\tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ that is not guarded by h. On the contrary, $f(l_3 \cdot 0 + l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0))$ and $f(\tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0))$ are not weakly bisimilar. In fact, since $l_3 \cdot 0 + l_4 \cdot 0 + \tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ can perform l_3 and l_4, the former process can perform l_3 and l_4 and can remain in the same state, i.e. it can perform an infinite sequence with both l_3 and l_4, whereas no subprocess reachable by $\tau \cdot (\tau \cdot l_3 \cdot 0 + \tau \cdot l_4 \cdot 0)$ can perform both l_3 and l_4 and, therefore, the latter process cannot perform an infinite sequence with both l_3 and l_4.

5 The Soundness of SBSNNI Format

In this section we prove that SBSNNI operators except high prefixing preserve SBSNNI and SBNDC. Since at first glance it could seem that SBSNNI and SBNDC coincide under the assumption of patience and H-patience rules, we show that this is not the case, thus requiring a proof for each of the two properties.

Example 7. For process $p \equiv h \cdot l \cdot 0$ and the function f such that

$$\frac{E \xrightarrow{\mu} E'}{f(E) \xrightarrow{\mu} f(E')} \text{ for any } \mu \in Act \qquad \frac{E \xrightarrow{h} E'}{f(E) \xrightarrow{\tau} f(E')} \qquad \frac{E \xrightarrow{h} E'}{f(E) \xrightarrow{l} f(E')},$$

$f(p)$ is isomorphic to $\tau \cdot l \cdot 0 + l \cdot l \cdot 0 + h \cdot l \cdot 0$ and it is SBSNNI but not SBNDC.

As usual, a context $C(t_1, \ldots, t_n)$ is a term where terms $t_1, \ldots, t_n$ can appear. For context $C(E_1, \ldots, E_n)$ and terms $s_1, \ldots, s_n$, $C[s_1, \ldots, s_n \setminus E_1, \ldots, E_n]$ is the term obtained by replacing in $C(E_1, \ldots, E_n)$ each variable E_i with s_i.

The second sentence of the theorem below implies that SBSNNI is preserved by operators defined by SBSNNI format.

Theorem 1. *Let R be the set of pairs*

$$(C[r_1, \ldots, r_k \setminus E_1, \ldots, E_k] \setminus H, C[r'_1, \ldots, r'_k \setminus E_1, \ldots, E_k]/H)$$

where $C(E_1, \ldots, E_k)$ is a context that does not contain any term $h \cdot s$ with $h \in H$, and, for each $1 \leq i \leq k$, r_i, r'_i are SBSNNI and $r_i \setminus H \approx r'_i/H$. It holds that:

- *The set R is a weak bisimulation.*
- *Terms $C[r_1, \ldots, r_k \setminus E_1, \ldots, E_k]$ and $C[r'_1, \ldots, r'_k \setminus E_1, \ldots, E_k]$ are SBSNNI.*

Proof. For readability, in this proof we write $E \stackrel{A}{\Longrightarrow} E'$, with $A \subseteq Act$, to denote that there is a sequence $E \stackrel{\hat{\mu_1}}{\Longrightarrow} \ldots \stackrel{\hat{\mu_n}}{\Longrightarrow} E'$ with $\mu_1, \ldots, \mu_n \in A$.

We prove by induction over the syntactic structure of context $C(E_1, \ldots, E_k)$ the first sentence of the thesis. The second sentence follows from the first one. In fact, each process $\hat{r}$ reachable from $C[r_1, \ldots, r_k \setminus E_1, \ldots, E_k]$ has the form $C'[\hat{r}_1, \ldots, \hat{r}_k \setminus E_1, \ldots, E_k]$, for some context $C'(E_1, \ldots, E_k)$ that does not contain any subterm $h \cdot q$ with $h \in H$ and for some terms $\hat{r}_1, \ldots, \hat{r}_k$ that are reachable from $r_1, \ldots, r_k$, respectively (this fact can be immediately proved by induction over the number of transitions needed to reach $\hat{r}$). Now, since $\hat{r}_i$ is reachable from r_i and since r_i is SBSNNI, it holds that also $\hat{r}_i$ is SBSNNI, and, therefore, $\hat{r}_i \setminus H \approx$

$\hat{r}_i/H$. So, we can consider the first sentence of the thesis and we can instantiate, for each $1 \leq i \leq k$, r_i and r'_i with the same SBSNNI term $\hat{r}_i$, thus obtaining that $\hat{r} \setminus H \equiv C'[\hat{r}_1, \ldots, \hat{r}_k \setminus E_1, \ldots, E_k] \setminus H \approx C'[\hat{r}_1, \ldots, \hat{r}_k \setminus E_1, \ldots, E_k]/H \equiv \hat{r}/H$. So, since each term $\hat{r}$ reachable from $C[r_1, \ldots, r_k \setminus E_1, \ldots, E_k]$ satisfies $\hat{r} \setminus H \approx \hat{r}/H$, it holds that $C[r_1, \ldots, r_k \setminus E_1, \ldots, E_k]$ is SBSNNI. Analogously, we can prove that $C[r'_1, \ldots, r'_k \setminus E_1, \ldots, E_k]$ is SBSNNI.

So, let us prove by induction the first sentence of the thesis.

The base case $C(E_1, \ldots, E_n) \equiv c$ for a constant c is immediate, since clauses of SBSNNI format imply that each process reachable from c is a constant and that constants cannot perform high actions, thus ensuring that $c \setminus H \approx c/H$.

Also the base case $C(E_1, \ldots, E_k) \equiv E_i$ is immediate, since $E_i[r_1, \ldots, r_k \setminus E_1, \ldots, E_k] \equiv r_i$, $E_i[r'_1, \ldots, r'_k \setminus E_1, \ldots, E_k] \equiv r'_i$, and $r_i \setminus H \approx r'_i/H$ by the hypothesis.

As regards the inductive step, we assume the thesis for $C_1(E_1, \ldots, E_k)$, …, $C_n(E_1, \ldots, E_k)$, and we prove it for $f(C_1(E_1, \ldots, E_k), \ldots, C_n(E_1, \ldots, E_k))$. To this purpose, for each $1 \leq i \leq n$, let us denote with t_i the term $C_i[r_1, \ldots, r_k \setminus E_1, \ldots, E_k]$, and with s_i the term $C_i[r'_1, \ldots, r'_k \setminus E_1, \ldots, E_k]$. We must prove that $f(t_1, \ldots, t_n) \setminus H \approx f(s_1, \ldots, s_n)/H$ follows from $t_i \setminus H \approx s_i/H$, for $1 \leq i \leq n$.

It suffices to prove the following properties:

1. $f(t_1, \ldots, t_n) \setminus H \xrightarrow{\mu} t$ implies $f(s_1, \ldots, s_n)/H \xrightarrow{\hat{\mu}} s$, for some term s such that $(t, s) \in R$
2. $f(s_1, \ldots, s_n)/H \xrightarrow{\mu} s$ implies $f(t_1, \ldots, t_n) \setminus H \stackrel{\hat{\mu}}{\Longrightarrow} t$, for some term t such that $(t, s) \in R$.

We should prove both properties, since the proofs are not perfectly symmetric, but for lack of space we prove only the first.

Let us assume that $f(t_1, \ldots, t_n) \setminus H \xrightarrow{\mu} t$. We have one of the following three cases:

1. Transition $f(t_1, \ldots, t_n) \setminus H \xrightarrow{\mu} t$ is inferred by means of the following proof:

$$\dfrac{\dfrac{\{t_i \xrightarrow{l_i} t'_i \mid i \in I(\rho)\}}{f(t_1, \ldots, t_n) \xrightarrow{\mu} G(\hat{t}_1, \ldots, \hat{t}_n)}}{f(t_1, \ldots, t_n) \setminus H \xrightarrow{\mu} G(\hat{t}_1, \ldots, \hat{t}_n) \setminus H}$$

where $l_i \in L$ for each $i \in I(\rho)$, $\mu \in L \cup \{\tau\}$, $t \equiv G(\hat{t}_1, \ldots, \hat{t}_n) \setminus H$ and $\hat{t}_i \equiv \begin{cases} t'_i \text{ if } i \in I(\rho) \\ t_i \text{ otherwise.} \end{cases}$ For each index $i \in I(\rho)$, $t_i \xrightarrow{l_i} t'_i$ with $l_i \in L$ implies $t_i \setminus H \xrightarrow{l_i} t'_i \setminus H$, which, in turn, implies that there is a term s'_i such that $s_i/H \stackrel{\hat{l_i}}{\Longrightarrow} s'_i/H$ and $t'_i \setminus H \approx s'_i/H$. Therefore, there are terms s''_i and s'''_i such that $s_i \stackrel{H \cup \{\tau\}}{\Longrightarrow} s''_i \xrightarrow{l_i} s'''_i \stackrel{H \cup \{\tau\}}{\Longrightarrow} s'_i$. Now, by patience and H-patience rules we obtain that

$$\frac{\{s_i \stackrel{H\cup\{\tau\}}{\Longrightarrow} s_i'' \mid i \in I(\rho)\}}{\dfrac{f(s_1,\ldots,s_n) \stackrel{H\cup\{\tau\}}{\Longrightarrow} f(\hat{s}_1'',\ldots,\hat{s}_n'')}{f(s_1,\ldots,s_n)/H \stackrel{\hat{\tau}}{\Longrightarrow} f(\hat{s}_1'',\ldots,\hat{s}_n'')/H}}$$

where $\hat{s}_i'' \equiv \begin{cases} s_i'' \text{ if } i \in I(\rho) \\ s_i \text{ otherwise.} \end{cases}$ Now, it holds that

$$\frac{\{s_i'' \stackrel{l_i}{\longrightarrow} s_i''' \mid i \in I(\rho)\}}{\dfrac{f(\hat{s}_1'',\ldots,\hat{s}_n'') \stackrel{\mu}{\longrightarrow} G(\hat{s}_1''',\ldots,\hat{s}_n''')}{f(\hat{s}_1'',\ldots,\hat{s}_n'')/H \stackrel{\mu}{\longrightarrow} G(\hat{s}_1''',\ldots,\hat{s}_n''')/H}}$$

where $\hat{s}_i''' \equiv \begin{cases} s_i''' \text{ if } i \in I(\rho) \\ s_i \text{ otherwise.} \end{cases}$ Finally, by patience and H-patience rules we obtain

$$\frac{\{s_i''' \stackrel{H\cup\{\tau\}}{\Longrightarrow} s_i' \mid i \in I(\rho)\}}{\dfrac{G(\hat{s}_1''',\ldots,\hat{s}_n''') \stackrel{H\cup\{\tau\}}{\Longrightarrow} G(\hat{s}_1',\ldots,\hat{s}_n')}{G(\hat{s}_1''',\ldots,\hat{s}_n''')/H \stackrel{\hat{\tau}}{\Longrightarrow} G(\hat{s}_1',\ldots,\hat{s}_n')/H}}$$

where $\hat{s}_i' \equiv \begin{cases} s_i' \text{ if } i \in I(\rho) \\ s_i \text{ otherwise.} \end{cases}$ Summarizing, it holds that $f(s_1,\ldots,s_n)/H \stackrel{\hat{\mu}}{\Longrightarrow} G(\hat{s}_1',\ldots,\hat{s}_n')/H$. The term $G(\hat{s}_1',\ldots,\hat{s}_n')/H$ is the term s we were looking for. In fact, $(G(\hat{t_1},\ldots,\hat{t_n}) \setminus H, G(\hat{s}_1',\ldots,\hat{s}_n')/H)$ is a pair in R, since, for each $1 \leq i \leq n$, $\hat{t}_i$ and $\hat{s}_i'$ are reachable from t_i and s_i, respectively, and are SBSNNI, and since, for each $i \in I(\rho)$, it holds that $\hat{t}_i \setminus H \equiv t_i' \setminus H \approx s_i'/H \equiv \hat{s}_i'/H$, and, for each $i \notin I(\rho)$, it holds that $\hat{t}_i \setminus H \equiv t_i \setminus H \approx s_i/H \equiv \hat{s}_i'/H$.

2. Transition $f(t_1,\ldots,t_n) \setminus H \stackrel{\mu}{\longrightarrow} t$ is inferred by means of the following proof:

$$\frac{\{t_i \stackrel{h_i}{\longrightarrow} t_i' \mid i \in I(\rho)\}}{\dfrac{f(t_1,\ldots,t_n) \stackrel{\tau}{\longrightarrow} f(\hat{t}_1,\ldots,\hat{t}_n)}{f(t_1,\ldots,t_n) \setminus H \stackrel{\tau}{\longrightarrow} f(\hat{t}_1,\ldots,\hat{t}_n) \setminus H}}$$

where $h_i \in H$ for each $i \in I(\rho)$, $\mu = \tau$, $t \equiv f(\hat{t}_1,\ldots,\hat{t}_n) \setminus H$, and $\hat{t}_i \equiv \begin{cases} t_i' \text{ if } i \in I(\rho) \\ t_i \text{ otherwise.} \end{cases}$ For each $i \in I(\rho)$, $t_i \stackrel{h_i}{\longrightarrow} t_i'$ with $h_i \in H$ implies $t_i/H \stackrel{\tau}{\longrightarrow} t_i'/H$. Since t_i is SBSNNI, this last fact implies that $t_i \setminus H \stackrel{\hat{\tau}}{\Longrightarrow} t_i'' \setminus H$ for some term t_i'' such that $t_i'' \setminus H \approx t_i'/H$. It follows that there is a term s_i' such that $s_i/H \stackrel{\hat{\tau}}{\Longrightarrow} s_i'/H$ and $t_i'' \setminus H \approx s_i'/H$. Now, $s_i/H \stackrel{\hat{\tau}}{\Longrightarrow} s_i'/H$ is due to a sequence of transitions $s_i \stackrel{H\cup\{\tau\}}{\Longrightarrow} s_i'$. By patience and H-patience rules we obtain that

$$\frac{\{s_i \stackrel{H\cup\{\tau\}}{\Longrightarrow} s_i' \mid i \in I(\rho)\}}{\dfrac{f(s_1,\ldots,s_n) \stackrel{H\cup\{\tau\}}{\Longrightarrow} f(\hat{s}_1,\ldots,\hat{s}_n)}{f(s_1,\ldots,s_n)/H \stackrel{\hat{\tau}}{\Longrightarrow} f(\hat{s}_1,\ldots,\hat{s}_n)/H}}$$

where $\hat{s}_i \equiv \begin{cases} s'_i \text{ if } i \in I(\rho) \\ s_i \text{ otherwise.} \end{cases}$ Term $f(\hat{s}_1, \ldots, \hat{s}_n)/H$ is the term s we were looking for. In fact, $(f(\hat{t}_1, \ldots, \hat{t}_n) \setminus H, f(\hat{s}_1, \ldots, \hat{s}_n)/H)$ is a pair in R, since, for each $1 \leq i \leq n$, $\hat{t}_i$ and $\hat{s}_i$ are reachable from t_i and s_i, respectively, and are SBSNNI, and since, for each $i \in I(\rho)$, it holds that $\hat{t}_i \setminus H \equiv t'_i \setminus H \approx$ (since t'_i is reachable from t'_i and is SBSNNI) $t'_i/H \approx t''_i \setminus H \approx s'_i/H \equiv \hat{s}_i/H$, and, for each $i \notin I(\rho)$, it holds that $\hat{t}_i \setminus H \equiv t_i \setminus H \approx s_i/H \equiv \hat{s}_i/H$.

3. Transition $f(t_1, \ldots, t_n) \setminus H \xrightarrow{\mu} t$ is inferred by means of the following proof:

$$\dfrac{\dfrac{t_i \xrightarrow{\tau} t'_i}{f(t_1, \ldots, t_n) \xrightarrow{\tau} f(t_1, \ldots, t_{i-1}, t'_i, t_{i+1}, \ldots, t_n)}}{f(t_1, \ldots, t_n) \setminus H \xrightarrow{\tau} f(t_1, \ldots, t_{i-1}, t'_i, t_{i+1}, \ldots, t_n) \setminus H}$$

where $\mu = \tau$ and $t \equiv f(t_1, \ldots, t_{i-1}, t'_i, t_{i+1}, \ldots, t_n) \setminus H$. Since $t_i \xrightarrow{\tau} t'_i$, it holds that $t_i \setminus H \xrightarrow{\tau} t'_i \setminus H$, which implies that there is some term s'_i such that $s_i/H \stackrel{\hat{\tau}}{\Longrightarrow} s'_i/H$ and $t'_i \setminus H \approx s'_i/H$. The sequence of transitions $s_i/H \stackrel{\hat{\tau}}{\Longrightarrow} s'_i/H$ is inferred by a sequence $s_i \stackrel{H \cup \{\tau\}}{\Longrightarrow} s'_i$. By patience and H-patience rules we obtain

$$\dfrac{\dfrac{s_i \stackrel{H \cup \{\tau\}}{\Longrightarrow} s'_i}{f(s_1, \ldots, s_n) \stackrel{H \cup \{\tau\}}{\Longrightarrow} f(s_1, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n)}}{f(s_1, \ldots, s_n)/H \stackrel{\hat{\tau}}{\Longrightarrow} f(s_1, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n)/H}$$

The term $f(s_1, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n)/H$ is the term s we were looking for. In fact, the pair $(f(t_1, \ldots, t_{i-1}, t'_i, t_{i+1}, \ldots, t_n) \setminus H, f(s_1, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n)/H)$ is in R, since $t'_i \setminus H \approx s'_i/H$, t'_i and s'_i are reachable from t_i and s_i, respectively, and are SBSNNI, and, for each $j \neq i$, $t_j \setminus H \approx s_j/H$. □

The second sentence of the theorem below implies that SBNDC is preserved by operators defined by SBSNNI format.

Theorem 2. *Let R be the set of pairs*

$$(C[r_1, \ldots, r_k \setminus E_1, \ldots, E_k] \setminus H, C[r'_1, \ldots, r'_k \setminus E_1, \ldots, E_k] \setminus H)$$

where $C(E_1, \ldots, E_k)$ is a context that does not contain any term $h \cdot s$ with $h \in H$, and, for each $1 \leq i \leq k$, r_i, r'_i are SBNDC and $r_i \setminus H \approx r'_i \setminus H$. It holds that:

- *The set R is a weak bisimulation.*
- *Terms $C[r_1, \ldots, r_k \setminus E_1, \ldots, E_k]$, and $C[r'_1, \ldots, r'_k \setminus E_1, \ldots, E_k]$ are SBNDC.*

6 Conclusions

We have presented the SBSNNI format. It guarantees that all operators, except high prefixing, preserve SBSNNI and SBNDC [6,7,8], which are successful non

interference properties for systems running into dynamic environments (systems migrating on the network). Compositionality of non interference properties is useful since by composing secure (according to the property chosen) processes, one obtains secure processes. Moreover, compositionality can be exploited also to check non interference inductively with respect to the structure of the system.

We have compared our format with those in the literature. We have shown by some examples that all the restrictions imposed by the format are needed.

Our next aim is to extend our results by proposing formats for other non interference properties. We shall consider BNDC [6,7,8], which is a successful property for systems running into static environments, and the properties defined in [3,15,16]. Finally, we aim to understand what addition to our format is needed to have compositionality also w.r.t. high prefixing. Our starting point is that it seems natural to think that if E is secure, then $h \cdot E + \tau \cdot E$ is also secure, i.e. that high prefixing could be admitted provided that a duplicate of its derivative can be reached also through a silent action.

References

1. L. Aceto and W.J. Fokkink, editors, Special issue on process algebra, Information Processing Letters, 80, 2001.
2. L. Aceto, W.J. Fokkink, and C. Verhoef, Structural operational semantics, in J.A. Bergstra, A. Ponse, and S.A. Smolka, editors, Handbook of Process Algebra, Elsevier, Amsterdam, 2001, 197–292.
3. P.G. Allen, A comparison of non interference and non deducibility using CSP, Proc. IEEE Computer Security Foundation Workshop, IEEE Computer Society Press, 1991, 43–54.
4. D. Bell and L.J. La Padula, Secure computer systems: Unified exposition and multics interpretation, Technical report ESD-TR-75-301, MITRE MTR-2997, 1976.
5. B. Bloom, Structural operational semantics for weak bisimulation, Theoretical Computer Science, 146, 1995, 25–68.
6. R. Focardi and R. Gorrieri, A classification of security properties for process algebras, Journal of Computer Security, 3, 1995, 5–33.
7. R. Focardi and R. Gorrieri, The compositional security checker: A tool for the verification of information flow security properties, IEEE Transactions on Software Engineering, 23, 1997, 550–571.
8. R. Focardi and R. Gorrieri, Classification of security properties (Part I: Information flow), Foundations of Security Analysis and Design, Tutorial Lectures, Lecture Notes in Computer Science, 2171, Springer, Berlin, 2001, 331–396.
9. R. Focardi and S. Rossi, Information flow security in dynamic contexts, Proc. IEEE Computer Security Foundation Workshop, IEEE Computer Society Press, 2002, 307–319.
10. W.J. Fokkink, Rooted branching bisimulation as a congruence, Journal of Computer and System Sciences, 60, 2000, 13–37.
11. J.A. Goguen and J. Meseguer, Security policy and security models, Proc. IEEE Symposium on Security and Privacy, IEEE Computer Society Press, 1982, 11–20.
12. F. Martinelli, Partial model checking and theorem proving for ensuring security properties, Proc. IEEE Computer Security Foundation Workshop, IEEE Computer Society Press, 1998, 44–52.

13. R. Milner, Communication and concurrency, Prentice-Hall, London, 1989.
14. G. Plotkin, A structural approach to operational semantics, Technical report DAIMI FN-19, University of Aarhus, Denmark, 1981.
15. A.W. Roscoe, J.C.P. Woodcock, and L. Wulf, Non interference through determinism, Proc. European Symposium on Research in Computer Security, Lecture Notes in Computer Science, 875, Springer, Berlin, 1994, 33–53.
16. P.Y.A. Ryan, A CSP formulation of non-interference, Proc. IEEE Computer Security Foundation Workshop, IEEE Computer Society Press, 1990.
17. R. De Simone, Higher level synchronization devices in SCCS-Meije, Theoretical Computer Science, 37, 1985, 245–267.
18. I. Ulidowski and I. Phillips, Formats of ordered SOS rules with silent actions, Proc. Theory and Practice of Software Development, Lecture Notes in Computer Science, 1214, Springer, Berlin, 1997, 297–308.