

Operational semantics for Petri net components

Citation for published version (APA):

Groote, J. F., & Voorhoeve, M. (2003). *Operational semantics for Petri net components*. (Computer science reports; Vol. 0308). Technische Universiteit Eindhoven.

Document status and date:

Published: 01/01/2003

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Operational semantics for Petri Net components

Jan Friso Groote and Marc Voorhoeve

Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, the Netherlands
J.F.Groote@tue.nl M.Voorhoeve@tue.nl

Abstract

We develop a theory for net components with labeled interface places and transitions. Nets are shown to be isomorphic to algebraic terms, with marked places and transitions as atoms and arc addition, fusion and relabeling as operators. Net terms with the step firing rule are given a Plotkin-style SOS semantics, yielding compositionality of the operators. Some rules for reducing nets modulo strong and branching bisimilarity are given.

1 Introduction

In engineering, all kinds of models are created related to the artifacts to be built. In Software Engineering, a marked division exists between graphical (e.g. UML) and character-based (algebraic) approaches. Graphical models are used mostly in the initial requirements phases of a project and have gained wide acceptance, despite their often shaky semantics. Theorists are busy plugging holes, like in [14], but this seems a slow process which necessarily restricts the use of the modeling approach. Algebraic models are often used in the later design phases. Due to their formal semantics, they allow various kinds of manipulations and consistency checks, although they are harder to understand and discuss with end users.

It is claimed that Petri net based modeling languages, like Design/CPN [10] and ExSpect [8] combine an intuitive, graphical approach with mathematical rigor. This allows the use of the same modeling paradigm throughout the artifact's construction. However, the hierarchy concept supported by these languages presents some semantic problems. It is certainly possible to define the semantics of a marked net, e.g. by indicating enabled transitions and successor states, but this definition neglects the fact that nets modeling systems are “open” to interactions by an environment.

The approach usually taken is to “close” such nets by embedding it into some standard environment (context) that can exhibit any allowed interaction. However, one cannot be sure whether the behavior of the embedded net is indicative for its behavior in another context. Suppose two closed nets behave the same in their standard context. It must be proved that they then behave the same in any context.

This has been done in [16], featuring a definition for open nets and several operators to combine open subnets. The semantics of such an open net is defined by embedding it in a “universal context”. It is proved that open nets with the same semantics behave the same w.r.t. the operators defined.

In this paper we take their approach a step further. We give a structured operational semantics for open nets as algebraic terms, without any context. As in [16], we have unconnected places and transitions as atoms, the merge as binary operator and several unary operators: consumption/production (arc addition), node relabeling (hiding) and place/transition fusion (melding). A difference between our approach and [16] is the labeling of nodes; all nodes that are not labeled with the “internal” label ι are external nodes that can be interfaced with. Different external nodes can have the same label, causing nondeterminism. This complicates the definition of the unary operators, but in return operators are unconditional: any operator applied to any net yields some result net.

By labeling both places and transitions, our semantics respects the dual (place vs. transition) nature of Petri nets. By adopting a temporal language like HML [9], it is possible to express and verify properties of open nets that are both state and event based, like “if a request has arrived which has not been answered yet, place p is marked” and “if place p is marked, an answer will be issued”. Net-based modeling does benefit from the ability to reason about both states and events [12].

Our paper is structured as follows. After a preliminary section, we define labeled nets, net operators and terms. We then give a semantics for nets and show the equivalence of this semantics for net terms with another semantics defined by SOS rules [15]. In a next section, we explore the equivalence notions between nets that follow from our semantics and show some simple rules for reducing nets while staying within the same equivalence class. We terminate with a brief discussion and a comparison with related approaches.

2 Preliminaries

2.1 Relations and tuples

A relation between sets A and B is a subset of $A \times B$. Special subsets of the powerset $\mathbb{P}(A \times B)$ are the sets of functions ($A \rightarrow B$). If f is a function we write $y = f(x)$ iff $x R y$. If f is a function and X a set, then $f \upharpoonright X = \{(x, f(x)) \mid x \in \text{dom}(f) \cap X\}$. The composition $;$ of two relations is defined by $x (R ; S) z \Leftrightarrow \exists y : x R y \wedge y S z$. The composition $f \circ g$ of two functions f, g is the function defined by $g ; f$, so $(f \circ g)(x) = f(g(x))$. The k -iteration of a relation R is defined by $R^1 = R$, $R^{k+1} = R^k ; R$ and $R^+ = \bigcup_{k>0} R^k$. The inverse of a relation R is defined by $x R^{-1} y \Leftrightarrow y R x$. The identity functions are defined by $\text{id}(A) = \{(a, a) \mid a \in A\}$.

2.2 Bags

Let A be a set. The set $\mathbb{B}(A)$ of finite A -bags is the set of terms $n_1[a_1] + \dots + n_k[a_k]$ with the $a_i \in A$ and $n_i \in \mathbb{N}^+$. We denote the empty bag by $[\]$ and write $[a] = 1[a]$, $[ab] = [a] + [b]$, $[aa] = [a^2] = 2[a]$ and so on. We define addition and multiplication by integers $n \in \mathbb{N}$ in the obvious way. We set $\alpha \leq \beta$ iff $\exists \gamma : \alpha + \gamma = \beta$, $\beta \geq \alpha$ iff $\alpha \leq \beta$ and $\beta - \gamma = \alpha$ iff $\alpha + \gamma = \beta$. If α, β are bags, then their quotient $\alpha \div \beta$ is defined as the largest $n \in \mathbb{N}$ such that $n\beta \leq \alpha$. Application of a bag to an element is defined by $[\](a) = 0$, $a = 1$, $[b](a) = 0$ for $b \neq a$ and $(\alpha + \beta)(a) = \alpha(a) + \beta(a)$.

Definition 2.1. A relation $R \subseteq \mathbb{B}(A) \times \mathbb{B}(B)$ is called *additively closed* iff $[\] R [\]$ and $\forall x, y, z, w : x R y \wedge z R w \Rightarrow (x + z) R (y + w)$. The *additive closure* of a relation $R \subseteq \mathbb{B}(A) \times \mathbb{B}(B)$ is the smallest additively closed S such that $R \subseteq S$. If $R \subseteq A \times \mathbb{B}(B)$, then $\tilde{R} \subseteq \mathbb{B}(A) \times \mathbb{B}(B)$ is the additive closure of $\{([a], \beta) \mid a R \beta\}$. If $R \subseteq A \times B$, then $\bar{R} \subseteq \mathbb{B}(A) \times \mathbb{B}(B)$ is the additive closure of $\{([a], [b]) \mid a R b\}$.

If $f \in A \rightarrow B$ is a function, then \bar{f} is a function too, satisfying $\bar{f}(\Sigma_i[x_i]) = \Sigma_i[f(x_i)]$.

2.3 Transition systems

We presuppose a set \mathcal{E} of events and a special *silent* event $\tau \in \mathcal{E}$. A *transition system* is a pair (S, \longrightarrow) , where S is set of states and $\longrightarrow \in \mathbb{P}(S \times \mathcal{E} \times S)$ a ternary transition relation. We write $x \xrightarrow{e} x'$ iff $(x, e, x') \in \longrightarrow$. The interpretation of $x \xrightarrow{e} x'$ is that the system can move from state x to state x'

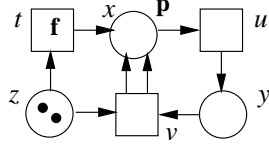


Figure 1: Example MLN

by event e . The set $\mathcal{R}(X)$ of states *reachable* from X is defined as the smallest set containing X such that for any $e \in \mathcal{E}$; $Y \in \mathcal{R}(X)$; $Y' \in S$ with $Y \xrightarrow{e} Y'$ we have $Y' \in \mathcal{R}(X)$.

We now define an equivalence relation between states of a transition system. Bisimilarity abstracts from states; branching bisimilarity also abstracts from silent events.

Definition 2.2. Let (S, \longrightarrow) be a transition system. A relation $R \in \mathbb{P}(S \times S)$ is called a *simulation* iff for all $x, y, x' \in S$; $e \in \mathcal{E}$ we have

$$x R y \wedge x \xrightarrow{e} x' \Rightarrow (\exists y' \in S : y \xrightarrow{e} y' \wedge x' R y').$$

A simulation R is a *bisimulation* iff R^{-1} is a simulation too. The states $X, Y \in S$ are called *bisimilar* (notation $X \sim Y$) iff there exists a bisimulation R such that $X R Y$.

The definition of branching bisimilarity requires an auxiliary notion [3]. We define $X \xrightarrow{(e)} X'$ iff $X \xrightarrow{e} X' \vee (e = \tau \wedge X = X')$. Also $X \Longrightarrow X'$ iff $X \xrightarrow{(\tau)^+} X'$.

Definition 2.3. Let S, \longrightarrow be a transition system. A relation $R \in \mathbb{P}(S \times S)$ is called an η -*simulation* iff for all $x, y, x' \in S_X$; $e \in \mathcal{E}$ we have

$$x R y \wedge x \xrightarrow{e} x' \Rightarrow \exists y'', y' : y \Longrightarrow y'' \xrightarrow{(e)} y' \wedge x R y'' \wedge x' R y'.$$

An η -simulation R is called a *branching bisimulation* iff R^{-1} is an η -simulation too. The states $X, Y \in S$ are called *branching bisimilar* (notation $X \sim_b Y$) iff there exists a branching bisimulation R such that $X R Y$.

Bisimulation is also called *strong* bisimulation. A strong bisimulation is also a branching bisimulation. The bisimilarity notions are equivalence relations on states.

3 Labeled nets

In labeled nets, nodes (both places and transitions) are labeled. A special label ι indicates *internal* nodes. The nodes with a different label are external, and constitute the net's *interface*. We assume a countably infinite universe \mathcal{N} of node labels with $\iota \in \mathcal{N}$ and set $\mathcal{N}^e = \mathcal{N} \setminus \{\iota\}$. We define nets as *tuples*, i.e. functions with as domain a finite set. If T is a tuple and $L \in \text{dom}(T)$, we write L_T instead of $T(L)$. We omit the subscript T if there is no confusion possible.

Definition 3.1. A *marked labeled net* (MLN) is a tuple with domain $\{P, T, F, I, O, L, M\}$, satisfying the following constraints. If X is an MLN, then P_X, T_X are disjoint finite sets (the places and transitions respectively), $F_X \in \mathbb{B}((T_X \times P_X) \cup (P_X \times T_X))$ is the flow function, $I_X, O_X \in T_X \rightarrow \mathbb{B}(P_X)$ satisfy $O_X(t)(p) = F_X(t, p)$ and $I_X(t)(p) = F_X(p, t)$ for all $t \in T_X, p \in P_X, L_X \in (P_X \cup T_X) \rightarrow \mathcal{N}$ is a labeling function and $M_X \in \mathbb{B}(P_X)$ is the initial marking.

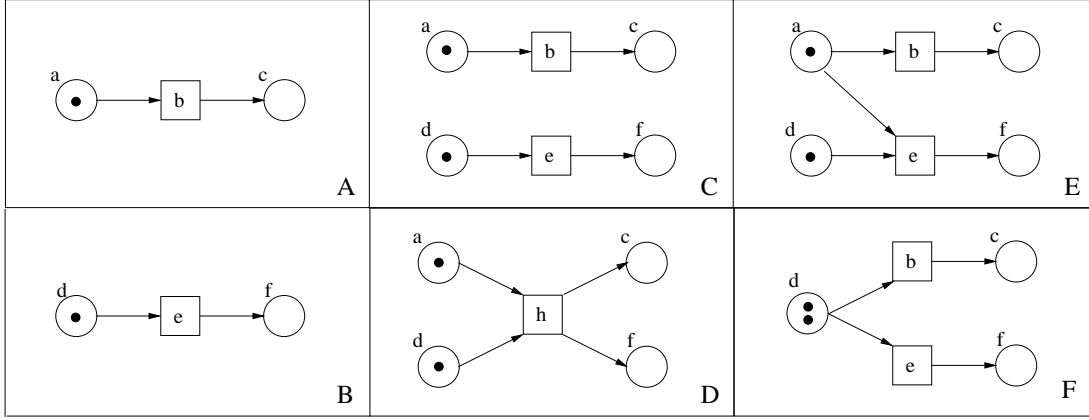


Figure 2: Examples of MLN operators

We draw MLNs as bipartite directed graphs like in Figure 1. There, an MLN is depicted with places x, y, z and transitions t, u, v . The nodes t, x have labels in boldface, t labels are omitted. So $P = \{x, y, z\}, T = \{t, u, v\}, L = \{(t, f), (x, p)\} \cup (\{u, v, y, z\} \times \{t\}), M = [r^2]$ and $F = \{[(t, x)(u, y)(v, x)(v, x)(x, u)(y, v)(z, t)(z, v)]\}$.

The elements of $P \cup T$ are the *nodes* of an MLN. The place $p \in P$ is said to contain $M(p)$ tokens. Usually $I(t), O(t)$ are written as $\bullet t$ and $t\bullet$ respectively. Since F can be constructed from I, O and vice versa, we introduce the functions cf, cio to construct MLNs. We have $X = cf(P, T, F, L, M)$ iff $P_X = P, T_X = T, F_X = F, L_X = L$ and $M_X = M$ and $X = cio(P, T, I, O, L, M)$ iff $P_X = P, T_X = T, I_X = I, O_X = O, L_X = L$ and $M_X = M$. An MLN is called *neat* iff no two places or transitions have the same label. It is called *P-concrete* resp. *T-concrete* iff no places or transitions are t -labeled. A *P-* and *T-concrete neat MLN* is called *controllable*.

We define isomorphism for MLNs. This is an equivalence relation; we tacitly assume that isomorphic MLNs are the same.

Definition 3.2. An *isomorphism* between the MLNs X and Y is an injective function f such that $\{f(p) \mid p \in P_X\} = P_Y, \{f(t) \mid t \in T_X\} = T_Y, L_Y \circ f = L_X, \bar{f}(M_X) = M_Y$ and $\forall a, b : a F_X b \Leftrightarrow f(a) F_Y f(b)$. The MLNs X and Y are *isomorphic* iff there exists an isomorphism between them.

3.1 Net operators

In this subsection, we treat operators for combining labeled nets. The binary merge operator (\parallel) juxtaposes two MLNs with disjoint nodes (disjointness being achieved by applying an isomorphism if necessary). The other operators are unary: consumption ($\gamma_{a,b}$ adding an arc from a b -labeled place to an a -labeled transition), production ($\pi_{a,b}$, adding an arc from an a -labeled transition to a b -labeled place), transition fusion ($\phi_{a,b,c}$, fusing an a and b -labeled transition into a c -labeled one), place fusion (φ_a , fusing all places with the label a) and renaming (ρ_f , applying a relabeling function f to the nodes). A special case of renaming is hiding ($\tau_A = \rho_g$, with $g = id(\mathcal{N} \setminus A) \cup (A \times \{t\})$, labeling all nodes with labels in A to t). To these operators we add the atomic MLNs $P_{a,n}$, consisting of one place with label a and n initial tokens and T_b , being a transition with label b .

We may omit brackets when the operator order is clear. An illustration of the operator's intentions is given in Figure 2. In the figure all nets are controllable; we identify nodes with labels. We have $C =$

$T_\gamma = \{t \in T \mid L(t) \neq a\}$	$\cup \{(t, p) \in T \times P \mid L(t) = a \wedge L(p) = b\},$
$I_\gamma(t) = I(t) \text{ if } t \in T \cap T_\gamma$	$, I_\gamma(t, p) = I(t) + [p] \text{ if } t \in T_\gamma \setminus T,$
$O_\gamma(t) = O(t) \text{ if } t \in T \cap T_\gamma$	$, O_\gamma(t, p) = O(t) \text{ if } t \in T_\gamma \setminus T,$
$L_\gamma(x) = L(x) \text{ if } x \in P \cup (T \cap T_\gamma)$	$, L_\gamma(t, p) = L(t) \text{ if } t \in T_\gamma \setminus T.$
$T_\pi = \{t \in T \mid L(t) \neq a\}$	$\cup \{(t, p) \in T \times P \mid L(t) = a \wedge L(p) = b\},$
$I_\pi(t) = I(t) \text{ if } t \in T \cap T_\pi$	$, I_\pi(t, p) = I(t) \text{ if } t \in T_\pi \setminus T,$
$O_\pi(t) = O(t) \text{ if } t \in T \cap T_\pi$	$, O_\pi(t, p) = O(t) + [p] \text{ if } t \in T_\pi \setminus T,$
$L_\pi(x) = L(x) \text{ if } x \in P \cup (T \cap T_\pi)$	$, L_\pi(t, p) = L(t) \text{ if } t \in T_\pi \setminus T.$
$T_\phi = \{t \in T \mid L(t) \notin \{a, b\}\}$	$\cup \{(t, u) \in T \times T \mid L(t) = a \wedge L(u) = b\},$
$I_\phi(t) = I(t) \text{ if } t \in T \cap T_\phi$	$, I_\phi(t, u) = I(t) + I(u) \text{ if } t \in T_\phi \setminus T,$
$O_\phi(t) = O(t) \text{ if } t \in T \cap T_\phi$	$, O_\phi(t, u) = O(t) + O(u) \text{ if } t \in T_\phi \setminus T,$
$L_\phi(x) = L(x) \text{ if } x \in P \cup (T \cap T_\phi)$	$, L_\phi(t, u) = c \text{ if } t \in T_\phi \setminus T.$
$P_\varphi = \{p \in P \mid L(p) \neq a\}$	$\cup \{p \in P \mid L(p) = a\},$
$F_\varphi(t, p) = F(t, p) \text{ if } p \in P \cap P_\varphi$	$, F_\varphi(t, Q) = \sum_{p \in Q} F(t, p) \text{ if } Q \in P_\varphi \setminus P,$
$F_\varphi(p, t) = F(p, t) \text{ if } p \in P \cap P_\varphi$	$, F_\varphi(Q, t) = \sum_{p \in Q} F(p, t) \text{ if } Q \in P_\varphi \setminus P,$
$L_\varphi(x) = L(x) \text{ if } x \in (P \cap P_\varphi) \cup T$	$, L_\varphi(Q) = a \text{ if } Q \in P_\varphi \setminus P,$
$M_\varphi(p) = M(p) \text{ if } p \in P \cap P_\varphi$	$, M_\varphi(Q) = \sum_{p \in Q} M(p) \text{ if } Q \in P_\varphi \setminus P$

Table 1: Net operator elaborations

$A \parallel B$ (merge), $D = \phi_{b,e,h}(C)$ (transition fusion), $E = \gamma_{e,a}(C)$ (consumption) and $F = \varphi_d(\rho_f(C))$ (renaming, then place fusion), where $f = id(\mathcal{N}^e \setminus \{a\}) \cup \{(a, d)\}$.

The actual definition of the operators is somewhat intricate, due to the fact that in “messy” (non-neat) MLNs several choices may exist for transition fusion or arc addition. This problem is solved by copying transitions that face a nondeterministic choice.

Definition 3.3. Let X and Y be MLNs such that the sets $P_X \cup T_X, P_Y \cup T_Y, (P_X \cup T_X) \times (P_X \cup T_X)$ and $\mathbb{P}(P_X \cup T_X)$ are mutually disjoint. Let $a, b, c \in \mathcal{N}^e, n \in \mathbb{N}, f \in \mathcal{N}^e \rightarrow \mathcal{N}; A \in \mathbb{P}(\mathcal{N}^e)$ such that a, b, c differ.

Then the net operators are defined by

place	$P_{a,n}$	$= cf(\{a\}, \emptyset, \emptyset, id(\{a\}), n[a])$
transition	T_b	$= cf(\emptyset, \{b\}, \emptyset, id(\{b\}), [])$
consumption	$\gamma_{a,b}(X)$	$= cio(P_X, T_\gamma, I_\gamma, O_\gamma, L_\gamma, M)$
production	$\pi_{a,b}(X)$	$= cio(P_X, T_\pi, I_\pi, O_\pi, L_\pi, M)$
transition fusion	$\phi_{a,b,c}(X)$	$= cio(P_X, T_\phi, I_\phi, O_\phi, L_\phi, M)$
place fusion	$\varphi_a(X)$	$= cf(P_\varphi, T_X, F_\varphi, L_\varphi, M_\varphi)$
relabeling	$\rho_f(X)$	$= cf(P_X, T_X, F_X, (f \cup \{(t, t)\}) \circ L_X, M_X)$
merge	$X \parallel Y$	$= cf(P_X \cup P_Y, T_X \cup T_Y, F_X \cup F_Y, L_X \cup L_Y, M_X + M_Y)$

where the T_γ etc. are given in Table 1.

Note that the actual nodes are abstracted from by isomorphy, so the disjointness requirements present no problem. We can show that the operators are congruences w.r.t. isomorphy. For example if f is an isomorphism between X and Y , then $f \cup \{(n, m), (f(n), f(m)) \mid n, m \in dom(f)\}$ is an isomorphism between $\gamma_{a,b}(X)$ and $\gamma_{a,b}(Y)$.

The operators $\gamma_{a,b}, \pi_{a,b}$ and $\phi_{a,b,c}$ are quite simple for neat MLNs if the labels a, b occur in it. In that case $\gamma_{a,b}, \pi_{a,b}$ add an input resp. output arc and $\phi_{a,b,c}$ fuses two transitions. The place fusion φ_a fuses

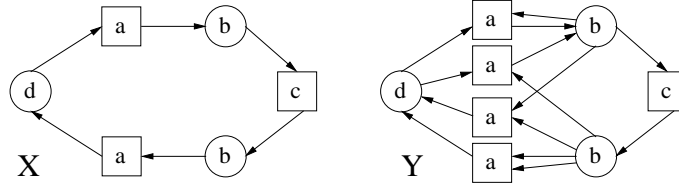


Figure 3: Messy consumption

all places with label a ; for neat MLNs a place can be renamed in order to obtain two places with the same label that can be fused, resulting in a neat MLN. In this case, our operators are similar to the ones in [16], as illustrated in Figure 2. However, our operators are *unconditional*: they are defined for any MLN. For example, if an MLN X does not possess b -labeled places, $\gamma_{a,b}(X)$ is derived from X by removing all a -labeled transitions.

As stated, the operators become somewhat tricky for messy MLNs. Figure 3 gives an example. There, $Y = \gamma_{a,b}(X)$. Since there are two b -labeled places, each of the a -labeled transitions is doubled and for each a, b -combination, an input arc is added.

We prove a few simple equations for the operators. The merge function is symmetric and associative and the γ and π functions do commute w.r.t. composition.

Lemma 3.4. *For MLNs X, Y, Z we have $X \parallel (Y \parallel Z) = (X \parallel Y) \parallel Z$ and $X \parallel Y = Y \parallel X$. Furthermore $f \circ g = g \circ f$ for any two $f, g \in \{\gamma_{a,b} \mid a, b \in \mathcal{N}^e\}$ or any two $f, g \in \{\pi_{a,b} \mid a, b \in \mathcal{N}^e\}$.*

Proof: The merge equations follow from the symmetry and associativity of the union and bag addition. The γ -commutativity is proved by writing out the components of the resulting MLNs. If $c \neq a$ we obtain

$$\begin{aligned}
\gamma_{a,b}(\gamma_{c,d}(X)) &= \text{cio}(P_X, T', I', O', L', M), \\
T' &= \{t \in T_X \mid L_X(t) \notin \{a, c\}\} \\
&\cup \{(t, p) \in T_X \times P_X \mid L_X(t) = a \wedge L_X(p) = b\} \\
&\cup \{(t, p) \in T_X \times P_X \mid L_X(t) = c \wedge L_X(p) = d\}, \\
I'(t) &= I_X(t) \text{ if } t \in T_X \cap T', \\
I'(t, p) &= I_X(t) + [p] \text{ if } t \in T_X \setminus T', \\
O'(t) &= O_X(t) \text{ if } t \in T_X \cap T', \\
O'(t, p) &= O_X(t) \text{ if } t \in T_X \setminus T', \\
L'(x) &= L_X(x) \text{ if } x \in P_X \cup (T \cap T'), \\
L'(t, p) &= L_X(t) \text{ if } t \in T_X \setminus T'.
\end{aligned}$$

For the case $c = a$ we have

$$\begin{aligned}
\gamma_{a,b}(\gamma_{a,d}(X)) &= \text{cio}(P_X, T', I', O', L', M), \\
T' &= \{t \in T_X \mid L_X(t) \neq a\} \\
&\cup \{(t, p, q) \in T_X \times P_X \times P_X \mid L_X(t) = a \wedge L_X(p) = b \wedge L_X(q) = d\}, \\
I'(t) &= I_X(t) \text{ if } t \in T_X \cap T', \\
I'(t, p, q) &= I_X(t) + [p] + [q] \text{ if } t \in T_X \setminus T', \\
O'(t) &= O_X(t) \text{ if } t \in T_X \cap T', \\
O'(t, p, q) &= O_X(t) \text{ if } t \in T_X \setminus T', \\
L'(x) &= L_X(x) \text{ if } x \in P_X \cup (T_X \cap T'), \\
L'(t, p) &= L_X(t) \text{ if } t \in T_X \setminus T'.
\end{aligned}$$

Clearly both MLNs are symmetric; they do not depend on the order. The π -commutativity is fully analogous. \square

With atoms, variables and operators we can build net terms. Let \mathcal{V} be a set of variables.

Definition 3.5. The set $\mathcal{T}(\mathcal{V})$ of *net terms* is defined as the smallest set satisfying

- $\mathcal{V} \cup \{P_{a,n} \mid a \in \mathcal{N} \wedge n \in \mathbb{N}\} \cup \{T_b \mid b \in \mathcal{N}\} \subseteq \mathcal{T}(\mathcal{V})$
- if $X, Y \in \mathcal{T}(\mathcal{V})$ then $X \parallel Y \in \mathcal{T}(\mathcal{V})$,
- if $X \in \mathcal{T}(\mathcal{V})$; g a unary operator, then $g(X) \in \mathcal{T}(\mathcal{V})$.

The set $\mathcal{T}(\emptyset)$ is the set of *closed terms* and $\mathcal{T}(\{\xi\})$ is the set of *contexts*, i.e. terms with a single variable ξ . Closed net terms clearly represent MLNs; different terms may represent the same MLN. We prove that every MLN is isomorphic to a closed net term in normal form. In order to define this normal form, we give some auxiliary definitions. We define addition for functions with bags as range and use it to define extended production and consumption operators.

Definition 3.6. For $\alpha \in \mathbb{B}(\mathcal{N}^e \times \mathcal{N}^e)$, we define the unary MLN operators $\Gamma_\alpha, \Pi_\alpha$ by

$$\begin{aligned} \Gamma_{[]} (X) &= X, & \Pi_{[]} (X) &= X, \\ \Gamma_{[(a,b)]} (X) &= \gamma_{a,b}(X), & \Pi_{[(a,b)]} (X) &= \pi_{a,b}(X), \\ \Gamma_{\alpha+\beta} (X) &= \Gamma_\alpha(\Gamma_\beta(X)), & \Pi_{\alpha+\beta} (X) &= \Pi_\alpha(\Pi_\beta(X)). \end{aligned}$$

By Lemma 3.4 the definitions of Γ_α and Π_α do not depend on the order in which the bag α is constructed. By the same lemma, we can adopt the notation $(\parallel_{i \in I} X_i)$ for the repeated merge of MLNs.

Definition 3.7. A closed MLN term T is in *normal form* iff it has the form

$$(\rho_L \circ \Pi_O \circ \Gamma_I)((\parallel_{t \in T} T_t) \parallel (\parallel_{p \in P} P_{p, M(p)}))$$

Theorem 3.8. *Every MLN is isomorphic to a closed net term in normal form.*

Proof: Let X be an MLN. Choose $N \subseteq \mathcal{N}^e$ such that there exist a bijection $\phi \in ((P_X \cup T_X) \rightarrow N)$. Set $P' = \phi(P_X), T' = \phi(T_X), F' = F_X \circ \phi^{-1}, L' = L_X \circ \phi^{-1}, M' = M_X \circ \phi^{-1}$. Let $Y_1 = (\parallel_{t \in T'} T_t) \parallel (\parallel_{p \in P'} P_{p, M'(p)})$. Then $Y_1 = (P', T', [], id(P' \cup T'), M')$. Let $Y_2 = \Pi_{F \cap (T \times P)}(\Gamma_{F \cap (P \times T)}(Y_1))$. Then $Y_2 = (P', T', F', id(P' \cup T'), M)$. Let $Y = \rho_L(Y_2)$. Then $Y = cf(P', T', F', L', M')$, which is isomorphic to X . \square

4 Operational semantics of MLNs

We define an operational semantics of MLNs in terms of processes. In an MLN, tokens can be added and (if present) removed explicitly from labeled places (cf. the ‘‘open’’ places of [2]). Also firing steps can occur, causing the implicit consumption and production of tokens. We denote addition, removal and firing steps respectively as the $\xrightarrow{\alpha^+}$, $\xrightarrow{\alpha^-}$ and $\xrightarrow{\alpha}$ relations. For example, if the MLN X satisfies $M_X \geq 2[p]$ and $L_X(p) = a$, then X satisfies $X \xrightarrow{2[a]^-} X'$, with $M'_{X'} = M_X - 2[p]$ and also $X' \xrightarrow{2[a]^+} X$. If $t \in T_X, I_X(t) = [p], O_X(t) = [], L_X(t) = b$ then X also satisfies $X \xrightarrow{2[b]} X'$.

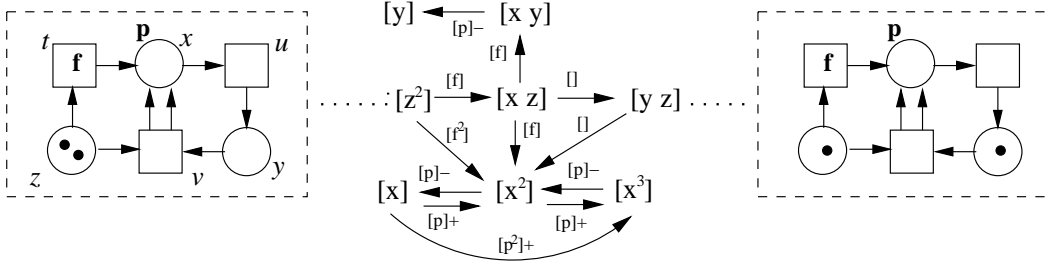


Figure 4: MLN Transition system

Our events \mathcal{E} are thus $\{\alpha \mid \alpha \in \mathbb{B}(\mathcal{N}^e)\} \cup \{\alpha+ \mid \alpha \in \mathbb{B}(\mathcal{N}^e)\} \cup \{\alpha- \mid \alpha \in \mathbb{B}(\mathcal{N}^e)\}$ and our states are the MLNs. Our silent event τ is $[],$ the empty firing. The rules for addition are completely determined by the global rule $x \xrightarrow{\alpha+} y \Leftrightarrow y \xrightarrow{\alpha-} x,$ so we only state the rules for $\xrightarrow{\alpha-}.$ We define addition for removals and additions: $\alpha + +\beta+ = \alpha + \beta+.$

Formally, if X is an MLN and $A \in \mathbb{B}(P_X),$ we denote the MLN $cf(P_X, T_X, F_X, L_X, M_X + A)$ by $A \triangleright X.$ Furthermore, we write $Y = A \triangleleft X$ iff $X = A \triangleright Y.$ We also make use of the $\bar{\cdot}$ and $\tilde{\cdot}$ operators in Definition 2.1, adding a third such operator. If $L \in A \rightarrow \mathcal{N},$ then $\hat{L} \in \mathbb{B}(A) \rightarrow \mathbb{B}(\mathcal{N}^e)$ is the additive closure of $\{([a], [L(a)]) \mid L(a) \neq \iota\} \cup \{([a], []) \mid L(a) = \iota\}.$

Definition 4.1. The MLN transition system is the pair $(\mathcal{M}, \longrightarrow),$ where \mathcal{M} is the set of all MLNs and \longrightarrow is smallest set of triples satisfying for any MLN X and any $A \in \mathbb{B}(P_X), B \in \mathbb{B}(T_X)$

$$\begin{aligned} A \triangleright X &\xrightarrow{\hat{L}_X(A)-} X \text{ if } [t] \not\leq \bar{L}_X(A), \\ \tilde{I}_X(B) \triangleright X &\xrightarrow{\hat{L}_X(B)} \tilde{O}_X(B) \triangleright X. \end{aligned}$$

The condition $[t] \not\leq \bar{L}_X(A)$ for removals entails that tokens cannot be removed (or added, due to the global rule) from internal places. Note that isomorphic MLNs are bisimilar. If f is an isomorphism between X and $Y,$ then the relation $R = \{(cf(P_X, T_X, F_X, L_X, M), cf(P_Y, T_Y, F_Y, L_Y, \tilde{f}(M))) \mid M \in \mathbb{B}(P_X)\}$ is a bisimulation such that $X R Y.$

Example: In Figure 4, part of the process of the MLN in Figure 1 is depicted. Two MLNs are depicted fully; the others are indicated by their state M only. We have added identifiers in italics to the nodes. For example, from the state $[xz],$ transitions t, u can fire concurrently resulting in step $[f]$ to state $[xy].$ The firing of u is hidden (i.e. labeled with the empty bag $[],$) but does not go unnoticed, since it removes a labeled (visible) token. From state $[x],$ e.g. the bag $2[x]$ can be added explicitly, resulting in the state $3[x].$

Note that every MLN X satisfies $X \xrightarrow{[]-} X \xrightarrow{[]+} X \xrightarrow{[]} X.$ The \longrightarrow relations resulting from firing constitute a step semantics of nets, which is what the above semantics amounts to in case all places are unlabeled. Our operators should be compositional w.r.t. this semantics. If A, B are MLNs with the same semantics, then for any context $E,$ the MLNs $E(A), E(B)$ should have the same semantics. To this end, we introduce an alternative structure operational semantics (SOS) for closed net terms.

Definition 4.2. The relation \xrightarrow{e} is the smallest relation satisfying the rules in Table 2 and in addition $X \xrightarrow{\alpha+} X' \Leftrightarrow X' \xrightarrow{\alpha-} X.$ The parameters in Table 2 are $n, m \in \mathbb{N}; a, b, c \in \mathcal{N}^e; \alpha, \beta \in \mathbb{B}(\mathcal{N}^e); A \in \mathbb{P}(\mathcal{N}^e); f \in \mathcal{N}^e \rightarrow \mathcal{N}$ and the MLNs X, X', X'', X''', Y, Y' such that a, b, c differ.

	<i>r(emoval)</i>	<i>s(tep)</i>
AT	$\frac{P_{a,n+m} \xrightarrow{(m[a])^-} P_{a,n}, T_b \xrightarrow{[]^-} T_b}{X \xrightarrow{\alpha^-} X', Y \xrightarrow{\beta^-} Y'}$	$\frac{P_{a,n} \xrightarrow{[]} P_{a,n}, T_b \xrightarrow{n[b]} T_b}{X \xrightarrow{\alpha} X', Y \xrightarrow{\beta} Y'}$
ME	$\frac{X \parallel Y \xrightarrow{(\alpha+\beta)^-} X' \parallel Y'}{X \xrightarrow{\alpha^-} X'}$	$\frac{X \parallel Y \xrightarrow{\alpha+\beta} X' \parallel Y'}{X \xrightarrow{\alpha} X'}$
CO	$\frac{\gamma_{a,b}(X) \xrightarrow{\alpha^-} \gamma_{a,b}(X')}{X \xrightarrow{\alpha^-} X'}$	$\frac{\gamma_{a,b}(X) \xrightarrow{\alpha} \gamma_{a,b}(X')}{X \xrightarrow{(\alpha(b),[a])^-} X'' \xrightarrow{\alpha} X'}$
PR	$\frac{\pi_{a,b}(X) \xrightarrow{\alpha^-} \pi_{a,b}(X')}{X \xrightarrow{\alpha^-} X'}$	$\frac{\pi_{a,b}(X) \xrightarrow{\alpha} \pi_{a,b}(X')}{X \xrightarrow{(\alpha(b),[a])^+} X'' \xrightarrow{\alpha} X'}$
TF	$\frac{\phi_{a,b,c}(X) \xrightarrow{\alpha^-} \phi_{a,b,c}(X')}{X \xrightarrow{\alpha^-} X'}$	$\frac{X \xrightarrow{\alpha+n[ab]} X', \alpha(a) = \alpha(b) = 0}{\phi_{a,b,c}(X) \xrightarrow{\alpha+n[c]} \phi_{a,b,c}(X')}$
PF	$\frac{\varphi_a(X) \xrightarrow{\alpha^-} \varphi_a(X')}{X \xrightarrow{\alpha^-} X', [t] \not\leq \tilde{f}(\alpha)}$	$\frac{X \xrightarrow{n[a]^-} X'' \xrightarrow{n[a]^+} X''' \xrightarrow{\alpha} X'}{\varphi_a(X) \xrightarrow{\alpha} \varphi_a(X')}$
RE	$\frac{\rho_f(X) \xrightarrow{\tilde{f}(\alpha)^-} \rho_f(X')}{X \xrightarrow{\alpha^-} X'}$	$\frac{X \xrightarrow{\alpha} X'}{\rho_f(X) \xrightarrow{\hat{f}(\alpha)} \rho_f(X')}$

Table 2: SOS rules for net operators

We prove that the two semantics agree when applicable, using an auxiliary definition and some lemmas.

Definition 4.3. The *structural equivalence* relation \simeq between closed net terms is the smallest relation such that for any $a \in \mathcal{L}$, $n, m \in \mathbb{N}$ and any unary net operator F ,

$$\frac{}{T_a \simeq T_a, P_{a,n} \simeq P_{a,m}}, \frac{X \simeq X'}{F(X) \simeq F(X')}, \frac{X \simeq X', Y \simeq Y'}{(X \parallel Y) \simeq (X' \parallel Y')}$$

Note that the \simeq relation addresses the structure of terms, so we can have terms X, Y such that $X = Y$ but not $X \simeq Y$.

Lemma 4.4. Let X be a closed net term and let e, X' such that $X \xrightarrow{e} X'$. Then there exists a closed net term X'' such that $X' = X''$ and $X \simeq X''$.

Proof: According to Definition 4.1, it is sufficient to prove that for any closed net term X we have $X \simeq (A \triangleright X)$ and $X \simeq (A \triangleleft X)$ whenever $A \triangleright X$ and/or $A \triangleleft X$ are defined. We prove this by induction w.r.t. the structure of X .

If X is a transition, then $A \triangleright X$ exists only if $A = []$, so $A \triangleright X = X$. If X is a place $P_{a,n}$, then $A \triangleright X$ exists only if $A = m[a]$ and $A \triangleright X = P_{a,n+m}$. In both cases, $X \simeq A \triangleright X$.

If $X = Y \parallel Z$ and $A \triangleright X$ is defined, so $A \in \mathbb{B}(P_X)$, then by the disjointness of the merge, we can write $A = B + C$, where $B \in \mathbb{B}(P_Y)$, $C \in \mathbb{B}(P_Z)$ and thus $A \triangleright X = (B \triangleright Y) \parallel (C \triangleright Z)$, and by the induction hypothesis we infer that $Y \simeq (B \triangleright Y)$ and $Z \simeq (C \triangleright Z)$.

If $X = \gamma_{a,b}(Y)$ and $A \triangleright X$ is defined, then by Definition 3.3, $A \triangleright Y$ is defined too and $X = \gamma_{a,b}(A \triangleright Y)$. We can use induction as above. The other unary operators except place fusion are analogous.

If $X = \varphi_a(Y)$ and $A \triangleright X$ is defined, so $A \in \mathbb{B}(P_X)$, then we can write $A = B + C$, where $B \in \mathbb{B}(P_Y)$ and $C = \sum_i k_i [p_i] \in \mathbb{B}(\{\{p \in P_Y \mid L_Y(p) = a\}\})$. Let $k = \sum_i k_i$ be the size of C . If $k > 0$, we can find a $p \in P_Y$ such that $L_Y(p) = a$ and $A \triangleright X = \varphi_a((B + k[p]) \triangleright Y)$. So we can use induction.

The \triangleleft cases are similar to the \triangleright case. \square

Note that the rules in Table 2 can be written as a set of rules of the form $P(\mapsto) \vdash C(\mapsto)$ about the \mapsto relation (P is the premise and C the conclusion). The next lemma discusses replacing \mapsto by \longrightarrow in those rules.

Lemma 4.5.

1. If $P(\mapsto) \vdash C(\mapsto)$ is a rule in Table 2, then $P(\longrightarrow) \Rightarrow C(\longrightarrow)$.
2. If $X \xrightarrow{e} X'$ for some closed net term X , there exists a rule $P(\mapsto) \vdash C(\mapsto)$ in Table 2 such that $C(\longrightarrow)$ equals $X \xrightarrow{e} X'$ and $P(\longrightarrow)$ holds.

Proof: The proof of the lemma is by tedious case analysis and will be treated in Appendix A. \square

We now prove our theorem.

Theorem 4.6. *Let X be a closed net term. Then for any event e and closed net term X'*

$$X \xrightarrow{e} X' \Leftrightarrow X \mapsto^e X'.$$

Proof: If $X \mapsto^e X'$, there exists a finite derivation chain of rules $P_i(\mapsto) \vdash C_i(\mapsto)$ allowing to deduce this fact. By Lemma 4.5.1, the rules $P_i(\longrightarrow) \Rightarrow C_i(\longrightarrow)$ hold, so we can prove $X \xrightarrow{e} X'$.

We use induction on the structure of X to prove that $X \xrightarrow{e} X'$ implies $X \mapsto^e X'$. If X is an atom, and $X \xrightarrow{e} X'$, then by Lemma 4.5.2 there is a rule such that $X \xrightarrow{e} X'$ equals $C(\longrightarrow)$; the only rules that apply are AT rules having the empty premise (*true*), so $P(\mapsto)$ holds and thus $C(\mapsto)$ and hence $X \mapsto^e X'$.

If X is not an atom, it is of the form $X_1 \parallel X_2$ or $f(Y)$ for some unary operator f . If $X \xrightarrow{e} X'$, then by Lemma 4.5.2 there is a rule such that $X \xrightarrow{e} X'$ equals $C(\longrightarrow)$ and $P(\longrightarrow)$ holds. There is only one set of rules that apply, depending on the operator. In each of these rules, the premise P is of the form $X_1 \xrightarrow{e_1} X'_1, X_2 \xrightarrow{e_2} X'_2$ in the merge case and $Y \xrightarrow{d_1} \dots \xrightarrow{d_n} Y'$ in the unary operator case. By Lemma 4.4, the final terms X'_1, X'_2, Y' and any intermediary terms are structurally equivalent to X_1, X_2, Y respectively. By the induction hypothesis, we may thus assume that $X_1 \mapsto^{e_1} X'_1, X_2 \mapsto^{e_2} X'_2$ or $Y \xrightarrow{d_1} \dots \xrightarrow{d_n} Y'$ and from the rule $P \vdash C$ we deduce $X \mapsto^e X'$. \square

As a corollary, we deduce the desired compositionality of the operators w.r.t. the semantics.

Theorem 4.7. *Let X, Y be MLNs and E a context. If $X \sim Y$ then $E(X) \sim E(Y)$ and if $X \sim_b Y$ then $E(X) \sim_b E(Y)$.*

Proof: Let X', Y' be closed net terms isomorphic to X and Y respectively. Since isomorphy implies bisimilarity and since bisimilarity is transitive, we deduce $X' \sim Y'$. By Theorem 4.6 and since the

SOS rules are in tyft/tyxt format [7], $E(X') \sim E(Y')$. Since the operators and thus contexts are congruences w.r.t. isomorphy, $E(X) \sim E(Y)$.

We can repeat the same proof for branching bisimilarity, using [6] instead of [7]. It is essential that the a ι -labeled node cannot be relabeled. We do not need rootedness due to the absence of a choice-like operator. \square

5 Net equivalence

In this section, we will discuss the equivalence notions we have so far: isomorphy, strong and branching bisimilarity. We already saw that isomorphy implies strong bisimilarity implies branching bisimilarity. Bisimilarity is connected to HML (Hennessy-Milner) temporal logic, which we define below. In order to avoid inconsistencies, conjunction is restricted.

Definition 5.1. The sets \mathcal{H} of HML predicates satisfying is the smallest set such that

$$\top \in \mathcal{H}, \quad \frac{L \in \mathcal{H}}{\neg L \in \mathcal{H}}, \quad \frac{A \subseteq \mathcal{H} \text{ countable}}{\bigwedge A \in \mathcal{H}}, \quad \frac{L \in \mathcal{H}, a \in \mathcal{N}}{\diamond_a L \in \mathcal{H}}.$$

Let X be an MLN. The set of predicates L such that X satisfies L (notation $X \models L$) is the smallest set satisfying

$$\frac{}{X \models \top} \quad \frac{X \not\models L}{X \models \neg L} \quad \frac{\forall M \in A : X \models M}{X \models \bigwedge A} \quad \frac{\exists X' : X \xrightarrow{a} X' \wedge X' \models L}{X \models \diamond_a L}$$

We introduce the following abbreviations:

$$\begin{aligned} \perp &= \neg \top & L \wedge M &= \bigwedge \{L, M\} & \bigvee A &= \neg \bigwedge \{\neg L \mid L \in A\} \\ L \vee M &= \bigvee \{L, M\} & \square_a L &= \neg \diamond_a \neg L & \forall i \in I : L_i &= \bigwedge \{L_i \mid i \in I\} \\ & & & & \exists i \in I : L_i &= \bigvee \{L_i \mid i \in I\} \end{aligned}$$

The combination of HML with our semantics allows to formulate both state-based and action-based properties of a component, like $\square_a \diamond_b \top$ (after every a -step a b -labeled token is present).

Two MLNs are bisimilar iff they satisfy the same HML formula's. Two MLNs are branching bisimilar iff they satisfy the same formula's from a somewhat weaker language [5] that abstracts from silent events. Instead of the unary \diamond_a operator this subset has the binary "until" operator U_a , where $\phi U_a \psi$ is equivalent to $\bigvee L_{\phi, \psi}$, where $L_{\phi, \psi}$ is the set $\{\phi \wedge \diamond_a \psi, \phi \wedge \diamond_\tau(\phi \wedge \diamond_a \psi), \phi \wedge \diamond_\tau(\phi \wedge \diamond_\tau(\phi \wedge \diamond_a \psi)) \dots\}$. We define subclasses of MLNs for which the equivalence notions coincide.

Theorem 5.2. *For controllable MLNs, bisimilarity coincides with isomorphy. For T -concrete MLNs, bisimilarity coincides with branching bisimilarity.*

Proof: Let X, Y be controllable MLNs. We can find Z, W isomorphic to X, Y respectively such that L_Z, L_W are identity functions. If $Z \neq W$, since we can interchange Z and W , one of the following statements must hold for a $p \in P_Z, t \in T_Z$ or α with $\alpha \leq M_Z$.

- 1 $p \notin P_W$
- 2 $t \notin T_W$
- 3 $\alpha \not\leq M_W$
- 4 $M_Z = M_W \wedge t \in T_W \wedge I_Z(t) \not\leq I_W(t)$
- 5 $M_Z = M_W \wedge t \in T_W \wedge (O_Z(t) - I_Z(t)) \not\leq (O_W(t) - I_W(t))$

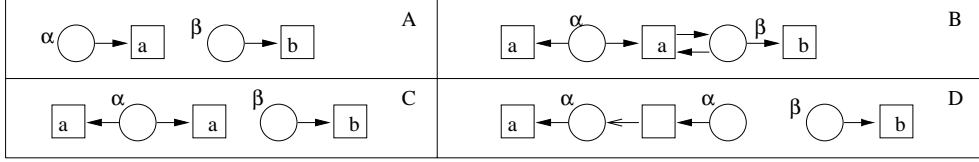


Figure 5: Equivalent nets

In each case, we give a HML predicate L such that $Z \models L$ and $W \not\models L$. So if Z and W are bisimilar, then $Z = W$ and thus X, Y are isomorphic. We now give the choices for L , writing $\langle \alpha \rangle$ instead of $\diamond \alpha$.

- 1 $\langle [p] \rangle \top$
- 2 $\langle I_Z(t) \rangle \langle [t] \rangle \top$
- 3 $\langle \alpha \rangle \top$
- 4 $\langle M_Z \rangle \langle I_Z(t) \rangle \langle [t] \rangle \top$
- 5 $\langle I_Z(t) \rangle \langle [t] \rangle \langle (M_Z(t) + O_Z(t)) \rangle \top$

If R is a branching bisimulation, its restriction to T -concrete processes is a strong bisimulation. This fact proves the second statement. \square

In Figure 5 a few MLNs are depicted. The nets A, B and C are bisimilar but not isomorphic. Nets A and D are branching bisimilar but not bisimilar.

Bisimilarity of nets is undecidable [4], but we give some simple rules for the reduction of nets modulo bisimilarity. We define the following reduction operators: R_n (node removal), Φ_A (place fusion) and $W_{B,C}$ (place weaving).

Definition 5.3. Let X be an MLN with a node $n \in P_X \cup T_X$, a place set $A \subseteq P_X$ such that $\forall p, q \in A : L(p) = L(q)$ and place sets $B, C \subseteq P_X$. Then $R_n(X) = Y, \Phi_A(X) = Z, W_{B,C}(X) = W$, where Y, Z, W are the MLNs defined in Table 3. By isomorphy, we may assume that added nodes in that table are new.

Note that $\varphi_a(X) = \Phi_A(X)$ with $A = \{p \mid L_X(p) = a\}$. Also, the place weave operator $W_{A,B}$ resembles transition fusion. We will conditions under which the application of a reduction operator leads to a result bisimilar to the operand net. We start by defining some concepts.

Definition 5.4. Let X be an MLN. A *place autobisimulation* of X is a relation $R \in P_X \times P_X$ containing the identity relation $id(P_X)$ such that the relation $(A \triangleright (M_X \triangleleft X), B \triangleright (M_X \triangleleft X)) \mid A \bar{R} B$ is a bisimulation. Places $p, q \in P_X$ are *place autobisimilar* iff there exists a place bisimulation R such that $p R q$.

A place $p \in P_X$ is called *redundant* in X iff $L_X(p) = \iota$ and for all $Y \in \mathcal{R}(X)$ and $\beta \in \mathbb{B}(T_X)$ we have $M_Y(p) \geq (M_Y \div \tilde{I}_Y(\beta)) \tilde{I}_Y(\beta)(p)$.

It is easy to prove that place autobisimilarity itself is a place autobisimulation. It can be computed by starting with the relation $\{(p, q) \mid L_X(p) = L_X(q)\}$ and removing pairs that turn out not to be related, c.f. [1]. If p, q are place autobisimilar, they must have the same label.

Redundancy of a place p means that p never contains too few tokens compared to the other places; if a step β cannot occur, it cannot occur even if an arbitrary amount of tokens were added to p . Often, redundancy of a place can be proved by invariants.

We now formulate reduction rules allowing place fusion, node removal and weaving respectively. Place removal is allowed for redundant places and transition removal is allowed for duplicate transi-

$$\begin{aligned}
P_Y &= P_X \setminus \{n\}, T_Y = T_X \setminus \{n\} \\
F_Y &= F_X \upharpoonright ((P_Y \cup T_Y) \times (P_Y \cup T_Y)) \\
L_Y &= L_X \upharpoonright (P_Y \cup T_Y) \\
M_Y &= M_X \upharpoonright P_X \\
P_Z &= (P_X \setminus A) \cup \{A\}, T_Z = T_X \\
\forall t \in T_Z, p \in P_X \cap P_Z : I_Z(t)(p) &= I_X(t)(p) \wedge O_Z(t)(p) = O_X(t)(p) \\
\forall t \in T_Z : I_Z(t)(A) &= \sum_{p \in A} I_X(t)(p) \wedge O_Z(t)(A) = \sum_{p \in A} O_X(t)(p) \\
\forall n \in (P_Z \cap P_X) \cup T_Z : L_Z(n) &= L_X(n) \\
\forall p \in A : L_Z(A) &= L_X(p) \\
\forall p \in (P_Z \cap P_X) : M_Z(p) &= M_X(p) \\
M_Z(A) &= \sum_{p \in A} M_X(p) \\
P_W &= (P_X \setminus (B \cup C)) \cup \{(p, q) \mid p \in B \wedge q \in C \wedge b \neq c\} \\
T_W &= T_X \\
\forall t \in T_Z, p \in P_X \cap P_W : I_W(t)(p) &= I_X(t)(p) \wedge O_W(t)(p) = O_X(t)(p) \\
\forall t \in T_Z, (p, q) \in P_W \setminus P_X : I_W(t)(p, q) &= I_X(t)(p) + I_X(t)(q) \\
\forall t \in T_Z, (p, q) \in P_W \setminus P_X : O_W(t)(p, q) &= O_X(t)(p) + O_X(t)(q) \\
\forall n \in (P_W \cap P_X) \cup T_W : L_W(n) &= L_X(n) \\
\forall (p, q) \in P_W \setminus P_X : L_W(p, q) &= \iota \\
\forall p \in (P_W \cap P_X) : M_W(p) &= M_X(p) \\
\forall (p, q) \in P_W \setminus P_X : M_W(p, q) &= M_X(p) + M_X(q)
\end{aligned}$$

Table 3: Reduction operator elaborations

tions or ι -labeled transitions with identical input and output bags. Weaving is allowed between input and output sets of ι -labeled transitions that have no conflicts.

Theorem 5.5. *Let X be an MLN.*

1. *If p is redundant in X , then X and $R_p(X)$ are bisimilar.*
2. *If $t \in T_X$ such that $I_X(t) = O_X(t)$ and $L_X(t) = \iota$ or if there exist transitions t, u in T_X such that $I_X(t) \subseteq I_X(u)$ and $I_X(u) + O_X(t) = I_X(t) + O_X(u)$ and $L_X(t) = L_X(u)$, then X and $R_t(X)$ are bisimilar.*
3. *If R is a place autobisimulation of X and $F \subseteq P_X$ such that $\forall p, q \in F : p R q$, then $\Phi_F(X)$ is bisimilar to X .*
4. *If $t \in T_X$ with $L_X(t) = \iota$ such that there exist $A, B \subseteq P_X$ and $I_X(t) = \sum_{p \in A} [p]$ and $O_X(t) = \sum_{p \in B} [p]$ and $\hat{L}_X(I_X(t)) = \hat{L}_X(O_X(t)) = []$ and $\forall u \in T_X, r \in I_X(t) : r \notin I_X(u)(r)$, then X and $R_t(W_{I_X(t), O_X(t)}(X))$ are branching bisimilar.*

Proof: We start with redundant place removal.

Let p be a redundant place in X . Set $Y = R_p(X)$ and $Y_0 = M_Y \triangleleft Y$. We prove that the function $R = \{(Z, (M_Z - M_Z(p))[p]) \triangleright Y_0 \mid Z \in \mathcal{R}(X)\}$ is a bisimulation and $M_X R M_Y$. If $U R V$, then since $U \in \mathcal{R}(X)$, the redundancy of p entails that for each $\beta \in \mathbb{B}(T_U)$ we have $\tilde{I}_U(\beta) = \tilde{I}_V(\beta)$. If $U \xrightarrow{\alpha} U'$ and α is a step, that there exists a a bag $\beta \in \mathbb{B}(T_U)$ and a U_0 such that $U = \tilde{I}_U(\beta) \triangleright U_0$ and $U' = \tilde{O}_U(\beta) \triangleright U_0$ and $\alpha = \hat{L}_U(\beta)$. Let V' such that $U' R V'$. From the above it is easy to prove that $V \xrightarrow{\alpha} V'$. For additions/removals the fact that $L_X(p) = \iota$ is sufficient.

Next comes transition removal.

The relation $R = \{(Z, R_t(Z)) \mid Z \in \mathcal{R}(X)\}$ is a bisimulation. Its proof is by case analysis, but rather straightforward.

Next comes the fusion rule.

Set $Y = \Phi_F(X)$ and let $X_0 = M_X \triangleleft X$, $Y_0 = M_Y \triangleleft Y$. Let $f = id(P_X \cap P_Y) \cup \{(p, F) \mid p \in F\}$. By the definition of the fusion operator, $(A \triangleright X_0) \xrightarrow{\alpha} (A' \triangleright X_0) \Rightarrow (\bar{f}(A) \triangleright Y_0) \xrightarrow{\alpha} (\bar{f}(A') \triangleright Y_0)$ and $(B \triangleright Y_0) \xrightarrow{\alpha} (B' \triangleright Y_0)$ implies $\exists A, A' : \bar{f}(A) = B \wedge \bar{f}(A') = B' \wedge (A \triangleright X_0) \xrightarrow{\alpha} (A' \triangleright X_0)$.

Let $Q = \{(A \triangleright X_0, B \triangleright Y_0) \mid A \bar{f} B\}$. We shall prove that Q is a bisimulation. Let $U \ Q \ V$, where $U = A \triangleright X_0$, so $V = \bar{f}(A) \triangleright Y_0$. If $U \xrightarrow{\alpha} U'$ (where $U' = A' \triangleright X_0$), then we can choose $V' = \bar{f}(A') \triangleright Y_0$, so Q is a simulation. Conversely, if $V \xrightarrow{\alpha} V'$, there exist A_0, A'_0 such that $\bar{f}(A_0) = \bar{f}(A)$, $V' = \bar{f}(A'_0) \triangleright Y_0$ and $(A_0 \triangleright X_0) \xrightarrow{\alpha} (A'_0 \triangleright X_0)$. Since $\bar{f}(A_0) = \bar{f}(A)$, we have by the definition of f that $A_0 \bar{R} A$. Since R is a place bisimulation, there exists an A' such that $U \xrightarrow{\alpha} (A' \triangleright X_0)$ and $\bar{f}(A') = \bar{f}(A'_0)$. So we can choose $U' = A' \triangleright X_0$, proving that Q^{-1} is a simulation. Clearly, $\bar{f}(M_X) = M_Y$, so X and Y are bisimilar.

Finally comes the weave rule.

Set $Y = W_{I_X(t), O_X(t)}(X)$. We show that the X and Y are branching bisimilar. Since $L_Y(t) = \iota$ and $I_Y(t) = O_Y(t)$ we can then apply the transition removal rule.

Set $U_A = A \triangleright (M_X \triangleleft X)$. We show that the function $f = \{(U_A, W_{I_X(t), O_X(t)}(U_A)) \mid A \in \mathbb{B}(P_X)\}$ is a branching bisimulation. Let $U \in dom(f)$, so U and X only differ in their M component (as do $f(U)$ and $f(X)$) and $\hat{L}_U(M_U) = \hat{L}_{f(U)}(M_{f(U)})$. If $U \xrightarrow{\alpha} U'$ then $f(U) \xrightarrow{\alpha} f(U')$. If $f(U) \xrightarrow{\alpha} V'$, there exists an U' such that $U \xrightarrow{\alpha} U'$ and $V' = f(U')$. The global rule takes care of additions.

For steps, note that by the weave construction, $f(I_U(u) \triangleleft U) = I_{f(U)}(u) \triangleleft f(U)$ if $M_U \geq I_U(u)$ and $f(O_U(u) \triangleright U) = O_{f(U)}(u) \triangleright f(U)$ for all $u \in T_X$ and $U \in dom(f)$. Suppose $U \xrightarrow{\alpha} U'$, then there exists a $\beta \in \mathbb{B}(T_U)$ and U_0 such that $U = \tilde{I}_U(\beta) \triangleright U_0$ and $U' = \tilde{O}_U(\beta) \triangleright U_0$ and $\hat{L}(\beta) = \alpha$. By the weave construction, $f(U) = \tilde{I}_{f(U)}(\beta) \triangleright f(U_0)$ and $f(U') = \tilde{O}_{f(U)}(\beta) \triangleright f(U_0)$. So $f(U) \xrightarrow{\hat{L}_{f(U)}(\beta)} f(U')$ and $L_{f(U)}$ and L_U are the same on transitions, so $f(U) \xrightarrow{\alpha} f(U')$.

For $U \in dom(f)$, let $g(U) = U - n \cdot I_U(t) + n \cdot O_U(t)$, with $n = U \div I_U(t)$. We have $f(U) = f(V)$ iff $g(U) = g(V)$, so also $f(g(U)) = f(U)$. By the condition that $r \notin I_X(u)$ if $u \in T_X$, $r \in I_X(t)$ and since $L_X(t) = \iota$, it follows that if $U \xrightarrow{\alpha} U'$, then $g(U) \xrightarrow{\alpha} g(U')$. Let $V \in ran(f)$. Let $h = \{(f(U), g(U)) \mid U \in dom(f)\}$, which is a function. By the construction of f , it follows that if $V \xrightarrow{\alpha} V'$, then $h(V) \xrightarrow{\alpha} h(V')$. So if $f(U) \xrightarrow{\alpha} V'$, we set $U'' = h(U)$ and we can find $U' = g(V')$ so that $U \implies U'' \xrightarrow{\alpha} U'$, $f(U'') = f(U)$ and $f(U') = V'$. Therefore, f is a branching bisimulation. \square

Note that after nontrivial place fusion duplicate transitions can be removed. Also note that the weave rule may remove a transition by augmenting the number of places, so it does not necessarily simplify the net. In Figure 6, examples of net reductions are given. In that figure, $Y = (R_t \circ R_{u'} \circ \Phi_{\{x,z\}} \circ \Phi_{\{y,w\}})(X)$ is bisimilar to X by the fusion and transition removal rules. Also $W = (R_t \circ W_{\{x,y\},\{p\}} \circ R_z)(Z)$ is branching bisimilar to Z by the place removal (since the place z is redundant in Z) and weave rules.

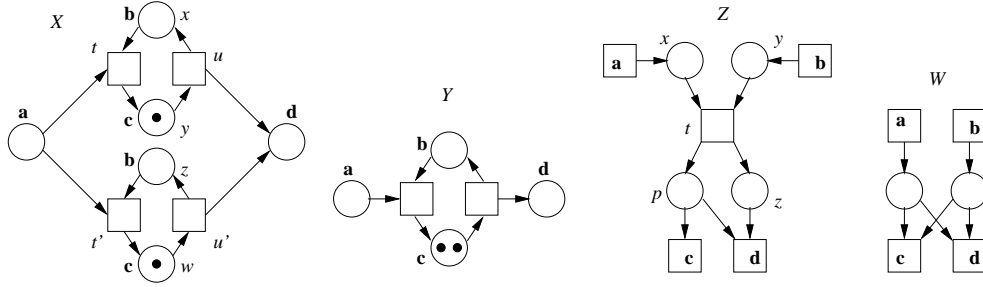


Figure 6: Net reduction examples

6 Example

We use MLNs to model components. The tokens in the labeled places define the component’s visible state, that can be inspected and updated by components in its interface. The transitions represent the actions or methods that can be called by interfacing components, directly by fusion (rendez-vous) or indirectly by message passing. The unlabeled places and transitions represent the internal state and hidden methods of the component.

The specification of a component is the equivalence class modulo branching bisimilarity of its MLN model. The MLN itself defines the component’s implementation. It is possible to change the implementation without altering the specification. By compositionality, the specification of a component will remain the same if the specification of the subcomponents is not changed.

We give a small data communication example. A buffer α that can be filled with tokens is connected to a component I that offers them one-by-one to the network N . The network transfers them to another location, where the tokens are inserted in buffer β by a third component O . The interface between I , O and N is by transition fusion, between I and α by consumption and between O and β by production. In Figure 7, the specifications of the subcomponents and their interconnection are shown.

Net X is $P_{\alpha,0} \parallel I \parallel N \parallel O \parallel P_{\beta,0}$. Net Y is $(\tau_{\{a,c,c'\}} \circ \gamma_{a,\alpha} \circ \pi_{w,\beta} \circ \phi_{s,r,c} \circ \phi_{s',r',c'})(X)$. By the “weave” rule, the middle transitions can be short-circuited modulo branching bisimilarity, leading to Z . Z is not branching bisimilar to V ; V satisfies $(\top U_{2[\alpha]^+} \top) U_{\tau} \dashv (\top U_{[\alpha]^-} \top)$: we can pass from the stage where α has two tokens to the stage where α is empty by a single hidden step. This cannot occur in Z (and Y). The nets Z and V are weakly bisimilar, though. The net W is not equivalent to Y in any way; in W , tokens consumed from α appear immediately in β .

7 Conclusion

This paper uses techniques combines Petri net modeling and techniques from process algebra. The aim is to support Petri net modeling, in contrast to the Petri box algebra [13] where Petri nets support algebraic modeling.

We define a semantics for “open” nets (MLNs) and operators for combining them. The semantics preserves the state-event duality typical of Petri nets. Our step semantics does not preserve causal dependencies between events, unlike e.g. [16, 11]. Sacrificing causality allows simple SOS rules and the possibility to interface with algebraically specified components.

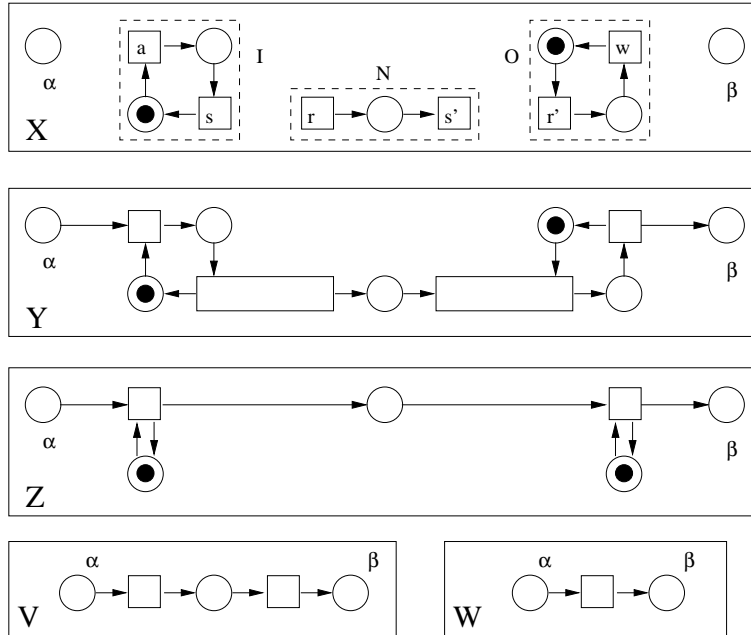


Figure 7: Token transfer example

In order to arrive at a fully compositional net-based specification language, we need to address some version of “colored” nets [10, 8]. The addition of color does not invalidate the approach presented here; problems are mainly technical.

References

- [1] C. Autant and Ph. Schnoebelen. Place Bisimulations in Petri Nets. In K. Jensen, editor, *Application and Theory of Petri Nets 1992, 13th. International Conference, Proceedings*, volume 616 of *Lecture Notes in Computer Science*, pages 45–61. Springer-Verlag, Berlin, 1992.
- [2] P. Baldan, A. Corradini, H. Ehrig, and R. Heckel. Compositional modeling of reactive systems using open nets. In *Proceedings CONCUR '01*, volume 2154 of *Lecture Notes in Computer Science*, pages 502–518. Springer-Verlag, Berlin, 2001.
- [3] T. Basten. Branching Bisimilarity is an Equivalence Indeed! *Information Processing Letters*, 58:141–147, 1996.
- [4] P. Jančar. Undecidability for Petri nets and some related problems. *Theoretical Computer Science*, 148(2):281–301, 1995.
- [5] R. De Nicola and F.W. Vaandrager. Three logics for branching bisimulation. *Journal of the ACM*, 42(2):458–487, 1995.
- [6] W.J. Fokkink. Rooted branching bisimulation as a congruence. *Journal of Computer and System Sciences*, 60(1):13–37, 2000.
- [7] J.F. Groote and F.W. Vaandrager. Structured operational semantics and bisimulation as a congruence. *Information and Computation*, 100(2):202–260, 1992.

- [8] K.M. van Hee. *Information Systems Engineering: a Formal Approach*. Cambridge University Press, Cambridge, 1994.
- [9] M. Hennesy and R. Milner. Algebraic Laws for Nondeterminism and Concurrency. *Journal of the ACM*, 32(1):137–161, 1985.
- [10] K. Jensen. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use*. EATCS monographs on Theoretical Computer Science. Springer–Verlag, Berlin, 1992.
- [11] E. Kindler. A compositional partial order semantics for Petri net components. In P. Azéma and G. Balbo, editors, *Proceedings ATPN '97*, volume 1248 of *Lecture Notes in Computer Science*. Springer–Verlag, Berlin, 1997.
- [12] E. Kindler and T. Vesper. ESTL: A temporal logic for events and states. In J. Desel and M. Silva, editors, *Proceedings ATPN '98*, volume 1420 of *Lecture Notes in Computer Science*. Springer–Verlag, Berlin, 1997.
- [13] M. Koutny and E. Best. Operational and denotational semantics for the box algebra. *Theoretical Computer Science*, 211:1–83, 1999.
- [14] W.E. McUumber and B.H.C. Cheng. A general framework for formalizing UML with formal languages. In *Proceedings ICSE '01*, pages 433–442. IEEE press, 2001.
- [15] G.D. Plotkin. A structural approach to operational semantics. Technical Report DAIMI FN-19, Computer Science Dept., Aarhus University, 1981.
- [16] L. Priese and H. Wimmel. A uniform approach to true-concurrency and interleaving semantics for Petri nets. *Theoretical Computer Science*, 206:219–256, 1998.

Appendix A:

We prove Lemma 4.5 here. We start with the first proposition:

If $P(\vdash \longrightarrow) \vdash C(\vdash \longrightarrow)$ is a rule in Table 2, then $P(\longrightarrow) \Rightarrow C(\longrightarrow)$.

Proof: The proof is by case analysis, using Definitions 3.3 and 4.1.

AT: (the premise is true, so we must prove the conclusions)

We have $P_{a,n+m} \xrightarrow{m[a]^-} (m[a] \triangleleft P_{a,n+m}) = P_{a,n}$. We have $T_b = (n.\tilde{I}_{T_b} \triangleright T_b) \xrightarrow{n[b]} (\tilde{O}_{T_b} \triangleright T_b) = T_b$ and of course $P_{a,n} \xrightarrow{\square} P_{a,n}$ and $T_b \xrightarrow{\square} T_b$.

ME:

If $X \xrightarrow{\alpha^-} X'$ and $Y \xrightarrow{\beta^-} Y'$, then there exist A, B with $\bar{L}_X(A) = \alpha$, $\bar{L}_X(B) = \beta$ such that $X' = A \triangleleft X$ and $Y' = B \triangleleft Y$, so $X' \parallel Y' = (A + B) \triangleleft (X \parallel Y)$ and since $[t] \not\leq A + B$, we have $X \parallel Y \xrightarrow{(\alpha+\beta)^-} X' \parallel Y'$. The proof for the step relation is even simpler.

CO_r,PR_r,TF_r,PF_r:

If $X \xrightarrow{\alpha^-} X'$, then there is an A with $\bar{L}_X(A) = \alpha$ such that $X' = A \triangleleft X$. Then we also have $f(X') = A \triangleleft f(X)$ for $f \in \{\gamma_{a,b}, \pi_{a,b}, \phi_{a,b,c}\}$. There is also an A' with $\bar{L}_{\varphi_a(X)}(A') = \alpha$ such that $\varphi_a(X') = A' \triangleleft \varphi_a(X)$. So $f(X) \xrightarrow{\alpha^-} f(X')$ in all these cases.

RE:

If $X \xrightarrow{\alpha^-} X'$, then there is an A with $\bar{L}_X(A) = \alpha$ such that $X' = A \triangleleft X$. Thus $\rho_f(X') = A \triangleleft \rho_f(X)$ and if $[t] \not\leq \bar{f}(\alpha)$, then $[t] \not\leq \bar{L}_{\rho_f(X)}(A)$, so $\rho_f(X) \xrightarrow{\bar{f}(\alpha)^-} \rho_f(X')$. The REs rule is analogous, but simpler since the extra condition is missing.

CO_s,PR_s:

If $X \xrightarrow{\alpha(b).[a]^-} X'' \xrightarrow{\alpha} X'$, then there is an $n = \alpha(b)$ and A, E with $\bar{L}_X(A) = n[a]$ and $\hat{L}_X(E) = \alpha$ such that $X' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleleft (A \triangleleft X))$. Since $(\hat{L}_X(E))(b) = n$, we can find a list of transition-place pairs $(t_1, p_1) \dots (t_n, p_n)$ such that $L_X(p_i) = a$, $L_X(t_i) = b$ for all i and $\Sigma_i[p_i] = A$ and $\hat{L}_X(E - \Sigma_i[t_i])(b) = 0$. This means that the bag $F = E - \Sigma_i[t_i] + \Sigma_i[(t_i, p_i)]$ contains transitions of $Y = \gamma_{a,b}(X)$ and $\hat{L}_Y(F) = \alpha$ and that $Y' = \tilde{O}_Y(F) \triangleright (\tilde{I}_Y(F) \triangleleft Y)$ exists and $Y' = \gamma_{a,b}(X')$. So $Y \xrightarrow{\alpha} Y'$. The production step rule is fully analogous.

TF_s:

If $X \xrightarrow{\alpha+n[ab]} X'$ and $\alpha(a) = \alpha(b) = 0$, then there are E, F such that $\hat{L}_X(E) = \alpha$, $\hat{L}_X(F) = n[ab]$ and $X' = \tilde{O}_X(E + F) \triangleright (\tilde{I}_X(E + F) \triangleleft X)$. We can find a list of transition pairs $(t_1, u_1) \dots (t_n, u_n)$ such that $L_X(t_i) = a$, $L_X(u_i) = b$ for all i and $\Sigma_i[t_i, u_i] = F$. Since $\alpha(a) = \alpha(b) = 0$, the bag $G = E + \Sigma_i[(t_i, u_i)]$ is a transition bag of $Y = \phi_{a,b,c}(X)$ and $\tilde{I}_Y(G) = \tilde{I}_X(E + F)$, $\tilde{O}_Y(G) = \tilde{O}_X(E + F)$ and $\hat{L}(G) = \alpha + n[c]$. Thus $Y' = \tilde{O}_Y(G) \triangleright (\tilde{I}_Y(G) \triangleleft Y)$ exists and $Y' = \phi_{a,b,c}(X')$. So $Y \xrightarrow{\alpha+n[c]} Y'$.

PF_s:

Let $X \xrightarrow{n[a]^-} X''' \xrightarrow{n[a]^+} X'' \xrightarrow{\alpha} X'$ for some $n \geq 0$. So there are $A, B \in \mathbb{B}(P_X)$ and $E \in \mathbb{B}(T_X)$ such that $X'' = B \triangleright (A \triangleleft X)$ and $X' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleleft X'')$ and $\bar{L}_X(A) = \bar{L}_X(B) = n[a]$ and $\hat{L}_X(E) = \alpha$. By Definition 3.3, we have that $Y = \varphi_a(X) = \varphi_a(X'')$, that $T_Y = T_X$ and that $Y' = \tilde{O}_Y(E) \triangleright (\tilde{I}_Y(E) \triangleleft Y)$ satisfies $Y' = \varphi_a(X')$. So $\varphi_a(X) \xrightarrow{\alpha} \varphi_a(X')$. \square

We now move to the second proposition:

If $X \xrightarrow{e} X'$ for some closed net term X , there exists a rule $P(\vdash \longrightarrow) \vdash C(\vdash \longrightarrow)$ in Table 2 such that $C(\longrightarrow)$ equals $X \xrightarrow{e} X'$ and $P(\longrightarrow)$ holds.

Proof: Again we use case analysis for each rule type and Definitions 3.3 and 4.1. Note that the term X and event e determine what rule type should be used.

AT:

If X is an atom and $X \xrightarrow{e} X'$, the existence of a rule in the AT row of Table 2 is immediate from the two definitions.

ME:

If $X = U \parallel V$ and $X \xrightarrow{\alpha} X'$, then there exists $E \in \mathbb{B}(T_U \cup T_V)$ such that $X' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleleft X)$ and $\hat{L}_X(E) = \alpha$. We can write $E = F + G$, where $F \in \mathbb{B}(T_U)$, $G \in \mathbb{B}(T_V)$. Since U and V have disjoint nodes, we know that $\tilde{O}_U(F) + \tilde{O}_V(G) = \tilde{O}_X(E)$ and $\tilde{I}_U(F) + \tilde{I}_V(G) = \tilde{I}_X(E)$ and so there exists $\beta = \hat{L}_U(F)$ and $\gamma = \hat{L}_V(G)$ such that $U \xrightarrow{\beta} U'$ and $V \xrightarrow{\gamma} V'$ and $X' = U' \parallel V'$. This corresponds to the MEs rule of Table 2. The MEr rule is similar.

CO_r,PR_r,TF_r:

If $X = f(Y)$, where $f \in \{\gamma_{a,b}, \pi_{a,b}, \phi_{a,b,c}\}$ and $X \xrightarrow{\alpha^-} X'$, then there exists a bag $A \in \mathbb{B}(P_X)$ such that $X' = A \triangleleft X$ and $[t] \not\prec \bar{L}_X(A)$. Then by Definition 3.3, $P_X = P_Y$ and $Y' = A \triangleleft Y$ exists and $X' = f(Y')$ and since $L_Y(p) = L_X(p)$ for all $p \in P_X$, we have indeed $Y \xrightarrow{\alpha^-} Y'$.

PF_r:

If $X = \varphi_a(Y)$ and $X \xrightarrow{\alpha^-} X'$, then there exists a bag $A \in \mathbb{B}(P_X)$ such that $X' = A \triangleleft X$ and $[t] \not\prec \bar{L}_X(A)$. Let $g \in P_Y \rightarrow P_X$ be defined by $g(p) = p$ if $L_Y(p) \neq a$ and $g(p) = \{q \mid L_Y(q) = a\}$ if $L_Y(p) = a$ and let $h = \bar{g}$. By Definition 3.3, there exists a bag $B \in \mathbb{B}(P_Y)$ with $h(B) = A$ such that $Y' = B \triangleleft Y$ exists and $X' = f(Y')$. Since $L_Y(B) = L_X(A)$, we have indeed $Y \xrightarrow{\alpha^-} Y'$.

RE:

If $X = \rho_f(Y)$ and $X \xrightarrow{\alpha^-} X'$, then there exists a bag $A \in \mathbb{B}(P_X)$ such that $X' = A \triangleleft X$ and $\bar{L}_X(A) = \alpha$. Of course, $[t] \not\prec \alpha$. Thus, $Y' = A \triangleleft Y$ exists and $(\bar{f} \circ L_Y)(A) = \alpha$. So $Y \xrightarrow{\bar{f}(\alpha)^-} Y'$ and $[t] \not\prec \bar{f}(\alpha)$. The REs rule is similar.

CO_s,PR_s:

If $X = \gamma_{a,b}Y$ and $X \xrightarrow{\alpha} X'$, then there exists a bag $E \in \mathbb{B}(T_X)$ such that $X' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleleft X)$ and $\hat{L}_X(E) = \alpha$. By Definition 3.3, $P_X = P_Y$, $M_X = M_Y$ and $Y' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleright Y)$ exists. We can write $E = E_s + E_d$ where $E_s \in \mathbb{B}(T_X \cap T_Y)$ and $E_d \in \mathbb{B}(T_X \setminus T_Y)$. The transitions in $T_X \setminus T_Y$ are pairs (t, p) and we define $\phi \in T_X \rightarrow T_Y$ by $\phi(t) = t$ if $t \in T_Y$ and $\phi(t, p) = t$ if $(t, p) \notin T_Y$. Let $F = \tilde{\phi}(E)$. We have $\hat{L}_Y(F) = \hat{L}_X(E) = \alpha$ and $\tilde{O}_X(E) = \tilde{O}_Y(F)$ and $\tilde{I}_X(E) = \tilde{I}_Y(F) + \Sigma_I[p_i]$, where $\hat{L}_X(\Sigma_I[p_i]) = \alpha(b)[a]$. So $Y'' = \Sigma_I[p_i] \triangleleft Y$ exists and $\tilde{O}_Y(F) \triangleright (\tilde{I}_Y(F) \triangleleft Y'') = Y'$. By using Definition 4.1, we conclude that $Y \xrightarrow{\alpha(b).[a]^-} Y'' \xrightarrow{\alpha} Y'$. The production step rule is analogous.

TF_s:

If $X = \phi_{a,b,c}Y$ and $X \xrightarrow{\beta^-} X'$, then there exists a bag $E \in \mathbb{B}(T_X)$ such that $X' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleleft X)$ and $\hat{L}_X(E) = \beta$. By Definition 3.3, $P_X = P_Y$, $M_X = M_Y$ and $Y' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleright Y)$ exists. We can write $E = E_s + E_d$ where $E_s \in \mathbb{B}(T_X \cap T_Y)$ and $E_d \in \mathbb{B}(T_X \setminus T_Y)$. The transitions in $T_X \setminus T_Y$ are pairs (t, u) and we define $\phi \in T_X \rightarrow [(T_Y)]$ by $\phi(t) = [t]$ if $t \in T_Y$ and $\phi(t, u) = [t, u]$ if $(t, u) \notin T_Y$. Let $F = \tilde{\phi}(E)$. We have $\tilde{O}_X(E) = \tilde{O}_Y(F)$ and likewise for I . So $Y' = \tilde{O}_Y(F) \triangleright (\tilde{I}_Y(F) \triangleright Y)$ exists and $\hat{L}_Y(F) = \hat{L}_X(E_s) + n[ab]$, where n equals the number of transitions in E_d . Thus β can be written as $\alpha + n[c]$ and since X cannot contain any a - or b -labeled transitions, we have $Y \xrightarrow{\alpha+n[ab]} Y'$ with $\alpha(a) = \alpha(b) = 0$.

PF_s:

If $X = \varphi_aY$ and $X \xrightarrow{\alpha^-} X'$, then there exists a bag $E \in \mathbb{B}(T_X)$ such that $X' = \tilde{O}_X(E) \triangleright (\tilde{I}_X(E) \triangleleft X)$ and $\hat{L}_X(E) = \beta$. By Definition 3.3, $T_X = T_Y$, $\bar{L}_X(\tilde{I}_X(E)) = \bar{L}_Y(\tilde{I}_Y(E))$, $L_X(\tilde{O}_X(E)) = \bar{L}_Y(\tilde{O}_Y(E))$ and $\tilde{I}_Y(E) = A + B$, where A contains only a -labeled places (so $L_Y(A) = m[a]$) and B all the other places. So $I_Y(E) \triangleleft (A \triangleright Y)$ exists and $\bar{L}_Y(A + M_Y - I_Y(E)) = m[a] + \bar{L}_X(M_X - I_X(E))$.

So there exists a $C \in \mathbb{B}(Y)$ with $L_Y(C) = m[a]$ such that $(C + I_Y(E)) \triangleleft (A \triangleright Y)$ exists. Let D be the largest bag such that $D \leq A$ and $D \leq C$ and let $A' = A - D$, $C' = C - D$, $n[a] = L_Y(A') = L_Y(C')$, $Y' = O_Y(E) \triangleright ((C + I_Y(E)) \triangleleft (A \triangleright Y))$. Then $(C + I_Y(E)) \triangleleft (A \triangleright Y) = I_Y(E) \triangleleft (A' \triangleright (C' \triangleleft Y))$ and $Y \xrightarrow{n[a]^-} C' \triangleleft Y \xrightarrow{n[a]^+} A' \triangleright (C' \triangleleft Y) \xrightarrow{\alpha} Y'$. This completes the proof. \square