UNIVERSITÀ CA' FOSCARI DI VENEZIA
Dipartimento di Informatica
Technical Report Series in Computer Science

Rapporto di Ricerca CS-2006-9

Novembre 2006

P. Baldan, A. Corradini, H. Ehrig, R. Heckel, B. König

Bisimilarity and Behaviour-Preserving Reconfigurations of Open Petri Nets

Dipartimento di Informatica, Università Ca' Foscari di Venezia
Via Torino 155, 30172 Mestre–Venezia, Italy

Bisimilarity and Behaviour-Preserving Reconfigurations of Open Petri Nets*

Paolo Baldan[1], Andrea Corradini[2], Hartmut Ehrig[3], Reiko Heckel[4], and Barbara König[5]

[1] Dipartimento di Informatica, Università Ca' Foscari di Venezia, Italy
[2] Dipartimento di Informatica, Università di Pisa, Italy
[3] Institut für Softwaretechnik und Theoretische Informatik, Technische Universität Berlin, Germany
[4] Department of Computer Science, University of Leicester, UK
[5] Institut für Informatik und Interaktive Systeme, Universität Duisburg-Essen, Germany

Abstract. We propose a framework for the specification of behaviour-preserving reconfigurations of systems modelled as Petri nets. The framework is based on open nets, a mild generalisation of ordinary Place/Transition nets suited to model open systems which might interact with the surrounding environment and endowed with a colimit-based composition operation. We show that natural notions of (strong and weak) bisimilarity over open nets are congruences with respect to the composition operation. We also provide an up-to technique for facilitating bisimilarity proofs. The theory is used to identify suitable classes of reconfiguration rules (in the double-pushout approach to rewriting) whose application preserves the observational semantics of the net.

Introduction

Petri nets are a well-known model of concurrent and distributed systems, widely used both in theoretical and applicative areas [17]. In classical approaches nets represent closed, completely specified systems evolving autonomously through the firing of transitions. However, ordinary Petri nets are not adequate to model *open* systems, namely systems which can interact with the surrounding environment or, in a different view, systems which are only partially specified.

Firstly, in this setting a large (possibly still open) system is typically built out of smaller open components. Syntactically, an open system is characterised by the description of suitable interfaces, over which the interaction with the external environment can happen. Semantically, openness can be represented by defining the behaviour of the component as if it were embedded in general environments, determining any possible interaction over the interfaces.

Secondly, often the building components of an open system are not statically determined, but they can change during the evolution of the system, according to predefined reconfiguration rules triggered by internal or external solicitations.

* Research partially supported by the EC RTN 2-2001-00346 SegraVis, the EU IST-2004-16004 SEnSOria the MIUR Project ART and the DFG project SANDS.

In this paper we provide a framework where open systems can be modelled as Petri nets and suitable reconfigurations of such systems can be specified, which preserve the behaviour of the system. The framework is based on so-called *open nets*, a mild generalisation of ordinary Petri nets introduced in [2, 3] to answer the first of the requirements above, i.e., the possibility of interacting with the environment and of composing a larger net out of smaller open components. An open net is an ordinary net with a distinguished set of places, designated as open, through which the net can interact with the surrounding environment. As a consequence of such interaction, tokens can be freely generated and removed in open places. In the mentioned papers open nets are endowed with a composition operation, characterised as a pushout in the corresponding category, suitable to model both interaction through open places and synchronisation of transitions. A deterministic process semantics *à la* Goltz-Reisig is shown to be compositional with respect to such composition operation.

In the following sections we first introduce *marked* open nets, i.e., open Petri nets with a distinguished initial marking. The existing theory of open nets, including the pushout-based composition operation, is extended to the marked case and a compositionality result for step sequences is proved in this context.

Next we introduce bisimulation-based observational equivalences for open Petri nets. Following the intuition about reactive systems discussed in [9], such equivalences are based on the observation of the interactions between the given net and the surrounding environment. The framework treats uniformly *strong bisimilarity*, where every transition firing is observed, and *weak bisimilarity*, where a subset of unobservable transition labels is fixed and the firings of transitions carrying such labels (corresponding to τ transitions) are considered invisible. Bisimilarity is shown to be a congruence with respect to the composition operation over open nets. Interestingly enough, this holds also when the set of non-observable events is not empty, i.e., for weak bisimilarity (some natural questions regarding the relation with weak bisimilarity in CCS are addressed in an Appendix). In addition, we also define an up-to technique for facilitating bisimulation proofs.

Exploiting the results in the first part of the paper we introduce next a framework for open Petri net reconfigurations. The fact that open net components are combined by means of categorical colimits, naturally suggests a mathematical framework for specifying net reconfigurations, based on double-pushout (DPO) rewriting [6]. Using the congruence result for bisimilarity we identify classes of transformation rules which ensure that reconfigurations of the system do not affect its observational behaviour.

A concluding section discusses some related work, including the results on strong bisimilarity for Petri nets presented in [13, 18] and the work on reconfigurations of ordinary Petri nets by means of rewriting rules in [1, 10, 16].

1 Marked open nets

An *open net*, as introduced in [2, 3], is an ordinary P/T Petri net with a distinguished set of places. These places are intended to represent the interface of the net towards the environment, which can interact with the net by adding or removing some tokens in the open places. Concretely, an open place can be an *input* or an *output* place (or both), meaning that the environment can put or remove tokens from that place. In this section we introduce the basic notions for open nets as presented in [3], generalising them to nets with initial marking.

Given a set X we write $\mathbf{2}^X$ for the powerset of X and $X^{\oplus}$ for the free commutative monoid over X. Moreover, given a function $h : X \to Y$ we denote by the same symbol $h : \mathbf{2}^X \to \mathbf{2}^Y$ its extension to sets and by $h^{\oplus} : X^{\oplus} \to Y^{\oplus}$ its monoidal extension. Given a multiset $u \in X^{\oplus}$, with $u = \bigoplus_{x \in X} u_x \cdot x$, for $x \in X$ we will write $u(x)$ to denote the coefficient u_x. The symbol 0 denotes the empty multiset.

Definition 1 (multiset projection). *Given a function $f : X \to Y$ and a multiset $u \in Y^{\oplus}$ we denote by $(u \downarrow f)$ the projection of u along f, which is the multiset over X defined as $(u \downarrow f) = \bigoplus_{x \in X} u_{f(x)} \cdot x$.*

In other words, $(\_ \downarrow f) : Y^{\oplus} \to X^{\oplus}$ is the monoidal extension of the function $(\_ \downarrow f) : Y \to X^{\oplus}$ defined by $(y \downarrow f) = x_1 \oplus \ldots \oplus x_n$ when $f^{-1}(y) = \{x_1, \ldots, x_n\}$. In the following we will mainly work with injective functions, for which the projection operation satisfies some expected properties, such as $f^{\oplus}((u \downarrow f)) \leq u$ and $(f^{\oplus}((u \downarrow f)) \downarrow f) = (u \downarrow f)$.

We consider nets where transitions are labelled over a fixed set of labels Λ.

Definition 2 (P/T Petri net). *A* P/T Petri net *is a tuple $N = (S, T, \sigma, \tau, \lambda)$ where S is the set of places, T is the set of transitions (with $S \cap T = \emptyset$), $\sigma, \tau : T \to S^{\oplus}$ are functions mapping each transition to its pre- and post-set and $\lambda : T \to \Lambda$ is a labelling function for transitions.*

In the following we will denote by ${}^{\bullet}(\cdot)$ and $(\cdot)^{\bullet}$ the monoidal extensions of the functions σ and τ to functions from $T^{\oplus}$ to $S^{\oplus}$. Furthermore, given a place $s \in S$, the pre- and post-set of s are defined by ${}^{\bullet}s = \{t \in T : s \in t^{\bullet}\}$ and $s^{\bullet} = \{t \in T : s \in {}^{\bullet}t\}$.

Definition 3 (Petri net category). *Let N_0 and N_1 be Petri nets. A* Petri net morphism *$f : N_0 \to N_1$ is a pair of total functions $f = \langle f_T, f_S \rangle$ with $f_T : T_0 \to T_1$ and $f_S : S_0 \to S_1$, such that for all $t_0 \in T_0$, ${}^{\bullet}f_T(t_0) = {f_S}^{\oplus}({}^{\bullet}t_0)$, $f_T(t_0)^{\bullet} = {f_S}^{\oplus}(t_0{}^{\bullet})$ and $\lambda_1(f_T(t_0)) = \lambda_0(t_0)$. The category of P/T Petri nets and Petri net morphisms is denoted by* **Net**.

Recall that category **Net** is a subcategory of the category **Petri** of [11], which has the same objects, but more general morphisms which can map a place to a multiset of places.

We next introduce the notion of open net. As anticipated above, differently from [2, 3], we work here with marked nets. This will be used in the treatment of bisimilarity.

Definition 4 (open net). *An* open net *is a pair $Z = (N_Z, O_Z)$, where $N_Z = (S_Z, T_Z, \sigma_Z, \tau_Z, \lambda_Z)$ is a P/T Petri net and $O_Z = (O_Z^+, O_Z^-) \in \mathbf{2}^{S_Z} \times \mathbf{2}^{S_Z}$ are the sets of input and output open places of the net. A* marked *open net is a pair $(Z, \hat{u})$ where Z is an open net and $\hat{u} \in {S_Z}^{\oplus}$ is the initial marking.*

Hereafter, unless stated otherwise, all open nets will be implicitly assumed to be marked. An open net will be denoted simply by Z and the corresponding initial marking by $\hat{u}$. Subscripts carry over to the net components. The graphical representation for open nets is similar to that for standard nets. In addition, the fact that a place is input or output open is represented by an ingoing or outgoing dangling arc, respectively. For instance, in net Z_1 of Fig. 3, place s is both input and output open, while s' is only output open.

The notion of enabledness for transitions is the usual one, but, besides the changes produced by the firing of the transitions of the net, we consider also the interaction with the environment which is modelled by actions, denoted by $+_s$ and $-_s$, which produce or consume a token in an open place s.

Definition 5 (set of extended events). *Let Z be an open net. The* set of extended events *of Z, denoted by $\bar{T}_Z$ and ranged over by ϵ is defined as*

$$\bar{T}_Z = T_Z \cup \{+_s : s \in O_Z^+\} \cup \{-_s : s \in O_Z^-\}.$$

Defining ${}^\bullet{+_s} = 0$ and ${+_s}^\bullet = s$, and symmetrically, ${}^\bullet{-_s} = s$ and ${-_s}^\bullet = 0$, the notion of pre- and post-set extends in the obvious way to multisets of extended events.

We are now able to define firing and steps in an open net. Given a marking $u \in S^\oplus$ we will denote by $+_u$ the multiset $\bigoplus_{s \in S} u(s) \cdot +_s$ (provided that u contains only input open places). Similarly $-_u = \bigoplus_{s \in S} u(s) \cdot -_s$.

Definition 6 (firings and steps). *Let Z be an open net. A* step *in Z consists of the execution of a multiset of (extended) events $A \in \bar{T}_Z^\oplus$, i.e.,*

$$u \oplus {}^\bullet A \; [A\rangle \; u \oplus A^\bullet.$$

A step is called a firing *when it consists of a single event, i.e., $A = \epsilon \in \bar{T}_Z$.*

Note that a firing can be (i) the execution of a transition $u \oplus {}^\bullet t \; [t\rangle \; u \oplus t^\bullet$, with $u \in {S_Z}^\oplus$, $t \in T_Z$; (ii) the creation of a token by the environment $u \; [+_s\rangle \; u \oplus s$, with $s \in O_Z^+$, $u \in {S_Z}^\oplus$; (iii) the deletion of a token by the environment $u \oplus s \; [-_s\rangle \; u$, with $u \in {S_Z}^\oplus$, $s \in O_Z^-$. A step consists of the firing of a multiset of transitions and of interactions with the environment, i.e., it is of the kind $A \oplus -_w \oplus +_v$ for $w, v \in {S_Z}^\oplus$.

Definition 7 (open nets category). *An* open net morphism *$f : Z_1 \to Z_2$ is a Petri net morphism $f : N_{Z_1} \to N_{Z_2}$ such that, if we define $\mathsf{in}(f) = \{s \in S_1 : {}^\bullet f_S(s) - f_T({}^\bullet s) \neq \emptyset\}$ and $\mathsf{out}(f) = \{s \in S_1 : f_S(s)^\bullet - f_T(s^\bullet) \neq \emptyset\}$, then*

1. *(i) $f_S^{-1}(O_2^+) \cup \mathsf{in}(f) \subseteq O_1^+$ and (ii) $f_S^{-1}(O_2^-) \cup \mathsf{out}(f) \subseteq O_1^-$.*
2. *$\hat{u}_1 = (\hat{u}_2 \downarrow f)$ (*reflection of initial marking*).*

The morphism f is called an open net embedding *if both f_T and f_S are injective. We will denote by* **ONet** *the category of open nets and open net morphisms.*

Intuitively, an embedding $f : Z_1 \to Z_2$ "inserts" net Z_1 into a larger net Z_2, which constrains the behaviour of Z_1. Conditions 1.(i) and 1.(ii) first require that open places are reflected and hence that places which are "internal" in Z_1 cannot be promoted to open places in Z_2. Furthermore, the context in which Z_1 is inserted can interact with Z_1 only through the open places. To understand how this is formalised, observe that $\mathsf{in}(f)$ includes all the places of Z_1 whose image $f_S(s)$ is in the post-set of a "new" transition, where "new" means not in the image of Z_1. Intuitively we can think that in Z_2 new transitions are attached to s and can produce tokens in such place. This is the reason why condition 1.(i) also asks any place in $\mathsf{in}(f)$ to be an input open place of Z_1. Condition 1.(ii) is analogous for output places. Consistently with the intuition that f inserts Z_1 into a larger context represented by Z_2, condition 2 requires that the marking of Z_1 is the projection of the marking of Z_2: any place $s_1 \in S_1$ must carry the same number of tokens of its image $f(s_1) \in S_2$, i.e., $u_1(s_1) = u_2(f(s_1))$ for any $s_1 \in S_1$. All morphisms f_1, f_2, α_1 and α_2 in Fig. 3 are examples of open net embeddings (the mappings on places and transitions are those suggested by the shape and labelling of the nets). Consider, for instance, morphism $f_1 : Z_0 \to Z_1$. Note that in Z_1 a "new" c-labelled transition is attached to the places s and s'. This is legal since the corresponding places in Z_0 are output open and input open, respectively, in Z_1. Note also that the number of tokens in places in Z_0 and in their image through f_1 is the same. Instead, the number of tokens in the place s'' in Z_1 is not constrained since it is not in the image of f_1: the place is marked, but f_1 would have been a legal morphism also if s'' were not marked.

It is worth observing that most of the constructions in the paper will be defined for open net embeddings, hence readers can limit their attention to embeddings if this helps the intuition. Still, on the formal side, working in a larger host category with more general morphisms is essential to obtain a characterisation of the composition operation in terms of pushouts. Specifically, non-injective open net morphisms are needed as mediating morphisms (recall that the category of sets with injective functions does not have pushouts).

Observe that the constraints characterising open nets morphisms have an intuitive graphical interpretation:

- The connections of transitions to their pre-set and post-set have to be preserved. New connections cannot be added.
- In the larger net, a new arc may be attached to a place only if the corresponding place of the subnet has a dangling arc in the same direction. Dangling arcs may be removed, but cannot be added in the larger net.
- The number of tokens in each place in the source net must be preserved in the target. Instead, there are no restrictions on the marking of places of the target net which are not in the image of the source net.

In the sequel, given an open net morphism $f = \langle f_S, f_T \rangle : Z_1 \to Z_2$, to lighten the notation we will omit the subscripts "S" and "T" in its place and transition components, writing $f(s)$ for $f_S(s)$ and $f(t)$ for $f_T(t)$. Moreover we will write $f^\oplus : \bar{T}^\oplus_{Z_1} \to \bar{T}^\oplus_{Z_2}$ to denote the monoidal function defined on the generators by $f^\oplus(t_1) = f(t_1)$ and $f^\oplus(x_{s_1}) = x_{f(s_1)}$ for $x \in \{+, -\}$.

The next proposition explicitly shows that category **ONet**, as introduced in Definition 7, is well defined. To prove this fact we will use the well-definedness of the category of non marked open nets, introduced in [3]. This category, denoted here by $\mathbf{ONet}^u$ has (unmarked) open nets as objects and mappings satisfying only condition 1 in Definition 7 as morphisms. These will be referred to as *unmarked open net morphisms.*

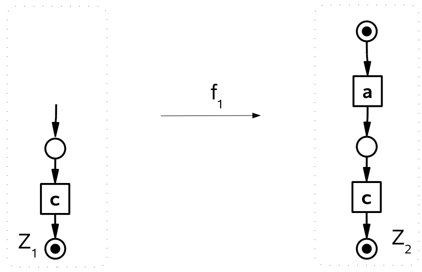


Fig. 1. Open net morphisms are not simulations.

Proposition 1. *Open net morphisms are closed under composition.*

Proof. Let $f_1 : Z_1 \to Z_2$ and $f_2 : Z_2 \to Z_3$ be open net morphisms. Then f_1 and f_2 are unmarked open net morphisms and thus, since $\mathbf{ONet}^u$ is a well-defined category, also $f_2 \circ f_1$ is an unmarked open net morphism. In order to prove that $f_2 \circ f_1$ is a well defined open net morphism it remains to show that it satisfies also condition 2 in Definition 7, i.e., that it reflects the initial marking. But this fact follows easily from the definition. In fact, for any $s_1 \in S_1$, Therefore

$$\begin{aligned} & \hat{u}_3(f_2(f_1(s_1))) = \text{[since } f_2 \text{ is an open net morphism]} \\ & \quad = \hat{u}_2(f_1(s_1)) \ \ \text{[since } f_1 \text{ is an open net morphism]} \\ & \quad = \hat{u}_1(s_1) \end{aligned}$$

□

Note that, different from most of the morphisms considered over Petri nets in the literature, open net morphisms are *not* simulations. Instead, since open net embeddings are designed to capture the idea of inserting a net into a larger one, they are expected to reflect the behaviour, in the sense that given an embedding $f : Z_0 \to Z_1$, the behaviour of Z_1 can be projected along f to the behaviour of Z_0. This is consistent with the fact that we perform system composition by colimits: any component will have a morphism into the full system and we can not expect that the larger system is able to simulate a (less specified) component of itself. The target net of a morphism is in general more "instantiated" and thus more constrained than the source net (e.g., a place which is open in the source net can be closed in the target).

As an example, consider the open net embedding in Fig. 1. While the transition labelled c in the net N_1 can fire infinitely many times, its image in the second net N_2 can fire only once.

Another possibility would be to perform system composition via limits and to use morphisms that *are* simulations (similar to the morphisms considered in [19]). But doing composition via colimits is usually more straightforward and gives us more control over how places should be attached.

To formalise reflection of the behaviour along open nets embeddings, we define the projection operation also over steps.

Definition 8 (projecting extended events). *Given an open net embedding $f : Z \to Z'$ and an extended event $\epsilon' \in \bar{T}_{Z'}$ we define the* projection of ϵ' along f *as follows:*

- *if $\epsilon' = t' \in T_{Z'}$ is a transition then*
$$(t' \Downarrow f) = \begin{cases} t & \text{if } t \in T_Z \text{ and } f(t) = t' \\ -_{({}^\bullet t' \downarrow f)} \oplus +_{(t'^\bullet \downarrow f)} & \text{if } t' \notin f(T_Z) \end{cases}$$
- *if $\epsilon' = x_{s'}$, with $x \in \{+, -\}$, is an interaction with the environment, then $(x_{s'} \Downarrow f) = x_{(s' \downarrow f)}$.*

The projection operation over multisets of extended events $(\_ \Downarrow f) : \bar{T}^{\oplus}_{Z'} \to \bar{T}^{\oplus}_{Z}$, is defined as the monoidal extension of the projection of firings.

In words, if we think of the embedding as an inclusion, given a transition t', the projection $(t' \Downarrow f)$ is the transition itself if t' is in Z. Otherwise, if t' is not in Z but it consumes or produces tokens in places of Z, the projection of t' contains the corresponding extended events, expressing the interactions over open places. The projection of an extended event $+_{s'}$ is the empty multiset, if $s' \notin f(S_Z)$ and it is the same extended event if, instead, $s' \in f(S_Z)$.

It is worth noticing that the step produced by the projection operation is well-defined, in the sense that, e.g., if $+_s \in (\epsilon \Downarrow f)$ then $s \in O^+_Z$. In fact, if $+_s \in (t' \Downarrow f)$ then $s \in \mathsf{in}(f) \subseteq O^+_Z$. If, $+_s \in (+_{s'} \Downarrow f)$ this means that $s' \in O^+_{Z'}$ and thus, since $f(s) = s'$, by definition of open net morphism, $s \in O^+_Z$.

The next lemma express some properties of the projection operator over multisets and over steps.

Lemma 1 (properties of projection). *Let $f : Z \to Z'$ be an open net embedding. Then*

1. *for $x'_1, x'_2 \in {S_{Z'}}^{\oplus}$ we have*
$$((x_1 \oplus x_2) \downarrow f) = (x_1 \downarrow f) \oplus (x_2 \downarrow f) \quad \text{and} \quad (0 \downarrow f) = 0$$
and for $x \in {S_Z}^{\oplus}$
$$(f^{\oplus}(x) \downarrow f) = x$$
2. *for $x'_1, x'_2 \in \bar{T}^{\oplus}_{Z'}$ we have*
$$((x_1 \oplus x_2) \Downarrow f) = (x_1 \Downarrow f) \oplus (x_2 \Downarrow f) \quad \text{and} \quad (0 \Downarrow f) = 0$$
and for $x \in {S_Z}^{\oplus}$
$$(f^{\oplus}(x) \Downarrow f) = x$$
3. *given $A' \in \bar{T}_{Z'}$*
$$(({}^\bullet A') \downarrow f) = {}^\bullet(A' \Downarrow f) \quad \text{and} \quad ((A'^\bullet) \downarrow f) = (A' \Downarrow f)^\bullet$$
4. *for $u \in {S_{Z'}}^{\oplus}$ we have*
$$f^{\oplus}((u' \downarrow f)) \leq u$$

Proof. We prove only the third point. Since ${}^\bullet(\cdot)$ and $(\cdot)^\bullet$ are monoidal functions it is sufficient to prove the result only on the generators. We concentrate on ${}^\bullet(\cdot)$, since the proof for $(\cdot)^\bullet$ is completely analogous.

We distinguish various cases:

- $A' = t' \in T_{Z'}$
  If there exists $t \in T_Z$ such that $f(t) = t'$, then $(t' \Downarrow f) = t$. Since f is an open net morphism $f^{\oplus}({}^\bullet t) = {}^\bullet t'$ and thus, as desired
  $${}^\bullet(t' \Downarrow f) = {}^\bullet t = (f^{\oplus}({}^\bullet t) \downarrow f) = (({}^\bullet t') \downarrow f)$$
  where the second equality is justified by point (2).
  If, instead, $t' \notin f(T_Z)$ we have that $(t' \Downarrow f) = -_{({}^\bullet t' \downarrow f)} \oplus +_{(t'^\bullet \downarrow f)}$. Hence, in this case the result is obvious since
  $${}^\bullet(t' \Downarrow f) = {}^\bullet(-_{({}^\bullet t' \downarrow f)} \oplus +_{(t'^\bullet \downarrow f))} = (({}^\bullet t') \downarrow f) = (({}^\bullet t') \downarrow f)$$
- $A' = +_{s'}$ or $A' = -_{s'}$

  Suppose, e.g., that $A' = -_{s'}$. In this case $(A' \Downarrow f) = -_{(s' \downarrow f)}$ and the result trivially holds. □

Lemma 2 (reflection of behaviour). *Let $f : Z \to Z'$ be an open net embedding. For every step $u'\ [A'\rangle\ v'$ in Z' there is a step $(u' \downarrow f)\ [(A' \Downarrow f)\rangle\ (v' \downarrow f)$ in Z, called the* projection *of the step $u'\ [A'\rangle\ v'$ over Z.*

Proof. Let $f : Z \to Z'$ be an open net embedding and assume that $u'\ [A'\rangle\ v'$ is a step in Z'. Therefore

$$u' = u'' \oplus {}^\bullet A' \quad \text{and} \quad v' = u'' \oplus A'^\bullet$$

Now, we have

$$\begin{aligned}(u' \downarrow f) = & & \text{[by Lemma 1.(1)]}\\ = (u'' \downarrow f) \oplus (({}^\bullet A') \downarrow f) & & \text{[by Lemma 1.(3)]}\\ = (u'' \downarrow f) \oplus {}^\bullet(A' \Downarrow f) & &\end{aligned}$$

and similarly

$$(v' \downarrow f) = (u'' \downarrow f) \oplus (A' \Downarrow f)^\bullet$$

Therefore, as desired, there is the step

$$(u' \downarrow f) = (u'' \downarrow f) \oplus {}^\bullet(A' \Downarrow f)\ [(A' \Downarrow f)\rangle\ (u'' \downarrow f) \oplus (A' \Downarrow f)^\bullet = (v' \downarrow f).$$

□

Observe that there is an obvious forgetful functor $\mathcal{F} : \mathbf{ONet} \to \mathbf{Net}$, defined by $\mathcal{F}(Z) = N_Z$ and $\mathcal{F}(f : Z_0 \to Z_1) = f : N_{Z_0} \to N_{Z_1}$. Since functor $\mathcal{F}$ acts on arrows as identity, with abuse of notation, given an open net morphism $f : Z_0 \to Z_1$ we will often write $f : \mathcal{F}(Z_1) \to \mathcal{F}(Z_2)$ instead of $\mathcal{F}(f) : \mathcal{F}(Z_1) \to \mathcal{F}(Z_2)$.

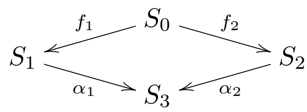
2 Composing open nets

We introduce next a basic mechanism for composing open nets which is characterised as a pushout construction in the category of open nets. The case of unmarked nets was already discussed in [3]. Here we extend the theory to deal with marked open nets. This will allow later to define reconfigurations of open nets, where the applicability of a reconfiguration rule can depend on the marking. Intuitively, two open nets Z_1 and Z_2 are composed by specifying a common subnet Z_0, and then by joining the two nets along Z_0. The categorical characterisation will be useful for proving the results about behaviour preserving transformations in Section 4, but the concrete characterisation in terms of a join along a common subnet can help the intuition.

Let us start with a technical definition which will be useful below.

Proposition 2 (composition of multisets). *Consider a pushout diagram in the category of sets as below, where all morphisms are injective.*

Given $u_1 \in S_1{}^{\oplus}$ and $u_2 \in S_2{}^{\oplus}$ such that $(u_1 \downarrow f_1) = (u_2 \downarrow f_2) = u_0$, then there is a (unique) multiset $u_3 \in S_3{}^{\oplus}$ such that $(u_3 \downarrow \alpha_i) = u_i$, for $i \in \{1, 2\}$. Such a marking u_3 will be denoted by $u_3 = u_1 \uplus_{u_0} u_2$.



Additionally, if $u_3 = u_1 \uplus_{u_0} u_2$ and $u_3' = u_1' \uplus_{u_0'} u_2'$, then $u_3 \oplus u_3' = (u_1 \oplus u_1') \uplus_{(u_0 \oplus u_0')} (u_2 \oplus u_2')$.

Proof. Define $u_3 = \bigoplus_{s_3 \in S_3} \max\{u_1(\alpha_1^{-1}(s_3)), u_2(\alpha_2^{-1}(s_3))\} \cdot s_3$.[6] In order to prove that $(u_3 \downarrow \alpha_1) = u_1$, notice that, since f_1 is an embedding, this amount to show that for any $s_1 \in S_1$ we have $u_1(s_1) = u_3(\alpha_1(s_1))$. Now that for $s_1 \in S_1$, we have have two cases:

- $s_1 = f_1(s_0)$
  Then, since by hypothesis $(u_1 \downarrow f_1) = u_0 = (u_2 \downarrow f_2)$ we have
  $$\begin{aligned} u_3(\alpha_1(s_1)) &= \max\{u_1(f_1(s_0)), u_2(f_2(s_0))\} \\ &= \max\{u_0(s_0), u_2(f_2(s_0))\} \\ &= u_0(s_0) \\ &= u_1(s_1) \end{aligned}$$
- s_1 not in the image of $f_1(s_0)$
  In this case we have:
  $$\begin{aligned} u_3(\alpha_1(s_1)) &= \max\{u_1(s_1), u_2(\alpha_2^{-1}(\alpha_1(s_1)))\} \\ &= \max\{u_1(s_1), u_2(\emptyset)\} \\ &= u_1(s_1) \end{aligned}$$

The uniqueness of u_3 is immediate, since α_1 and α_2 are jointly epi.

Concerning the second part of the statement, let $u_3 = u_1 \uplus_{u_0} u_2$ and $u_3' = u_1' \uplus_{u_0'} u_2'$. Then just observe that by Lemma 1.(1), we have for $i \in \{1, 2\}$

$$(u_3 \oplus u_3' \downarrow \alpha_i) = (u_3 \downarrow \alpha_i) \oplus (u_3' \downarrow \alpha_i) = u_i$$

[6] Recall that the α_i are injective and note that we are abusing the notation considering $u_i(\emptyset) = 0$ and $u_i(\{s_i\}) = u_i(s_i)$.

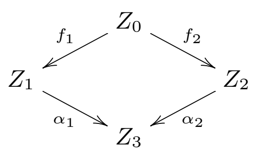


Fig. 2. Pushout in **ONet**.

hence the result $u_3 \oplus u'_3 = (u_1 \oplus u'_1) \uplus_{(u_0 \oplus u'_0)} (u_2 \oplus u'_2)$, follows by the defining property of the sum of markings. □

Intuitively, the multiset $u_1 \uplus_{u_0} u_2$ can be seen as the "least upper bound" of the images of the multisets over S_1 and S_2.

As in [2, 3], we say that two embeddings $f_1 : Z_0 \to Z_1$ and $f_2 : Z_0 \to Z_2$ are *composable* if the places which are used as interface by f_1, namely the places $\mathsf{in}(f_1)$ and $\mathsf{out}(f_1)$, are mapped by f_2 to input and output open places of Z_2, respectively, and also the symmetric condition holds. If, and only if, these conditions hold the pushout of f_1 and f_2 can be computed in **Net** and then lifted to **ONet**.

Definition 9 (composability). *Let $f_1 : Z_0 \to Z_1$, $f_2 : Z_0 \to Z_2$ be embeddings in* **ONet** *(see Fig. 2). We say that f_1 and f_2 are* composable *if*

1. *$f_2(\mathsf{in}(f_1)) \subseteq O^+_{Z_2}$ and $f_2(\mathsf{out}(f_1)) \subseteq O^-_{Z_2}$;*
2. *$f_1(\mathsf{in}(f_2)) \subseteq O^+_{Z_1}$ and $f_1(\mathsf{out}(f_2)) \subseteq O^-_{Z_1}$.*

Proposition 3 (pushouts in ONet). *Let $f_1 : Z_0 \to Z_1$ and $f_2 : Z_0 \to Z_2$ be embeddings in* **ONet** *(see Fig. 2). Compute the pushout of the corresponding diagram in category* **Net** *(componentwise on places and transitions) obtaining the net N_{Z_3} and the morphisms α_1 and α_2, and then take as open places, for $x \in \{+, -\}$,*

$$O^x_{Z_3} = \{s_3 \in S_3 : \alpha_1^{-1}(s_3) \subseteq O^x_{Z_1} \ \wedge \ \alpha_2^{-1}(s_3) \subseteq O^x_{Z_2}\}$$

and as marking $u_3 = u_1 \uplus_{u_0} u_2$, defined according to Proposition 2. Then $(\alpha_1, Z_3, \alpha_2)$ is the pushout in **ONet** *of f_1 and f_2 if and only if f_1 and f_2 are composable. In this case we write $Z_3 = Z_1 +_{f_1, f_2} Z_2$.*

Proof. We know, that the above result holds for unmarked nets, i.e., in the category $\mathbf{ONet}^u$. Here we must additionally show that (i) α_i are marked morphisms and that (ii) if we take any other net Z'_3, with $\alpha'_i : Z_i \to Z'_3$ making the diagram commute, then the mediating morphism $\gamma : Z_3 \to Z'_3$ (which exists uniquely as an unmarked net morphism by the result in [3]) respects the condition on the marking.

Now, (i) is immediate since Proposition 2 tells us that $(\hat{u}_3 \downarrow \alpha_i) = \hat{u}_i$ for $i \in \{1, 2\}$. Property (ii) can be proved along the same lines. □

As an example, the open net embeddings f_1 and f_2 in Fig. 3 are composable and Z_3 is the resulting pushout object.

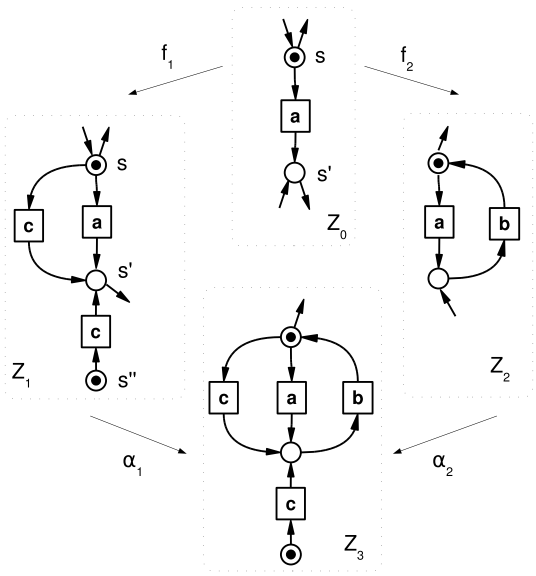


Fig. 3. An example of a pushout in **ONet**.

3 Composing steps

In this section we analyse the behaviour of an open net Z_3 arising as the composition of two open nets Z_1 and Z_2 along an interface net Z_0. More specifically, concentrating on steps, we show that steps of the component nets Z_1 and Z_2 can be "composed" to give rise to a step of Z_3 when they agree on the interface and satisfy suitable compatibility conditions.

We start with a technical lemma which will be pivotal in the paper.

Lemma 3. *Let $f_1 : Z_0 \to Z_1$ and $f_2 : Z_0 \to Z_2$ be composable embeddings in* **ONet** *and let $Z_3 = Z_1 +_{f_1,f_2} Z_2$ (see Fig. 2). Let $u_1\ [A_1\rangle\ v_1$ and $u_2\ [A_2\rangle\ v_2$ be steps in Z_1 and Z_2, respectively, such that $(u_1 \downarrow f_1) = (u_2 \downarrow f_2) = u_0$ and $A_2 = {f_2}^{\oplus}((A_1 \Downarrow f_1))$.*

Then, $(v_1 \downarrow f_1) = v_0 = (v_2 \downarrow f_2)$ and, if we define $A_3 = {\alpha_1}^{\oplus}(A_1)$,

$$u_1 \uplus_{u_0} u_2\ [A_3\rangle\ v_1 \uplus_{v_0} v_2.$$

Proof. First observe that, since $A_2 = {f_2}^{\oplus}((A_1 \Downarrow f_1))$, by Lemma 1.(2),

$$(A_2 \Downarrow f_2) = (A_1 \Downarrow f_1).$$

Let $A_0 = (A_i \Downarrow f_i)$ be the common projection.

By the above, obviously, we have ${}^{\bullet}(A_2 \Downarrow f_2) = {}^{\bullet}(A_1 \Downarrow f_1)$ and thus, by Lemma 1.(3)

$$(({}^{\bullet}A_1) \downarrow f_1) = (({}^{\bullet}A_2) \downarrow f_2)$$

so that we can consider the sum of markings ${}^\bullet A_1 \uplus_{{}^\bullet A_0} {}^\bullet A_2$. We claim that

$$ {}^\bullet A_3 = {}^\bullet A_1 \uplus_{{}^\bullet A_0} {}^\bullet A_2 \tag{1} $$

and similarly, $((A_1{}^\bullet)\downarrow f_1) = ((A_2{}^\bullet)\downarrow f_2)$ and

$$ A_3{}^\bullet = A_1{}^\bullet \uplus_{A_0{}^\bullet} A_2{}^\bullet $$

In fact, let us concentrate on ${}^\bullet(\cdot)$. We have that

$$\begin{aligned} (({}^\bullet A_3)\downarrow \alpha_1) = & \quad \text{[by Lemma 1.(3)]} \\ = {}^\bullet(A_3 \Downarrow \alpha_1) = & \quad \text{[by definition of } A_3\text{]} \\ = {}^\bullet(\alpha_1{}^\oplus(A_1)\Downarrow \alpha_1) = & \quad \text{[by Lemma 1.(2)]} \\ = {}^\bullet A_1 & \end{aligned}$$

Therefore, to conclude the validity of (1) we only need to show that also $({}^\bullet A_3 \downarrow \alpha_2) = {}^\bullet A_2$ (see Proposition 2). Now, we have

$$\begin{aligned} (({}^\bullet A_3)\downarrow \alpha_2) = & \quad \text{[by Lemma 1.(3)]} \\ = {}^\bullet(A_3 \Downarrow \alpha_2) = & \quad \text{[by definition of } A_3\text{]} \\ = {}^\bullet(\alpha_1{}^\oplus(A_1)\Downarrow \alpha_2) & \end{aligned}$$

Therefore we must get ${}^\bullet(\alpha_1{}^\oplus(A_1)\Downarrow\alpha_2) = {}^\bullet A_2$ and this is proved by showing

$$ (\alpha_1{}^\oplus(A_1)\Downarrow\alpha_2) = A_2 \tag{2} $$

To this aim, we proceed by induction on the cardinality of A_1:

- $A_1 = 0$
  In this case $A_2 = f_2{}^\oplus((A_1 \Downarrow f_1)) = 0 = (\alpha_1{}^\oplus(A_1)\Downarrow\alpha_2)$, as desired.
- $A_1 = t_1$
  We distinguish two subcases. If $(t_1 \Downarrow f_1) = t_0 \in T_{Z_0}$ then $A_2 = f_2(t_0) = (\alpha_1(t_1)\Downarrow \alpha_2)$, as desired, by construction of the pushout.
  If, instead, $(t_1 \Downarrow f_1) = -_{({}^\bullet t_1 \downarrow f_1)} \oplus +_{(t_1{}^\bullet \downarrow f_1)}$, then by hypothesis, $A_2 = f_2{}^\oplus((A_1 \Downarrow f_1))$ and thus
  $A_2 = -_{f_2{}^\oplus(({}^\bullet t_1 \downarrow f_1))} \oplus +_{f_2{}^\oplus((t_1{}^\bullet \downarrow f_1))}$
  On the other hand, we have
  $(\alpha_1{}^\oplus(A_1)\Downarrow\alpha_2) = (\alpha_1(t_1)\Downarrow\alpha_2) = -_{(({}^\bullet \alpha_1(t_1))\downarrow\alpha_2)} \oplus +_{(((\alpha_1(t_1))^\bullet\downarrow\alpha_2)}$
  Now, by exploiting the fact that Z_3 is a pushout, it is easy to see that $f_2{}^\oplus(({}^\bullet t_1 \downarrow f_1)) = (({}^\bullet\alpha_1(t_1))\downarrow\alpha_2)$ and similarly $f_2{}^\oplus((t_1{}^\bullet \downarrow f_1)) = ((\alpha_1(t_1))^\bullet \downarrow \alpha_2)$. Hence we conclude that $A_2 = (\alpha_1{}^\oplus(A_1)\Downarrow\alpha_2)$, as desired.
- $A_1 = +_{s_1}$ or $A_1 = -_{s_1}$
  Assume, for instance, that $A_1 = +_{s_1}$ (the other case is completely analogous). Therefore
  $A_2 = f_2{}^\oplus((A_1 \Downarrow f_1)) = +_{f_2{}^\oplus((s_1 \downarrow f_1))}$
  On the other hand
  $(\alpha_1{}^\oplus(A_1)\Downarrow\alpha_2) = (+_{\alpha_1(s_1)}\Downarrow\alpha_2) = +_{(\alpha_1(s_1)\downarrow\alpha_2)}$
  and, again, by the fact that Z_3 is a pushout, we deduce easily that $f_2{}^\oplus((s_1 \downarrow f_1)) = (\alpha_1(s_1)\downarrow\alpha_2)$, hence the desired equality.

– $A_1 = A_1' \oplus A_1''$, with $A_1', A_1'' \neq 0$.
In this case we have

$$\begin{aligned}
&(\alpha_1{}^{\oplus}(A_1)\downarrow\alpha_2) = \\
&\quad = ((\alpha_1{}^{\oplus}(A_1') \oplus \alpha_1{}^{\oplus}(A_1''))\Downarrow\alpha_2) = \quad \text{[by Lemma 1.(1)]} \\
&\quad = (\alpha_1{}^{\oplus}(A_1')\Downarrow\alpha_2) \oplus (\alpha_1{}^{\oplus}(A_1'')\Downarrow\alpha_2)
\end{aligned}$$

On the other hand

$$\begin{aligned}
&A_2 = f_2{}^{\oplus}((A_1\Downarrow f_1)) = \quad \text{[by Lemma 1.(1)]} \\
&\quad = f_2{}^{\oplus}((A_1'\Downarrow f_1) \oplus (A_1''\Downarrow f_1)) = \\
&\quad = f_2{}^{\oplus}((A_1'\Downarrow f_1)) \oplus f_2{}^{\oplus}((A_1''\Downarrow f_1)) = \quad \text{[by inductive hypothesis]} \\
&\quad = (\alpha_1{}^{\oplus}(A_1')\Downarrow\alpha_2) \oplus (\alpha_1{}^{\oplus}(A_1'')\Downarrow\alpha_2)
\end{aligned}$$

Hence, the desired equality follows.

This concludes the proof of (2), from which (1) follows.

Now, by exploiting (1) we can easily conclude. In fact, the steps in Z_1 and Z_2 are of the kind

$$u_i = u_i' \oplus {}^{\bullet}A_i\ [A_i\rangle\ u_i' \oplus A_i{}^{\bullet}$$

for $i \in \{1, 2\}$. First observe that, since $(u_1\downarrow f_1) = (u_2\downarrow f_2)$ and $({}^{\bullet}A_1\downarrow f_1) = ({}^{\bullet}A_2\downarrow f_2)$, we immediately get:

$$(u_1'\downarrow f_1) = (u_2'\downarrow f_2)$$

Let $u_0' = (u_i'\downarrow f_i)$ be the common projection. Since $v_i = u_i' \oplus A_i{}^{\bullet}$, for $i \in \{1, 2\}$, by the fact that $(A_1{}^{\bullet}\downarrow f_1) = (A_2{}^{\bullet}\downarrow f_2)$, we deduce that, as desired

$$(v_1\downarrow f_1) = (v_2\downarrow f_2)$$

Hence, if $v_0 = (v_i\downarrow f_i)$ is the common projection, we can define $v_3 = v_1 \uplus_{v_0} v_2$.

Now, if we set $u_3' = u_1' \uplus_{u_0'} u_2'$ we have

$$\begin{aligned}
&u_3 = u_1 \uplus_{u_0} u_2 = \\
&\quad = (u_1' \oplus {}^{\bullet}A_1) \uplus_{u_0' \oplus {}^{\bullet}A_0} (u_2' \oplus {}^{\bullet}A_2) = \quad \text{[by Proposition 2]} \\
&\quad = (u_1' \uplus_{u_0'} u_2') \oplus ({}^{\bullet}A_1) \uplus_{{}^{\bullet}A_0} {}^{\bullet}A_2) = \quad \text{[by (1)]} \\
&\quad = u_3' \oplus {}^{\bullet}A_3
\end{aligned}$$

Therefore we have the step

$$u_3\ [A_3\rangle\ u_3' \oplus A_3{}^{\bullet}.$$

By a sequence of passages analogous to those used above, we can show that $u_3' \oplus A_3{}^{\bullet} = v_3$ and thus, as desired $u_3\ [A_3\rangle\ v_3$.

The fact that such step projects to $u_i\ [A_i\rangle\ v_i$ for $i \in \{1, 2\}$ immediately follows by construction. □

We are then able to show how steps of the components net can be "joined" to lead a step of their composition. This can be done when the steps satisfy a suitable compatibility condition, defined below.

Definition 10 (compatible steps). *Let* $Z_3 = Z_1 +_{f_1,f_2} Z_2$ *be a pushout in* **ONet** *(see Fig. 2). We say that two steps* $u_i\ [A_i\rangle\ v_i$ *(*$i \in \{1, 2\}$*) are* compatible *if* $(u_1\downarrow f_1) = (u_2\downarrow f_2)$ *and we can decompose the steps as* $A_i = A_i' \oplus A_i''$ *(*$i \in \{1, 2\}$*) such that*

$$A'_2 = {f_2}^{\oplus}((A'_1 \Downarrow f_1)) \quad \textit{and} \quad A''_1 = {f_1}^{\oplus}((A''_2 \Downarrow f_2))$$

It is immediate to see that if A_1 and A_2 are compatible, then $(A_1 \Downarrow f_1) = (A_2 \Downarrow f_2)$.

Lemma 4 (composing steps). *Let $f_1 : Z_0 \to Z_1$ and $f_2 : Z_0 \to Z_2$ be composable embeddings in* **ONet** *and let $Z_3 = Z_1 +_{f_1,f_2} Z_2$. Let $u_1\ [A_1\rangle\ v_1$ and $u_2\ [A_2\rangle\ v_2$ be compatible steps*

Then there exists a unique step $u_3\ [A_3\rangle\ v_3$ which is projected to $u_i\ [A_i\rangle\ v_i$ along α_i for $i \in \{1, 2\}$.

Proof. By definition of compatibility, we know that A_1 and A_2 can be decomposed as $A_i = A'_i \oplus A''_i$ ($i \in \{1, 2\}$) such that

$$A'_2 = {f_2}^{\oplus}((A'_1 \Downarrow f_1)) \quad \text{and} \quad A''_1 = {f_1}^{\oplus}((A''_2 \Downarrow f_2)).$$

Moreover, $(u_1 \downarrow f_1) = u_0 = (u_2 \downarrow f_2)$.

Now, since $u_i\ [A'_i \oplus A''_i\rangle\ v_i$, we can find markings u'_i, u''_i, v'_i, v''_i such that

$u'_i\ [A'_i\rangle\ v'_i$ and $(u'_1 \downarrow f_1) = (u'_2 \downarrow f_2) = u'_0$
$u''_i\ [A''_i\rangle\ v''_i$ and $(u''_1 \downarrow f_1) = (u''_2 \downarrow f_2) = u''_0$

In fact, just observe that, since $u_i\ [A_i\rangle\ v_i$, the marking u_i must be of the kind $w_i \oplus {}^\bullet A'_i \oplus {}^\bullet A''_i$ and thus we could choose $u'_i = {}^\bullet A'_1$, $v'_i = A'^\bullet_i$, $u''_i = {}^\bullet A''_i \oplus w_i$ and $v''_i = A''^\bullet_i \oplus w_i$.

Therefore, we can use Lemma 3 and, defining $u'_3 = u'_1 \uplus_{u'_0} u'_2$, $u''_3 = u''_1 \uplus_{u''_0} u''_2$, $v'_3 = v'_1 \uplus_{v'_0} v'_2$, $v''_3 = v''_1 \uplus_{v''_0} v''_2$, we conclude

$u'_3\ [{\alpha_1}^{\oplus}(A'_1)\rangle\ v'_3$ and $u''_3\ [{\alpha_2}^{\oplus}(A''_2)\rangle\ v''_3$

Therefore

$u'_3 \oplus u''_3\ [{\alpha_1}^{\oplus}(A'_1) \oplus {\alpha_2}^{\oplus}(A''_1)\rangle\ v'_3 \oplus v''_3$

By exploiting Proposition 2, we easily see that $u'_3 \oplus u''_3 = (u'_1 \oplus u''_1) \uplus_{u'_0 \oplus u''_0} (u'_2 \oplus u''_2) = u_1 \uplus_{u_0} u_2$, where u_0 denotes the common projection of u_1 and u_2 over Z_0. Similarly, $v'_3 \oplus v''_3 = v_1 \uplus_{v_0} v''_1$ and thus

$u_1 \uplus_{u_0} u''_1\ [{\alpha_1}^{\oplus}(A'_1) \oplus {\alpha_2}^{\oplus}(A''_1)\rangle\ v_1 \uplus_{v_0} v''_1$

is the desired step. The fact that it projects over the steps we started from in Z_1 and Z_2 follows by construction. □

4 Bisimilarity of open nets

In this section we study (strong and weak) bisimilarity for open nets. Then we prove that bisimilarity is a congruence with respect to the colimit-based composition of open nets.

First, we define a labelled transition system associated to an open net. Net transitions carry a label which is observed when they fire. Additionally, in the labelled transition system we also observe what happens at the open places. As discussed in the conclusions, this resembles the labelled transition system arising from the view of Petri nets as reactive systems in [12, 18].

Given an open net Z, the corresponding labelled transition system has the markings of the net as states. Transitions are generated by the firings of Z and labelled over the set

$$\Lambda_Z = \Lambda \cup \{+_s : s \in O_Z^+\} \cup \{-_s : s \in O_Z^-\}.$$

For notational convenience we extend the labelling function λ_Z to the set of extended events $\bar{T}_Z$, by defining $\lambda_Z(x) = x$ for $x \in \bar{T}_Z - T_Z$ (i.e., when $x = +_s$ or $x = -_s$ with $s \in S_Z$).

Definition 11 (lts for an open net). *The* labelled transition system *associated to an open net Z, denoted by $\mathsf{lts}(Z)$, is the pair $\langle {S_Z}^{\oplus}, \rightarrow_Z \rangle$, where states are markings $u_Z \in {S_Z}^{\oplus}$ and the transition relation $\rightarrow_Z \subseteq {S_Z}^{\oplus} \times \Lambda_Z \times {S_Z}^{\oplus}$ includes all transitions*

$$u_Z \xrightarrow{\lambda_Z(x)}_Z u'_Z$$

such that there is a firing $u_Z \; [x\rangle \; u'_Z$ in Z.

It commonly happens that, when observing the behaviour of a system, only a subset of events is considered observable or important. In our setting this is formalised by selecting a subset of labels representing internal/not observable firings (playing the role of τ-transitions) and then considering a corresponding notion of weak bisimilarity.

Let $\Lambda_\tau \subseteq \Lambda$ be a subset of *unobservable* labels, which is fixed for the rest of the paper. Given a Λ-labelled open Petri net Z, for markings $v, v' \in {S_Z}^{\oplus}$ let us write $v \rightsquigarrow_Z v'$ if $v \xrightarrow{\ell}_Z v'$ with $\ell \in \Lambda_\tau$, and $v \overset{\ell}{\rightsquigarrow}_Z v'$ if $v \xrightarrow{\ell}_Z v'$ with $\ell \in \Lambda_Z - \Lambda_\tau$. Then we define

- $v \Longrightarrow_Z v'$ when $v(\rightsquigarrow_Z)^* v'$.
- $v \overset{\ell}{\Longrightarrow}_Z v'$ when $v(\rightsquigarrow_Z)^* \overset{\ell}{\rightsquigarrow}_Z (\rightsquigarrow_Z)^* v'$.

4.1 Bisimilarity

The notion of weak bisimilarity is now defined in a standard way (but note that when the set of unobservable labels is empty, this actually corresponds to strong bisimilarity). Only, we need to specify for each open place of one net which is the corresponding open place in the other net, i.e., bisimulations between two nets are parametrised by a bijection between their open places. Given two open nets Z_1 and Z_2 we write $\eta : O_1 \leftrightarrow O_2$ and say that η is a *correspondence* between Z_1 and Z_2 if $\eta : O_1^+ \cup O_1^- \rightarrow O_2^+ \cup O_2^-$ is a bijection such that for $s_1 \in O_1$, $x \in \{+, -\}$ we have $s_1 \in O_1^x$ iff $\eta(s_1) \in O_2^x$.

Definition 12 ((weak) bisimilarity). *Let Z_1, Z_2 be open nets and $\eta : O_1 \leftrightarrow O_2$ be a correspondence between Z_1 and Z_2. A (weak) η-bisimulation over Z_1 and Z_2 is a relation over their markings $\mathcal{R} \subseteq {S_1}^{\oplus} \times {S_2}^{\oplus}$ such that if $(u_1, u_2) \in \mathcal{R}$ then*

- *if $u_1 \rightsquigarrow_{Z_1} u'_1$, then there exists u'_2 such that $u_2 \Longrightarrow_{Z_2} u'_2$ and $(u'_1, u'_2) \in \mathcal{R}$;*

– *if* $u_1 \overset{\ell}{\rightsquigarrow}_{Z_1} u_1'$*, then there exists* u_2' *such that* $u_2 \overset{\eta(\ell)}{\Longrightarrow}_{Z_2} u_2'$ *and* $(u_1', u_2') \in \mathcal{R}$*;*
– *the symmetric conditions holds;*

where $\eta(+_s) = +_{\eta(s)}$*,* $\eta(-_s) = -_{\eta(s)}$*, and* $\eta(\ell) = \ell$ *for any* $\ell \in \Lambda$*.*

Two open nets Z_1 *and* Z_2 *are* (weakly) η-bisimilar*, denoted* $Z_1 \approx_\eta Z_2$*, if* $\eta : O_1 \leftrightarrow O_2$ *is a correspondence and there exists a (weak)* η*-bisimulation* $\mathcal{R}$ *over* Z_1 *and* Z_2 *such that* $(\hat{u}_1, \hat{u}_2) \in \mathcal{R}$*. Sometimes we will simply say that* Z_1 *and* Z_2 *are (weakly) bisimilar, omitting the correspondence* η*.*

Notice that weak bisimulation boils down to the notion of strong bisimulation when all labels are observable, i.e., when $\Lambda_\tau = \emptyset$. For convenience of the reader we make explicit the notion of strong bisimilarity.

Definition 13 (strong bisimilarity). *When* Z_1 *and* Z_2 *are weakly* η*-bisimilar open nets, with* $\Lambda_\tau = \emptyset$ *we say that* Z_1 *and* Z_2 *are* strongly η-bisimilar*. Explicitly, a* strong η-bisimulation *over* Z_1 *and* Z_2 *is a relation over their markings* $\mathcal{R} \subseteq {S_1}^\oplus \times {S_2}^\oplus$ *such that if* $(u_1, u_2) \in \mathcal{R}$ *then*

– *if* $u_1 \xrightarrow{\ell}_{Z_1} u_1'$*, then there exists* u_2' *such that* $u_2 \xrightarrow{\eta(\ell)}_{Z_2} u_2'$ *and* $(u_1', u_2') \in \mathcal{R}$*;*
– *if* $u_2 \xrightarrow{\ell}_{Z_1} u_2'$*, then there exists* u_1' *such that* $u_1 \xrightarrow{\eta^{-1}(\ell)}_{Z_2} u_1'$ *and* $(u_1', u_2') \in \mathcal{R}$*;*

As already mentioned, open net morphisms are not simulations, since the target net can be more "instantiated" than the source net. However, according to the following lemma, which is a corollary of Lemma 3 and Lemma 4, given *composable* embeddings $f_1 : Z_0 \to Z_1$ and $f_2 : Z_0 \to Z_2$, the firing of a transition in Z_2, projected along f_2 to Z_0 can then be simulated in Z_1.

We first prove a preliminary lemma on unlabelled steps, which will have the desired result on labelled transitions as an easy corollary.

Lemma 5. *Let* $f_1 : Z_0 \to Z_1$ *and* $f_2 : Z_0 \to Z_2$ *be composable embeddings in* **ONet** *and let* $Z_3 = Z_1 +_{f_1,f_2} Z_2$*. Let* $u_2\ [A_2\rangle\ v_2$ *be a step consisting of transitions only, i.e.,* $A_2 \in {T_{Z_2}}^\oplus$*. Then* $u_1\ [{f_1}^\oplus((A_2 \Downarrow f_2))\rangle\ v_2$ *and* $u_3\ [{\alpha_2}^\oplus(A_2)\rangle\ v_3 = v_1 \uplus v_0 v_2$*.*

Proof. Let $A_1 = {f_1}^\oplus((A_2 \Downarrow f_2))$. First note that A_1 is well-defined, i.e., $A_1 \in \bar{T}^\oplus_{Z_1}$. In fact, let $+_{s_1} \in A_1$ and let us show that s_1 is input open. From the assumption we deduce that there is $+_{s_0} \in (A_2 \Downarrow f_2)$ with $f_1(s_0) = s_1$. Examining the definition of projection for steps, since A_2 consists only of transitions, this implies that $f(s_0) \in {}^\bullet t_2$, with $t_2 \notin f_2(T_{Z_0})$ and thus $s_0 \in \mathsf{in}(f_2)$. Since f_1 and f_2 are composable, we have that $s_1 = f_1(s_0) \in f_1(\mathsf{in}(f_2)) \subseteq O^+_{Z_1}$, as desired.

Now observe that

$$\begin{aligned} {}^\bullet A_1 &= {}^\bullet({f_1}^\oplus((A_2 \Downarrow f_2))) && \text{[by def. of open net morphism]} \\ &= {f_1}^\oplus\, {}^\bullet((A_2 \Downarrow f_2)) && \text{[by Lemma 1.(3)]} \\ &= {f_1}^\oplus(({}^\bullet A_2 \downarrow f_2)) \end{aligned}$$

Now, since the step $u_2\ [A_2\rangle\ v_2$ is enabled, we know that ${}^\bullet A_2 \leq u_2$, and thus

$$\begin{aligned} {}^\bullet A_1 &= {f_1}^\oplus(({}^\bullet A_2 \downarrow f_2)) \\ &\leq {f_1}^\oplus((u_2 \downarrow f_2)) && \text{[since } (u_2 \downarrow f_2) = (u_1 \downarrow f_1)\text{]} \\ &= {f_1}^\oplus((u_1 \downarrow f_1)) && \text{[by Lemma 1.(4)]} \\ &\leq u_1 \end{aligned}$$

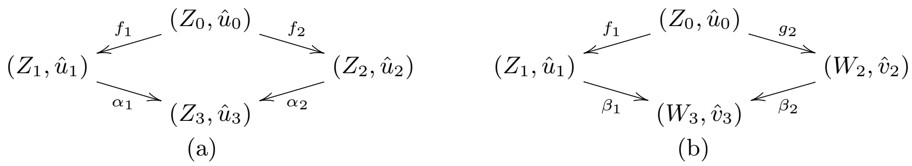


Fig. 4. Pushouts in **ONet**.

Hence, the step $u_1\ [A_1\rangle\ v_1$ can be performed. Clearly, the two steps in Z_1 and Z_2 are compatible, and thus we conclude by Lemma 4. $\square$

Lemma 6. *Let Z_0, Z_1, Z_2 be open nets and let $f_i : Z_0 \to Z_i$ ($i \in \{1,2\}$) be composable embeddings, as in Fig. 2. Furthermore, let $Z_3 = Z_1 +_{f_1,f_2} Z_2$.*

Assume $u_2 \xrightarrow{\ell}_{Z_2} u_2'$ where $\ell \in \Lambda$ and let $t \in T_2$ such that $\lambda_2(t) = \ell$ and $u_2\ [t\rangle\ u_2'$, let $u_0\ [A_0\rangle\ u_0'$ be its projection over Z_0 (hence $A_0 = (t \Downarrow f_2)$) and let $u_0 \xrightarrow{\ell_1}_{Z_0} \ldots \xrightarrow{\ell_n}_{Z_0} u_0'$ be any sequence of transitions in $\mathsf{lts}(Z_0)$ arising as a linearisation of such step in Z_0. Then we have, for any $u_1 \in {S_1}^{\oplus}$ such that $(u_1 \downarrow f_1) = u_0$:

$$u_1 \xrightarrow{\ell_1}_{Z_1} \ldots \xrightarrow{\ell_n}_{Z_1} u_1' \qquad \textit{and} \qquad u_1 \uplus_{u_0} u_2 \xrightarrow{\ell}_{Z_3} u_1' \uplus_{u_0'} u_2'.$$

Proof. Immediate by Lemma 5. $\square$

Note that above, when transition t is in the image of Z_0 the sequences of transitions in $\mathsf{lts}(Z_0)$ and $\mathsf{lts}(Z_1)$ are actually single firings. Instead, they are sequences of interactions over open places, possibly of length greater than one, otherwise.

By exploiting this lemma we can prove that bisimilarity is a congruence with respect to the composition operation on open nets.

Theorem 1 (bisimilarity is a congruence). *Let Z_0, Z_1, Z_2, W_2 be open nets. Let $Z_2 \approx_\eta W_2$, for some η. Consider the nets $Z_3 = Z_1 +_{f_1,f_2} Z_2$ and $W_3 = Z_1 +_{f_1,g_2} W_2$, as in Fig. 4 where f_1, f_2, g_2 are embeddings and f_1, f_2 and f_1, g_2 are composable.*

If $g_2|_{O_0} = \eta \circ (f_2|_{O_0})$ (i.e., f_2 and g_2 are consistent with η on open places) then $Z_3 \approx_{\eta'} W_3$, where η' is defined as follows: $\eta'(\alpha_1(s_1)) = \beta_1(s_1)$, whenever $\alpha_1(s_1)$ is open, and $\eta'(\alpha_2(s_2)) = \eta(s_2)$, whenever $\alpha_2(s_2)$ is open.

Proof. Let us compose the nets Z_0, Z_1, Z_2 and W_2 as shown in Fig. 4, where $(\eta \circ f_2)|_{O_0} = g_2|_{O_0}$. To simplify the notation, assume, without loss of generality, that, all the morphisms in the diagrams of Fig. 4 are inclusions and $\eta = id$. Hence $f_2|_{O_0} = g_2|_{O_0}$.

Now let $\mathcal{R}$ be a weak η-bisimulation over Z_2 and W_2 such that $(\hat{u}_2, \hat{v}_2) \in \mathcal{R}$, which exists by hypothesis. Consider the relation $\mathcal{R}'$ over Z_3 and W_3 defined as

$$\mathcal{R}' = \{(u_1 \uplus_{u_0} u_2, v_1 \uplus_{v_0} v_2) : (u_2, v_2) \in \mathcal{R} \ \wedge\ \exists u, v \in {S_{Z_0}}^{\oplus}.\ u_1 \oplus u = v_1 \oplus v\}$$

The condition on u_1 and v_1 means that the two markings can differ for the number of tokens in places of the interface net Z_0. Notice that the marking of Z_0 is completely determined by the marking of components Z_2 and W_2.

We claim that $\mathcal{R}'$ is a weak η'-bisimulation over Z_3 and W_3, where η' is again identity on open places. Since, by the construction of the pushout, $(\hat{u}_3, \hat{v}_3) = (\hat{u}_1 \uplus \hat{u}_0\hat{u}_2, \hat{u}_1 \uplus \hat{u}_0\hat{v}_2) \in \mathcal{R}'$, this provides the desired result.

In fact, assume that $u_3 \overset{\ell}{\rightsquigarrow}_{Z_3} u'_3$. We distinguish various possibilities:

1. $\ell = +_s$ (token put by the environment in an open input place)
   There are several possibilities:
   (a) $s \in S_0$.
   In this case, since $s \in O^+_{Z_3}$, necessarily $s \in O^+_{Z_i}$ for $i \in \{0, 1, 2\}$ and thus we can have the same firing in all the other components Z_0, Z_1 and Z_2, i.e., for $i \in \{1, 2, 3\}$
$$u_i \xrightarrow{+_s}_{Z_i} u'_i.$$
   and notice that $u'_3 = u'_1 \uplus_{u'_0} u'_2$.
   Since $(u_2, v_2) \in \mathcal{R}$, we have that
$$v_2 \overset{+_s}{\Longrightarrow}_{W_2} v'_2$$
   The above sequence of firings projects over Z_0 to
$$v_0 \xrightarrow{\ell_1}_{Z_0} \dots \xrightarrow{\ell_h}_{Z_0} \xrightarrow{+_s}_{Z_0} \xrightarrow{\ell'_1}_{Z_0} \dots \xrightarrow{\ell'_k}_{Z_0} v'_0$$
   where any $l_i \in \Lambda$ can be a τ-transition or an interaction with the environment.
   Hence, by Lemma 6 (applied several times) and recalling that $s \in O^+_1$ we deduce that there will be a firing sequence in Z_1
$$v_1 \xrightarrow{\ell_1}_{Z_1} \dots \xrightarrow{\ell_h}_{Z_1} \xrightarrow{+_s}_{Z_1} \xrightarrow{\ell'_1}_{Z_1} \dots \xrightarrow{\ell'_k}_{Z_1} v'_1 \qquad (3)$$
   and $v_3 \overset{+_s}{\Longrightarrow}_{W_3} v'_3$ for $v'_3 = v'_1 \uplus_{v'_0} v'_2$.
   Since $(u_3, v_3) \in \mathcal{R}'$, by construction u_1 and v_1 differ only for the number of tokens in the (the image of) Z_0. Hence, by construction, also u'_1 and v'_1 can differ only for the number of tokens in the places in Z_0 (these are the only places affected by the transitions labelled $+_s$, ℓ_1, ..., ℓ_h, ℓ'_1, ..., ℓ'_k). Hence $(u'_3, v'_3) \in \mathcal{R}$, as desired.
   (b) $s \in S_1 - S_0$
   Since $s \in O^+_{Z_3}$, necessarily $s \in O^+_{Z_1}$. The projections of the step in Z_3 over Z_1 is
$$u_1 \xrightarrow{+_s}_{Z_1} u_1 \oplus s.$$
   while Z_0 and Z_2 stay idle.
   Clearly an analogous step can be performed starting from v_1:
$$v_1 \xrightarrow{+_s}_{Z_1} v_1 \oplus s.$$
   and, since $s \notin S_0$, place s is open input also in W_3 The empty step in Z_2, by Lemma 4, can be composed with the above step leading to:
$$v_3 = v_1 \uplus_{v_0} v_2 \xrightarrow{+_s}_{Z_3} (v_1 \oplus s) \uplus_{v_0} v_{\oplus} s = v'_3$$
   and, as desired, $(u'_3, v'_3) \in \mathcal{R}$.
   (c) $s \in S_2 - S_0$
   Analogous to first case, but for the fact that firing $+_s$ is not in the firing sequence (3) for S_1.

2. $\ell = -_s$ (token taken by the environment from an open output place)
   Analogous to the previous case, by replacing $+_s$ with $-_s$. We only need the additional observation that in the firing sequence (3), the firing $-_s$ is possible, because it is possible in Z_0 and the marking of s is the same in Z_0 and Z_1.
3. $\ell = a \in \Lambda - \Lambda_\tau$
   In this case, the associated transition in Z_3 is of the kind

$$u_3[t\rangle u_3'$$

   where $\lambda_3(t) = a$. We distinguish various subcases.
   (a) *t in Z_1 and not in Z_2*
   The firing of t in Z_3 projects to the firing of a transition in Z_1, and to a (possibly empty) step consisting only of interactions with the external environment in Z_0 and Z_2. Let us consider a linearisation of such steps where all $-$'s firings precede the $+$'s firings, i.e.,

$$u_i \xrightarrow{\ell_1}_{Z_i} \dots \xrightarrow{\ell_n}_{Z_i} u_i' \quad i \in \{0, 2\}.$$

   where $\ell_i = -_{s_i}$ for $i \in \{1, \dots, h\}$ and $\ell_i = +_{s_i}$ for $i \in \{h+1, \dots, n\}$.
   By the fact that (Z_2, u_2) and (W_2, v_2) are weakly bisimilar we deduce that there is a sequence

$$v_2 \overset{\ell_1}{\Longrightarrow}_{Z_2} \dots \overset{\ell_n}{\Longrightarrow}_{Z_2} v_2' \tag{4}$$

   such that $(u_2', v_2') \in \mathcal{R}$.
   Again, we can rearrange the sequence (4) in order to have
   - first all the τ firings occurring in transitions $\overset{\ell_i}{\Longrightarrow}_{Z_2}$ ($i \in \{1, \dots, h\}$), which produce a marking allowing for the firings $-_{s_i}$ to be executed,
   - then $\ell_1, \dots, o_n$
   - and finally by the τ firings in $\overset{\ell_i}{\Longrightarrow}_{Z_2}$ ($i \in \{h+1, \dots, n\}$).

   This will project to a sequence

$$v_0 \xrightarrow{\ell_1'}_{Z_0} \dots \xrightarrow{\ell_k'}_{Z_0} v_0'' \xrightarrow{\ell_1}_{Z_0} \dots \xrightarrow{\ell_n}_{Z_0} \xrightarrow{\ell_1''}_{Z_0} \dots \xrightarrow{\ell_m''}_{Z_0} v_0' \tag{5}$$

   in Z_0.
   By Lemma 6, we deduce that the first part of the the sequence ℓ_1', ..., ℓ_k' can be performed also in Z_1, thus producing a marking v_1'' where ℓ_1, ..., ℓ_h are enabled. Now, since t is enabled in u_1 and the markings u_1 and v_1 differs only for tokens in the places of the interface and, on such tokens there are enough tokes, as witnessed by the enabling of ℓ_1, ..., ℓ_h, we have that the a-labelled transition t is enabled in Z_1 and can be performed. From the produced marking, again by Lemma 6 the firings ℓ_1'', ..., ℓ_m'' can be performed, i.e., we have:

$$v_1 \xrightarrow{\ell_1'}_{Z_1} \dots \xrightarrow{\ell_k'}_{Z_1} v_1'' \xrightarrow{a}_{Z_1} \xrightarrow{\ell_1''}_{Z_0} \dots \xrightarrow{\ell_m''}_{Z_0} v_1'$$

   which projects over (5) in Z_0.
   Therefore we can construct a marking

$$v_3' = v_1' \oplus_{v_0'} v_2'$$

   and $v_3 \overset{a}{\Longrightarrow}_{W_3} v_3'$. By definition of $\mathcal{R}'$ we thus finally have $(u_3', v_3') \in \mathcal{R}'$, as desired.

(b) t in Z_0 (hence t is both in Z_1 and Z_2)
In this case the firing projects to the firing of a transition in Z_i ($i \in \{0,1,2\}$)

$$u_i \xrightarrow{a}_{Z_i} u'_i \quad i \in \{0,1,2\}.$$

By the fact that (Z_2, u_2) and (W_2, v_2) are weakly bisimilar we deduce that

$$v_2 \stackrel{a}{\Longrightarrow}_{Z_2} v'_2 \tag{6}$$

such that $(u'_2, v'_2) \in \mathcal{R}$. Explicitly, the sequence (6) will be of the kind

$$v_2 \xrightarrow{\ell_1}_{Z_2} \dots \xrightarrow{\ell_k}_{Z_0} \xrightarrow{a}_{Z_2} \xrightarrow{\ell'_1}_{Z_2} \dots \xrightarrow{\ell'_m}_{Z_2} v'_2.$$

and thus it projects in Z_0 to

$$v_0 \xrightarrow{\ell''_1}_{Z_0} \dots \xrightarrow{\ell''_h}_{Z_0} v''_0 \xrightarrow{a}_{Z_0} \dots v'''_0 \xrightarrow{\ell'''_1}_{Z_0} \dots \xrightarrow{\ell'''_m}_{Z_0} v'_0 \tag{7}$$

where $\ell''_1, \dots, \ell''_h, \ell'''_1, \dots, \ell'''_m$ can be either τ's (projection of τ-transitions of Z_1 which are also in the interface Z_0) or interaction with the environment of the kind $+_{s'}$, $-_{s'}$.
By Lemma 6, we deduce that the same sequence can be performed also in Z_1

$$v_1 \xrightarrow{\ell''_1}_{Z_1} \dots \xrightarrow{\ell''_h}_{Z_1} v''_1 \xrightarrow{a}_{Z_1} \dots v'''_1 \xrightarrow{\ell'''_1}_{Z_1} \dots \xrightarrow{\ell'''_m}_{Z_1} v'_0 \tag{8}$$

and we can construct a marking

$$v'_3 = v'_1 \oplus_{v'_0} v'_2$$

and $v_3 \stackrel{a}{\Longrightarrow}_{W_3} v'_3$. By definition of $\mathcal{R}'$ we thus finally have $(u'_3, v'_3) \in \mathcal{R}'$, as desired.

(c) t in Z_2, but not in Z_0
In this case the firing projects to the firing of a transition in Z_2, and to (possibly empty) sequences of interactions with the external environment in Z_0 and Z_1

$$u_1 \xrightarrow{\ell_1}_{Z_1} \dots \xrightarrow{\ell_n}_{Z_1} u'_1 \tag{\dagger}$$

and

$$u_0 \xrightarrow{\ell_1}_{Z_0} \dots \xrightarrow{\ell_n}_{Z_0} u'_0.$$

By the fact that (Z_2, u_2) and (W_2, v_2) are weakly bisimilar we deduce that there exists

$$v_2 \stackrel{a}{\Longrightarrow}_{W_2} v'_2 \tag{9}$$

such that $(u'_2, v'_2) \in \mathcal{R}$. The firing sequence (9) projects over Z_0 to a sequence of interactions with the environment

$$v_0 \xrightarrow{\ell'_1}_{Z_0} \dots \xrightarrow{\ell'_k}_{Z_0} v'_0.$$

and, as in the previous cases, by using Lemma 6, we deduce that there will be a firing sequence in Z_1

$$v_1 \xrightarrow{\ell'_1}_{Z_1} \dots \xrightarrow{\ell'_k}_{Z_1} v'_1. \tag{\ddagger}$$

and we can define $v'_3 = v'_1 \oplus_{v'_0} v'_2$ such that $v_3 \stackrel{a}{\Longrightarrow}_{W_3} v'_3$. As in the previous cases we deduce that $(u'_3, v'_3) \in \mathcal{R}'$.

If instead, $u_3 \rightsquigarrow_{Z_3} u'_3$, then we must prove that $v_3 \Longrightarrow_{W_3} v'_3$, with $(u'_3, v'_3) \in \mathcal{R}'$. This should be completely analogous to item 3 above. □

4.2 Some properties and proof techniques

We next present some properties of (strong and weak) bisimilarity, which can be used in bisimulation proof.

The next result shows that given two bisimilar nets, if we "close" the same open places in both nets we still get two bisimilar nets. Given an open net Z and an open place $s \in O^x_Z$, let us denote by $Z - (s, x)$ the open net obtained from Z by closing place s, i.e., $Z' = (N, O_{Z'})$, where $O^x_{Z'} = O^x_Z - \{s\}$. The initial marking remains the same.

Proposition 4. *Let $Z_1 \sim_\eta Z_2$. Let $s \in O^x_1$ ($x \in \{-,+\}$) be an open place in Z_1. Then the nets $Z_1 - (s, x)$ and $Z_2 - (\eta(s), x)$ are strongly bisimilar.*

Proof. Let $Z'_1 = Z_1 - (s, x)$ and $Z'_2 = Z_2 - (\eta(s), x)$. Let $\mathcal{R} \subseteq {S_1}^{\oplus} \times {S_2}^{\oplus}$ be an η-bisimulation such that $(\hat{u}_1, \hat{u}_2) \in \mathcal{R}$. Then $\mathcal{R}$ is a bisimulation between Z'_1 and Z'_2. In fact, if $(u_1, u_2) \in \mathcal{R}$ and $u_1 \overset{\ell}{\rightsquigarrow}_{Z'_1} u'_1$ then clearly $u_1 \overset{\ell}{\rightsquigarrow}_{Z_1} u'_1$. Since $\mathcal{R}$ is a bisimulation for Z_1 and Z_2 this implies that $u_2 \overset{\eta(\ell)}{\Longrightarrow}_{Z_2} u'_2$ with $(u'_1, u'_2) \in \mathcal{R}$. Since ℓ is a label in Z'_1 where place s has been closed, we are sure that $\ell \neq x_s$, and thus $u_2 \overset{\eta(\ell)}{\Longrightarrow}_{Z_2} u'_2$ implies $u_2 \overset{\eta(\ell)}{\Longrightarrow}_{Z'_2} u'_2$. The case in which $u_1 \rightsquigarrow_{Z'_1} u'_1$ is treated analogously. Hence we get the desired result.

We next provide a kind of *up-to technique* for open net bisimilarity. Given an open net Z, let us define the *out-degree* of a place $s \in S$ as

$$deg(s) = \max\left(\{({}^{\bullet}t)(s) : t \in T_Z\} \cup \{1 : s \in O^-_Z\}\right)$$

The idea, formalised in the notion of up-to bisimulation, is to allow tokens to be removed from open input places, when they exceed the out-degree of the place. More precisely, given a net Z and a marking $u \in S^{\oplus}$, let us say that a marking $v \in (O^+_Z)^{\oplus}$ is *subtractable* from u if $\forall s \in O^+_Z.\ deg(s) \leq u(s) - v(s)$. Note that this implies that all transitions enabled in u are also enabled in $u \ominus v$.

Definition 14 (up-to bisimulation). *Let Z_1 and Z_2 be open nets, and let $\eta : O_1 \leftrightarrow O_2$ be a correspondence between Z_1 and Z_2. A relation $\mathcal{R} \subseteq {S_1}^{\oplus} \times {S_2}^{\oplus}$ between markings is called an* up-to η-bisimulation *if whenever $(u_1, u_2) \in \mathcal{R}$ then*

- *if $u_1 \rightsquigarrow_{Z_1} u'_1$, then there exists u'_2 such that $u_2 \Longrightarrow_{Z_2} u'_2$ and $v_1 \in (O^+_1)^{\oplus}$ subtractable from u_1 with $(u'_1 \ominus v_1, u'_2 \ominus \eta^{\oplus}(v_1)) \in \mathcal{R}$.*
- *if $u_1 \overset{\ell}{\rightsquigarrow}_{Z_1} u'_1$, then there exists u'_2 such that $u_2 \overset{\eta(\ell)}{\Longrightarrow}_{Z_2} u'_2$ and $v_1 \in (O^+_1)^{\oplus}$ subtractable from u_1 with $(u'_1 \ominus v_1, u'_2 \ominus \eta^{\oplus}(v_1)) \in \mathcal{R}$;*
- *the symmetric conditions hold;*

A first technical lemma shows an invariance property of up-to bisimulations, with respect to adding tokens in open places.

Lemma 7. *Let Z_1 and Z_2 be open nets, let $\eta : O_1 \leftrightarrow O_2$ be a correspondence between Z_1 and Z_2, and let $\mathcal{R}$ be an up-to η-bisimulation between Z_1 and Z_2. Then, given any $s \in O^+_1$, the relation $\mathcal{R}' = \mathcal{R} \cup \{(u_1 \oplus s, u_2 \oplus \eta(s)) : (u_1, u_2) \in \mathcal{R}\}$ is an up-to η-bisimulation.*

Proof. In order to simplify the notation, let us assume, without loss of generality, that η is the identity (i.e., $O_1^+ = O_2^+$ and $O_1^- = O_2^-$).

Let $(u_1 \oplus s, u_2 \oplus s) \in \mathcal{R}'$. Let us show that if $u_1 \oplus s \overset{\ell}{\rightsquigarrow}_{Z_1} u_1'$ then there exists $u_2 \oplus s \overset{\ell}{\Longrightarrow}_{Z_2} u_2'$ and $v \in O_1^+$ subtractable from u_1 with $(u_1' \ominus v, u_2' \ominus v) \in \mathcal{R}'$. The other cases are completely analogous.

Observe that, since $s \in O_1^+$, we have

$$u_1 \overset{+_s}{\rightsquigarrow}_{Z_1} u_1 \oplus s.$$

By definition of $\mathcal{R}'$, we have $(u_1, u_2) \in \mathcal{R}$ and thus

$$u_2 \overset{+_s}{\Longrightarrow}_{Z_2} u_2'' \quad \text{and} \quad (u_1 \oplus s \ominus v', u_2'' \ominus v') \in \mathcal{R}' \tag{10}$$

for a suitable $v' \in O_1^+$ subtractable from $u_1 \oplus s$. Also notice that, since a $+_s$ can always be performed, we can assume that the firing sequence (10) is of the kind

$$u_2 \xrightarrow{+_s}_{Z_2} u_2 \oplus s \Longrightarrow_{Z_2} u_2'' \tag{11}$$

Now, if $u_1 \oplus s \overset{\ell}{\rightsquigarrow}_{Z_1} u_1'$, then, since v' is subtractable from $u_1 \oplus s$, also $u_1 \oplus s \ominus v' \overset{\ell}{\rightsquigarrow}_{Z_1} u_1' \ominus v'$. Thus, by (10)

$$u_2'' \ominus v' \overset{\ell}{\Longrightarrow}_{Z_2} u_2''' \quad \text{and} \quad (u_1' \ominus v' \ominus v'', u_2''' \ominus v'') \in \mathcal{R}' \tag{12}$$

for a suitable $v'' \in O_1^{+\oplus}$, subtractable from $u_1 \ominus v''$.

Putting the above together with (11), we have that

$$u_2 \oplus s \Longrightarrow_{Z_2} u_2'' \overset{\ell}{\Longrightarrow}_{Z_2} u_2''' \oplus v'$$

i.e., $u_2 \oplus s \overset{\ell}{\Longrightarrow}_{Z_2} u_2''' \oplus v'$ and, if we denote $u_2' = u_2''' \oplus v'$, $(u_1' \ominus v' \ominus v'', u_2' \ominus v' \ominus v'') \in \mathcal{R}'$. It is immediate to see that $v' \oplus v''$ is subtractable from u_1', and thus we conclude. □

Corollary 1. *Let Z_1 and Z_2 be open nets, let $\eta : O_1 \leftrightarrow O_2$ be a correspondence between Z_1 and Z_2, and let $\mathcal{R}$ be an up-to η-bisimulation between Z_1 and Z_2. Then the relation $\mathcal{R}' = \mathcal{R} \cup \{(u_1 \oplus v_1, u_2 \oplus \eta^{\oplus}(v_1)) : (u_1, u_2) \in \mathcal{R} \ \wedge \ v_1 \in O_1^{+\oplus}\}$ is an up-to η-bisimulation.*

Proof. By an inductive reasoning, exploiting the previous result, we can show that $\mathcal{R}_n = \mathcal{R} \cup \{(u_1 \oplus v_1, u_2 \oplus \eta^{\oplus}(v_1)) : (u_1, u_2) \in \mathcal{R} \ \wedge \ v_1 \in O_1^{+\oplus} \ \wedge \ |v_1| \leq n\}$ is a weak bisimulation up-to for any n. Then we exploit the fact that the union of weak bisimulations up-to is again a weak-bisimulation up-to. □

Proposition 5. *Let Z_1 and Z_2 be open nets, and let $\eta : O_1 \leftrightarrow O_2$ be a correspondence between Z_1 and Z_2. Let $\mathcal{R}$ be an up-to η-bisimulation. Then for any $(u_1, u_2) \in \mathcal{R}$ we have that $(Z_1, u_1) \approx_\eta (Z_2, u_2)$.*

Proof. In order to simplify the notation, let us assume, without loss of generality, that η is the identity (i.e., $O_1^+ = O_2^+$ and $O_1^- = O_2^-$).

Let show that

$$\mathcal{R}' = \{(u_1 \oplus v, u_2 \oplus v) : (u_1, u_2) \in \mathcal{R} \ \wedge \ v \in (O_1^+)^{\oplus}\}$$

is an η-bisimulation. Let $(u_1 \oplus v, u_2 \oplus v) \in \mathcal{R}'$, with $(u_1, u_2) \in \mathcal{R}$ and $v \in O_1^+$, and assume that

$$u_1 \oplus v \overset{\ell}{\rightsquigarrow}_{Z_1} u_1'$$

(the case in which $u_1 \oplus v \rightsquigarrow_{Z_1} u_1'$ is treated analogously).

By Corollary 1 we know that $\mathcal{R}'$ is an up-to bisimulation, and thus there exists a transition

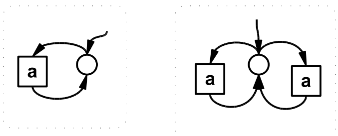
$$u_2 \oplus v \overset{\ell}{\Longrightarrow}_{Z_2} u_2'$$

and $v' \in O_1^{+\oplus}$, subtractable from u_1', such that $(u_1' \ominus v', u_2' \ominus v') \in \mathcal{R}'$. However, by construction of $\mathcal{R}'$, this implies that

$$(u_1', u_2') \in \mathcal{R}'$$

as desired. □

Note that, as it often happens with up-to techniques, the above result allows to show that two nets are bisimilar by exhibiting finite relations (while bisimulations are typically infinite). E.g., consider the open nets below, where label a is observable:



Then a bisimulation would include at least the pairs $\{(k \cdot s, k \cdot s) : k \in \mathbb{N}\}$, where s is the only place. Instead, according to the definition above $\{(0,0), (s,s)\}$ is an up-to bisimulation.

5 Behaviour preserving transformations

The results in the previous sections are used here to design a framework where a system specified as a (possibly open) Petri net can be dynamically reconfigured by transformation rules, triggered by the state/shape of the system. The congruence result allows to characterise classes of reconfigurations which preserve the observational behaviour of the system.

5.1 Transforming open nets

The fact that the composition operation over open nets is defined in terms of a pushout construction suggests naturally a way of reconfiguring open nets by using the double-pushout approach to rewriting [6].

A *rewriting rule* over open nets consists of a pair of morphisms in **ONet**:

$$p = L_p \overset{l_p}{\leftarrow} K_p \overset{r_p}{\rightarrow} R_p$$

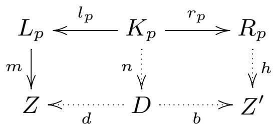


Fig. 5. Double-pushout rewriting over open nets

where L_p, K_p, R_p are open nets, called *left-hand side*, *interface* and *right-hand side* of the rule p, and l_p, r_p are open net embeddings. Intuitively, the rule specifies that, given a net Z, if the left-hand side L_p matches a subnet of Z then this can be reconfigured into Z' by replacing the occurrence of L_p with the right-hand side R_p, preserving the interface net K_p.

The notion of transformation is formally defined below.

Definition 15 (open net transformation). *Let p be a rewriting rule over open nets, let Z be an open net and let $m : L_p \to Z$ be a match, i.e., an open net embedding. We say that Z rewrites to Z' using p at match m, writing $Z \Rightarrow^{p,m} Z'$ or simply $Z \Rightarrow^{p} Z'$, if the diagram of Fig. 5 can be constructed in* **ONet***, where both squares are required to be pushouts and the arrows l_p, n and r_p, n are composable.*

We stress that we are interested in transformations where the two pushout squares are built from composable arrows (technically, this ensures that the transformation can be performed in **Net** and then "lifted" to **ONet**).

Hence, given a rule p and a match $m : L_p \to Z$, in order to perform the transformation

- The *pushout complement* of l_p and m must exist. The resulting arrows n and d must be such that l_p and n are composable. Additionally, there can be several pushout complements and in this case a canonical choice should be considered.
- The resulting arrow n must be composable with r_p.

Sufficient hypotheses under which these conditions are satisfied are made explicit below. Unfortunately, although a general theory of DPO rewriting has been recently developed in the framework of adhesive categories (see [8]), we cannot use the corresponding results here since the category of open nets falls outside the scope of the theory.

A first necessary condition for the existence of the pushout complement is a sort of *dangling condition*: a place can be deleted only if all the transitions connected to this place are removed as well (otherwise the flow arcs of this transition would remain dangling). This condition ensures that the pushout complement exists and is unique in the underlying category **Net**. Still, this might not be sufficient to conclude the existence of the pushout complement in **ONet**. Consider, for instance, the diagram in Fig. 6. It is easy to realise that the only place in D must be open input since a transition is attached to such place in Z. However, the resulting diagram is not a pushout: since the places in L and in D are input open also their image in Z should be input open.

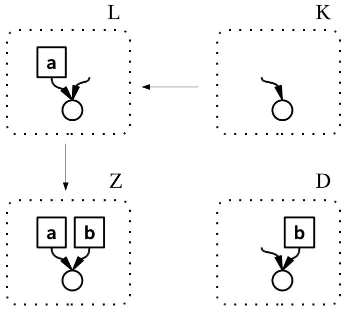


Fig. 6. A pushout complement in **Net** which cannot be lifted to **ONet**.

An additional requirement is thus imposed, which ensures the existence of a minimal pushout complement D, which embeds into any other pushout complement.

In order to simplify the notation, in the sequel we will assume, without loss of generality, that for any rule p, the arrows l_p and r_p are inclusions.

Lemma 8. *Let p be a rewriting rule over open nets, let Z be an open net and let $m : L_p \to Z$ be a match. Assume that*

1. *for all places $s \in L_p - K_p$ we have ${}^\bullet m(s), m(s)^\bullet \subseteq m(L_p - K_p)$ (*dangling condition*)*
2. *$m(l_p(\mathsf{in}(l_p)) \cap O^+_{L_p}) \subseteq O^+_Z$ and $m(l_p(\mathsf{out}(l_p)) \cap O^-_{L_p}) \subseteq O^-_Z$.*
3. *$m(O^x_L - l(K)) \subseteq O^x_Z$ for $x \in \{+, -\}$.*

Then the pushout complement exists in **Net***, defined as $D = Z - m(L_p - K_p)$ componentwise, and can be lifted to the minimal pushout complement in* **ONet** *by taking as open input places:*

$$O^+_D = d^{-1}(O^+_Z) \cup n(O^+_{K_p} - O^+_{L_p})$$

Open output places are defined analogously. The initial marking $\hat{u}_D$ is defined by $\hat{u}_D(s) = \hat{u}_Z(d(s))$ for any place $s \in S_D$.

Proof. The proof is long, but straightforward. We have already motivated the dangling condition above. In order to understand condition 2 above, observe that, roughly, $l_p(\mathsf{in}(l_p))$ is the set of places s in L_p, such that applying the rule p the place s is preserved but at least one transition in ${}^\bullet s$ is removed. Thus the corresponding places in D must be open, and, if they are open also in L_p, they must be open in the net Z. Similarly, for condition 3, if a place is open in L and it is not in the image of K then necessarily it will be open in Z.

Formally we have to show that (a) the mapping n and d are well-defined open net morphisms, (b) l and m are composable and (c) Z is the pushout.

(a.1) n is a well-defined open net morphism.
Let us prove that $n^{-1}(O_D^+) \cup \mathsf{in}(n) \subseteq O_K^+$, the condition on open output places is analogous. If $s \in n^{-1}(O_D^+)$ we have two possibilities according to the way O_D^+ is defined.

- If $n(s) \in d^{-1}(O_Z^+)$ then $d(n(s)) \in O_Z^+$. Since $d \circ n = m \circ l$ and $m \circ l$ is a well-defined open net morphism, we deduce that $s \in O_K^+$.
- If $n(s) \in n(O_K^+ - O_L^+)$, since n is injective, we have that $s \in O_K^+ - O_L^+ \subseteq O_K^+$.

Concerning the initial marking, note that for any $s \in S_K$ we have $\hat{u}_K(s) = \hat{u}_Z(m(l(s)) = \hat{u}_D(d(n(s)) = \hat{u}_D(s)$, where the last equality holds by construction.

(a.2) n is a well-defined open net morphism.
Also in this case we only prove that $d^{-1}(O_Z^+) \cup \mathsf{in}(d) \subseteq O_D^+$. If $s \in d^{-1}(O_Z^+)$ then $s \in O_D^+$ by definition. If, instead, $s \in \mathsf{in}(d)$ then it is easy to see that there exists $s' \in S_K$ such that $s' \in \mathsf{in}(l) \subseteq O_K^+$. Now, there are two subcases:

- If $l(s') \in O_L^+$ we have that $s' \in l(\mathsf{in}(l)) \cap O_L^+$ and thus $m(s') \in m(l(\mathsf{in}(l)) \cap O_L^+) \subseteq O_Z^+$ by condition 2. Since $d(s) = m(s')$ we deduce that $s \in d^{-1}(O_Z^+) \subseteq O_D^+$ by construction of D.
- If $l(s') \notin O_L^+$ then $s' \in O_K^+ - O_L^+$, and thus $sn(s') \in n(O_K^+ - O_L^+) \subseteq O_D^+$, by construction of D.

The condition over the initial marking is trivially satisfied by construction.

(b) n and l are composable.
We show the two conditions for composability separately:

- $n(\mathsf{in}(l))) \subseteq O_D^+$
  In fact, if $s \in \mathsf{in}(l)$, then it is easy to see that $m(l(s)) \in \mathsf{in}(d) \subseteq O_D^+$. Now, $m(l(s)) = d(n(s))$ and, since d is an open net morphism, it must reflect open places, and thus $n(s) \in O_D^+$.
- $l(\mathsf{in}(n))) \subseteq O_L^+$
  If $s \in l(\mathsf{in}(n))$ then, it is easy to see that $s \in \mathsf{in}(m) \subseteq O_L^+$, as desired.

(c) Z is the pushout.
We know that Z is the pushout of n and l in **Net**. We have to prove that it is also a pushout in **ONet**.
Concerning the set of open places we have to show that

$$O_Z^x \supseteq \{s \in S_Z : m^{-1}(s) \subseteq O_L^x \ \wedge \ d^{-1}(s) \subseteq O_D^x\}$$

the converse inclusion, and thus equality, follows from the fact that m and d are open net morphisms.
Let $s \in S_Z$ such that there are $s' \in O_L^+$ and $s'' \in O_D^+$ such that $m(s') = s = d(s'')$. Thus, there is $s''' \in S_K$ such that $l(s''') = s'$ and $n(s''') = s''$.
Since $s'' \in O_D^+$, then either $s'' \in d^{-1}(O_Z^+)$ or $s'' \in n(O_K^+ - O_L^+)$. Since $s' \in O_L^+$ and $l(s''') = s'$, the second possibility cannot arise. In the first case $s = d(s'') \in O_Z^+$, as desired.
When s is only in the image of D, the proof is analogous. When it is only in the image of L, we can use condition 3 in the hypothesis. $\square$

Given a match $m : L_p \to Z$ as in the proposition above, the transformation can be completed if $n : K_p \to D$ and $r_p : K_p \to R_p$ are composable. For this, first we introduce a special class of rules and then we suitably restrict matches.

Definition 16 (proper rules). *An open net rule p is called* proper *if $r_p(l_p^{-1}(O_L^+)) \subseteq O_R^+$.*

Intuitively, a rule is proper if open places left-hand side net, which are not deleted, remain open in the right-hand side net.

Definition 17 (proper match). *Let p be a rewriting rule over open nets and let Z be an open net. A match $m : L_p \to Z$ is called* proper *if it satisfies conditions 1, 2, and 3 in Lemma 8 and for any $s \in L_p$,*

$$\text{if } s \in l_p(\mathsf{in}(r_p)) \text{ then } m(s) \in O_Z^+$$

plus the dual condition on output places.

Intuitively, a match is proper if whenever $s \in l_p(\mathsf{in}(r_p))$, i.e., the rule p create a new (ingoing) transition connected to place s, then $m(s)$ is (input) open.

We finally arrive at the desired result.

Lemma 9. *Let p be a proper rule over open nets, let Z be an open net and let $m : L_p \to Z$ be a proper match. Then there exists a transformation $Z \Rightarrow^{p,m} Z'$.*

Proof (Sketch). Let p be a proper rule over open nets, let Z be an open net and let $m : L_p \to Z$ be a proper match.

Then, by using Lemma 8 we can construct the minimal pushout complement of l_p and m, as in Fig. 5.

In order to conclude, it suffices to show that n and r_p are composable. To this aim observe that

- $n(\mathsf{in}(r_p)) \subseteq O_D^+$ (+ same condition for $\mathsf{out}(.)$)
  This is ensured by properness of the match.
- $r(\mathsf{in}(n)) \subseteq O_R^+$ (+ same condition for $\mathsf{out}(.)$)
  This is a consequence of the properness of the rule. □

We conclude with a corollary of the theory developed in this section, which provides some sufficient conditions for the existence of a transformation, when the open places, i.e., the interface of a net, are not modified by the rewriting rules.

Lemma 10 (existence of transformations in ONet). *Let p be a rule over open nets, let Z be an open net and let $m : L_p \to Z$ be a match such that:*

1. *for all $s \in L_p - l_p(K_p)$ we have ${}^\bullet m(s) \subseteq m(L_p - K_p)$;*
2. *$O_R^+ = r_p(l_p^{-1}(O_L^+))$ and $O_L^+ = l_p(r_p^{-1}(O_R^+))$;*
3. *for all $s \in L_p$, if $s \in l_p(\mathsf{in}(r_p))$ then $m(s) \in O_Z^+$;*
4. *for all $s \in L_p$, if $s \in l_p(\mathsf{in}(l_p))$ then either $m(s) \in O_Z^+$ or $s \notin O_L^+$;*

and the dual conditions, obtained by replacing ${}^\bullet()$ by $()^\bullet$, $\mathsf{in}()$ by $\mathsf{out}()$ and $+$ by $-$, hold. Then, there exists a transformation $Z \Rightarrow^{p,m} Z'$, later referred to as a proper *transformation.*

As an example of a rewriting rule, consider the one on the top of Fig. 7. It is not difficult to see that rule and match satisfy the conditions of Lemma 10. Hence we can transform Z into Z', as depicted in the same figure.

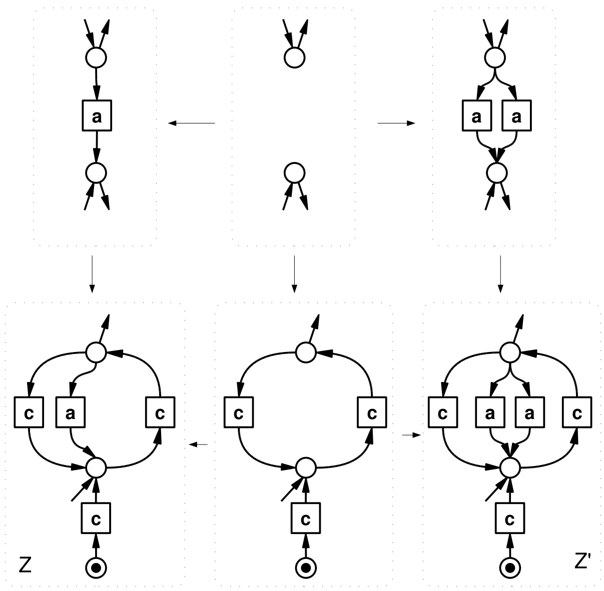


Fig. 7. Transforming open nets through DPO rewriting.

Remark 1 (On the choice of the PO complement). As already mentioned, the pushout complement is not unique in general. Consider for instance, the diagrams in Fig. 8, where both D and D' are pushout complements for l_p and m.

A possible solution is the one we adopted in the paper, i.e., to consider the minimal PO complement as a canonical choice. However there are other conceivable choices:

1. Restrict to *left-linear rules*, where l is regular mono. This guarantees uniqueness of the PO complement (notice that morphism l in the counterexample above is not regular mono since it does not preserve open places). Notice that if l is regular mono then $l(\mathsf{in}(l)) \subseteq l(O_K^+) \subseteq O_L^+$, hence the condition 2 in Lemma 8 would simplify to

$$m(l(\mathsf{in}(l))) \subseteq O_Z^+$$

   However it seems that in many practical situations, left-linear rules are too restrictive.
2. Consider any PO complement (not necessarily the minimal one) which ensures that the right PO exists.

   Having a non minimal PO complement D could allow to satisfy the condition above $n(\mathsf{in}(r)) \subseteq O_D^+$ for a larger class of transformations.

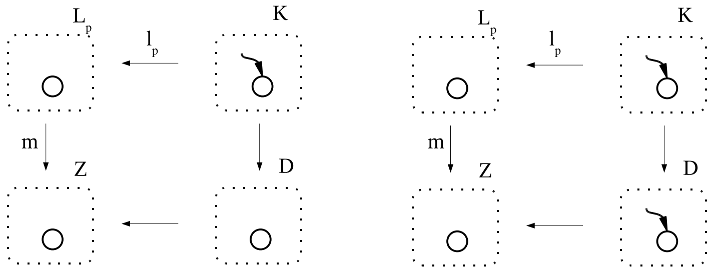


Fig. 8. Pushout complement is not unique in $\mathbf{ONet}^u$.

5.2 Preserving the behaviour

The framework of open Petri nets allows us to specify a system as built out of smaller components. Then, its behaviour is captured by the firing behaviour of the open net. However, for highly dynamic systems, as mentioned in the introduction, it can be quite useful to have the possibility of specifying that, under some conditions, some structural changes or reconfigurations can intervene. For instance the requirement of a service could trigger a rule which provides an implementation of the required service. The rewriting rules over open nets introduced above can do the job. From a subclass of such reconfigurations we expect that they "preserve" the behaviour of the system. Reconfigurations of this kind can be naturally characterised in our framework, as expressed by the following result, which follows from Theorem 1.

Theorem 2 (behaviour preserving reconfigurations). *Consider a proper open net rewriting rule p such that $L_p \approx_\eta R_p$, where $r_{p|O_{K_p}} = \eta \circ l_{p|O_{K_p}}$. Given an open net Z, if $Z \Rightarrow^{p,m} Z'$ via a proper match $m : L_p \to Z$, then $Z \approx_{\eta'} Z'$ for a correspondence η' defined as in Theorem 1.*

Proof (Sketch). Just observe that, in the DPO diagram

$$\begin{array}{ccccc} L_p & \xleftarrow{l_p} & K_p & \xrightarrow{r_p} & R_p \\ {\scriptstyle m}\downarrow & & {\scriptstyle n}\downarrow & & \downarrow{\scriptstyle h} \\ Z & \xleftarrow[d]{} & D & \xrightarrow[b]{} & Z' \end{array}$$

since the arrows l_p, n and r_p, n are composable, we can apply Theorem 1, and conclude that $Z \approx Z'$. □

For instance, consider again the DPO diagram in Fig. 7. It can be easily seen that the left-hand side and right-hand side of the applied rule are strongly bisimilar. Hence we can conclude that also Z and Z' are strongly bisimilar.

For a more elaborated example, consider net N_0 in the left of Fig. 10 which models the view of a traveller on the journey planning and ticket purchase services offered through a travel agency portal.

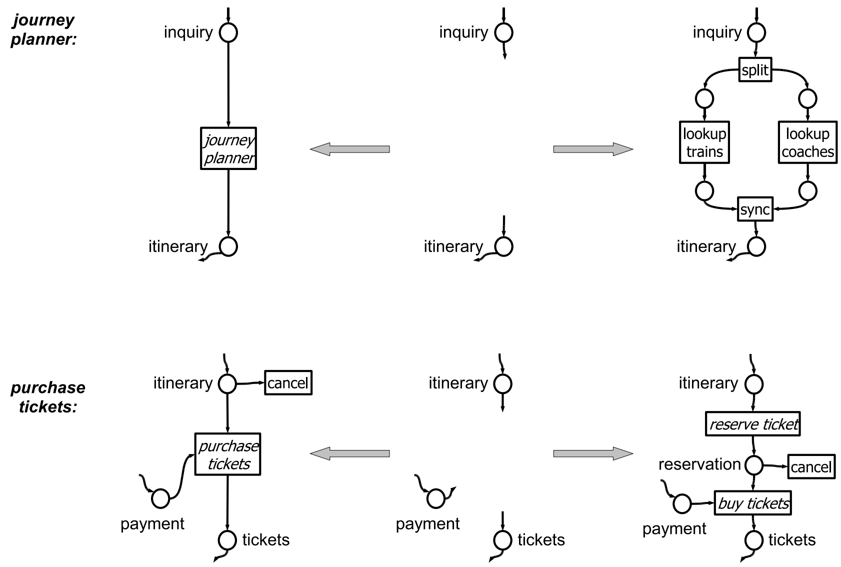


Fig. 9. Rules

We distinguish *abstract transitions* representing services that should be provided elsewhere and *concrete transitions* representing existing services, local and control flow actions. The invocation of an external service can be seen at different levels of abstraction. From the point of view of the client process it is captured by firing an abstract transition. For an internal view of the execution of the service, we capture its structure by a rule like the one in the top of Fig 9. An application of this rule, replacing the abstract transition by a new open net, represents the discovery and binding of the concrete services required. Note that the left-hand side and right-hand side of the rule are weakly bisimilar if we observe only the interactions at the open (interface) places, i.e., if we take $\Lambda_\tau = \Lambda$. This ensures that both in the abstract transition and in its concrete counterpart any inquiry will produce a corresponding itinerary.

The rule in the bottom of Fig. 9 represents a case where a simple pattern is replaced by a slightly richer one. On the left we say that, given an itinerary, we can either purchase the required tickets or cancel the processes. On the right the transaction is refined, adding a prior reservation phase, while keeping the option to cancel. As above, the rule has weakly bisimilar left- and right-hand sides, ensuring that the visible effect of the abstract and concrete transitions at the interfaces is the same. A possible sequence of transformations is shown in Fig. 10. By the above considerations, we are sure that the transformations do not change the observable behaviour of the system, a fact that can be interpreted as a proof of conformance of the provided service with respect to the abstract specification.

As they are given, rules could be applied either statically, assembling the process at design time, or dynamically at runtime. Notice that we could also enrich the rules by markings. For example, if rule *journey planner* was to contain a token at the

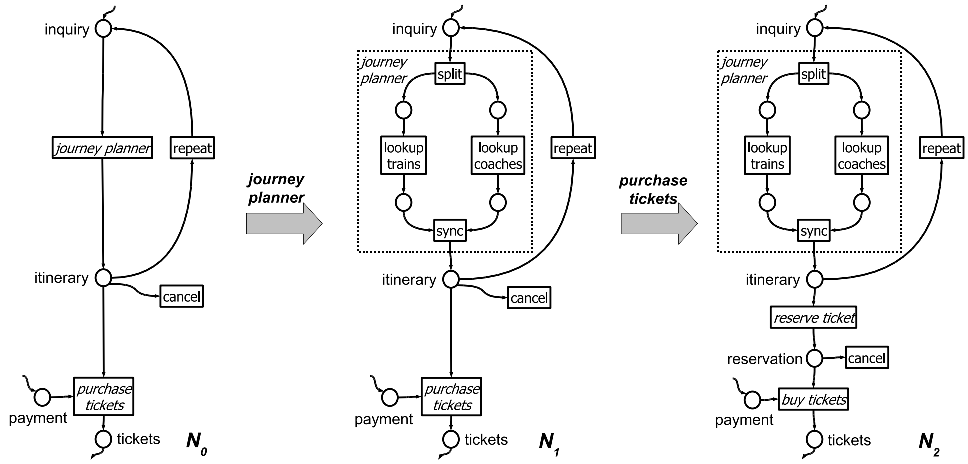


Fig. 10. Transformation of open nets representing a travel agent's portal.

inquiry place in its left- and right-hand side, the binding of the concrete services could only happen on demand at runtime.

Also note that, although in the considered examples the rewriting rules replace single transitions by a subnet, like in a refinement-based approach, the theory works for general reconfigurations, in which both the left- and right-hand sides can be arbitrarily large nets.

This leads to the definition of reconfigurable open net.

Definition 18 (reconfigurable open net). *A* reconfigurable open net *is a triple* $RZ = (Z, \hat{u}_Z, \Theta, \Theta_e)$ *where* Z *is an open net,* $\hat{u}_Z$ *is its initial marking,* Θ *is a set of open net rules, and* $\Theta_e \subseteq \Theta$ *is the subset of behaviour-preserving rules, i.e., such that for* $p \in \Theta_e$ *we have* $L_p \approx R_p$*.*

Given an open reconfigurable net as above, the state of the net consists of a pair (Z, u) where Z is an open net and u is a marking of Z. Its behaviour is defined by the following possible moves:

– *firing*
  if $u\ [x\rangle\ _Z u'$ then

$$(Z, u) \Rightarrow_{RZ} (Z, u')$$

– *reconfiguration*
  if $p \in \Theta$, $m : L_p \to Z$ is a proper match and $(Z, u) \Rightarrow^{m,p} (Z', u')$ then

$$(Z, u) \Rightarrow_{RZ} (Z', u')$$

Note that in the last case, if $p \in \Theta_e$ then $Z \approx Z'$.

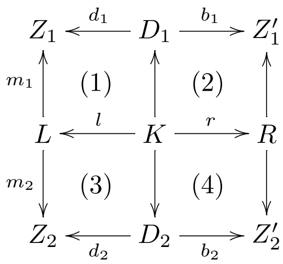
6 Behaviour consistent transformations

In this section we consider what we could call behaviour consistent reconfigurations, namely reconfigurations which preserve equivalence in the following sense.

Definition 19 (equivalence preserving matches). *Let Z be an open net, let $L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R$ a rewiring rule, where l preserves open places. A match $m : L \rightarrow Z$ is said* equivalence preserving *if transitions in $m(L)$ and in $Z - m(L)$ have distinct labels.*

Proposition 6 (consistent transformations). *Let Z_1 and Z_2 be open nets such that $Z_1 \sim_{id} Z_2$, let $L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R$ be a rewriting rule where l preserves open places and let $m_i : L \rightarrow Z_i$ be equivalence preserving matches. If Z_i rewrites to Z_i' along match m_i then $Z_1' \sim Z_2'$.*

Proof (Sketch). Let us consider the transformations of Z_i into Z_i'



and assume that all arrows are inclusions. Let $\mathcal{R}$ be an id-bisimulation between Z_1 and Z_2 [we also assume that the conditions to perform a rewriting step are satisfied, ensuring that all POs involve composable arrows].

Let us show that $\mathcal{R}' = \{((u_1 \downarrow d_1), (u_2 \downarrow d_2)) : (u_1, u_2) \in \mathcal{R}\}$ is a bisimulation between D_1 and D_2.

First observe that, since l is regular mono (preserves open places) also d_1 and d_2 satisfy the same property [to be checked]. Thus they are simulations.

Now, assume that for $(u_1, u_2) \in \mathcal{R}$ we have $(u_1 \downarrow d_1) \stackrel{\ell}{\longrightarrow}_{D_1} u_1'$. Let us show that also $(u_2 \downarrow d_2) \stackrel{\ell}{\longrightarrow}_{D_2} u_2'$ and $(u_1', u_2') \in \mathcal{R}'$.

We distinguish various cases:

1. $\ell = +_s, -_s$
   In this case, since d_1 preserves open places, we know that $u_1 \stackrel{\ell}{\longrightarrow}_{Z_1} u_1''$ and $(u_1'' \downarrow d_1) = u_1'$. Since Z_1 and Z_2 are bisimilar, the above implies that $u_2 \stackrel{\ell}{\longrightarrow}_{Z_2} u_2''$ and $(u_1'', u_2'') \in \mathcal{R}$. Hence, by reflection of the behaviour for open net embeddings, we have that

   $$(u_2 \downarrow d_2) \stackrel{\ell}{\longrightarrow}_{D_2} (u_2'' \downarrow d_2).$$

   But since $(u_1'', u_2'') \in \mathcal{R}$, by definition of $\mathcal{R}'$ we have that, as desired, $((u_1'' \downarrow d_1), (u_2'' \downarrow d_2)) \in \mathcal{R}'$.

2. $\ell = a$
   In this case, since d_1 is a marked open net embedding, the same firing can be done in Z_1, i.e.,

$$u_1 \xrightarrow{a}_{Z_1} u_1''$$

   Since Z_1 and Z_2 are bisimilar, the above implies that $u_2 \xrightarrow{a}_{Z_2} u_2''$ and $(u_1'', u_2'') \in \mathcal{R}$.
   By reflection of behaviour along open net embeddings, we know that the firing in Z_2 can be projected to a step in D_2. Now, since the match m_1 is equivalence preserving, we are sure that the transition executed in Z_2 does not have a counterimage in L. Hence it must have a counterimage in D_2 in a way that the projection of the step from Z_2 to D_2 leads to a firing

$$(u_2 \downarrow d_2) \xrightarrow{a}_{D_2} (u_2'' \downarrow d_2).$$

   Hence, as above, we conclude, that, as desired, $((u_1'' \downarrow d_1), (u_2'' \downarrow d_2)) \in \mathcal{R}'$. □

7 Conclusion and Related Work

Open nets, introduced in [2, 3], are a reactive extension of standard Petri nets which allows to model systems interacting with an unspecified environment. In this paper, firstly we have generalised the theory of open nets, including the characterisation of net composition using pushouts, to the case of marked nets. Several other approaches to Petri net composition and reactivity have been proposed in the literature (see, e.g., [4, 15, 7], to mention a few) and a detailed comparison with the open net model can be found in [3].

Next we have introduced the notions of strong and weak bisimilarity over open nets. Weak bisimilarity (and, as a particular case, also strong bisimilarity) is shown to be a congruence with respect to the colimit-based composition operation over open nets. To the best of our knowledge, this is the first time that a compositionality result is given for weak bisimilarity over Petri nets. This is different from what happens, e.g., in CCS [14], where weak bisimilarity fails to be a congruence: a more detailed comparison can be found in Appendix A. Weak bisimilarity for Petri nets with a composition operation is studied for example in [15], but it is not congruence, though a context closure allows one to get a congruence which is then characterised by means of a universal context. Our result about strong bisimilarity can be seen as a generalisation of those in [13, 18], which essentially are developed for a special kind of open nets, arising by viewing them as bigraphical reactive systems or as reactive systems over a cospan category. In the resulting reactive Petri net model there is no distinction between open input and output places. Furthermore the composition operation used in these papers does not allow synchronisation of transitions.

In the second part of the paper we have proposed a rewriting-based framework for Petri nets with reconfigurations. We have shown how our congruence results for the observational semantics can be used to identify classes of reconfigurations which do not alter the observational behaviour of the system. This is applied to a small case study of a workflow-like model of a travel agency. There we showed how abstract services can be replaced by more concrete implementations and how we can ensure that the behaviour of the full net is preserved under such operations.

The idea of using rewriting techniques for providing a reconfiguration mechanism for Petri nets has been already explored in the literature (see, e.g., reconfigurable nets of [1, 10] and high-level replacement systems applied to Petri nets in [16]). In future work, besides analysing the relationships between these approaches and ours, we will continue to study the notion of reconfigurable open nets and describe in more detail how reconfigurations can be triggered by the net itself (e.g., by reaching certain markings or by firing certain transitions, following an intuition similar to that of dynamic nets [5]).

References

1. E. Badouel, M. Llorens, and J. Oliver. Modeling concurrent systems: Reconfigurable nets. In H. R. Arabnia and Y. Mun, editors, *Proceedings of PDPTA'03*, volume 4, pages 1568–1574. CSREA Press, 2003.
2. P. Baldan, A. Corradini, H. Ehrig, and R. Heckel. Compositional modeling of reactive systems using open nets. In K.G. Larsen and M. Nielsen, editors, *Proceedings of CONCUR'01*, volume 2154 of *LNCS*, pages 502–518. Springer, 2001.
3. P. Baldan, A. Corradini, H. Ehrig, and R. Heckel. Compositional semantics for open Petri nets based on deterministic processes. *Mathematical Structures in Computer Science*, 15(1):1–35, 2004.
4. E. Best, R. Devillers, and J. G. Hall. The Petri box calculus: a new causal algebra with multi-label communication. In G. Rozenberg, editor, *Advances in Petri Nets*, volume 609 of *LNCS*, pages 21–69. Springer, 1992.
5. M.G. Buscemi and V.. Sassone. High-level Petri nets as type theories in the join calculus. In *Proceedings of FoSSaCS'01*, volume 2030 of *LNCS*. Springer, 2001.
6. H. Ehrig. Tutorial introduction to the algebraic approach of graph-grammars. In H. Ehrig, M. Nagl, G. Rozenberg, and A. Rosenfeld, editors, *Proceedings of the 3rd International Workshop on Graph-Grammars and Their Application to Computer Science*, volume 291 of *LNCS*, pages 3–14. Springer, 1987.
7. E. Kindler. A compositional partial order semantics for Petri net components. In P. Azema and G. Balbo, editors, *Application and Theory of Petri Nets*, volume 1248 of *LNCS*, pages 235–252. Springer, 1997.
8. S. Lack and P. Sobociński. Adhesive and quasiadhesive categories. *RAIRO – Theoretical Informatics and Applications*, 39(3), 2005.
9. J. Leifer and R. Milner. Deriving bisimulation congruences for reactive systems. In C. Palamidessi, editor, *Proceedings of CONCUR'00*, volume 1877 of *LNCS*, pages 243–258. Springer, 2000.
10. M. Llorens and J. Oliver. Introducing structural dynamic changes in Petri nets: Marked-controlled reconfigurable nets. In F. Wang, editor, *Proceedings of ATVA'04*, volume 3299, pages 310–323. Springer, 2004.
11. J. Meseguer and U. Montanari. Petri nets are monoids. *Information and Computation*, 88:105–155, 1990.
12. R. Milner. Bigraphical reactive systems. In K. G. Larsen and M. Nielsen, editors, *Proceedings of CONCUR'01*, volume 2154 of *LNCS*, pages 16–35. Springer, 2001.
13. R. Milner. Bigraphs for Petri nets. In J. Desel, W. Reisig, and G. Rozenberg, editors, *Lectures on Concurrency and Petri Nets*, volume 3098 of *LNCS*, pages 686–701. Springer, 2003.
14. Robin Milner. *A Calculus of Communicating Systems*. Springer, 1980. LNCS 92.
15. M. Nielsen, L. Priese, and V. Sassone. Characterizing Behavioural Congruences for Petri Nets. In *Proceedings of CONCUR'95*, volume 962 of *LNCS*, pages 175–189. Springer, 1995.

16. J. Padberg, H. Ehrig, and L. Ribeiro. High level replacement systems applied to algebraic high level net transformation systems. *Mathematical Structures in Computer Science*, 5(2):217–256, 1995.
17. W. Reisig. *Petri Nets: An Introduction.* EATCS Monographs on Theoretical Computer Science. Springer, 1985.
18. V. Sassone and P. Sobocinski. A congruence for Petri nets. In *Proceedings of PNGT'04*, number 127(2) in Electronic Notes in Computer Science, pages 107–120. Elsevier Science, 2005.
19. Glynn Winskel. Event structures. In *Petri Nets: Applications and Relationships to Other Models of Concurrency*, pages 325–392. Springer, 1987. LNCS 255.

A Comparison to CCS

We give some hints as to why weak bisimilarity is a congruence in the case of open nets, but not in CCS [14]. Remember that a classical counterexample for CCS is as follows: $p_1 = \tau.a.0 \approx a.0 = p_2$, but $q_1 = \tau.a.0 + b.0 \not\approx a.0 + b.0 = q_2$. The reason for the latter inequality is that q_1 can do a τ and become $a.0$, while q_2 cannot mimic this step.

Fig. 11 shows a similar situation of nondeterministic choice for open nets, where τ is the only unobservable label. However, note that here the two nets Z_1 (corresponding to $\tau.a.0$) and Z_1' (corresponding to $a.0$) are *not* weakly bisimilar. Whenever the τ-transition is fired in Z_1, resulting in the marking m_1, this can not be mimicked in Z_1' by staying idle, since then in Z_1' a transition with label $-_{s_1'}$ is possible, while a transition labelled $-_{s_1}$ is not possible for the net Z_1 with marking m_1. Also note that the places s_1 respectively s_1' have to be output open in order to allow composition with the net Z_2.

That means that for open nets we are always able to observe the first invisible action in an open component, which is reminiscent of the definition of observation congruence (denoted by $\approx^c$) in CCS: two processes p, q are called observation congruent whenever they are weakly bisimilar, with the additional condition that whenever the first step of p is a τ-action, then it has to be answered by at least one τ-action of q (and vice versa). In both cases it is only the first τ-action that can be observed but not the subsequent ones.

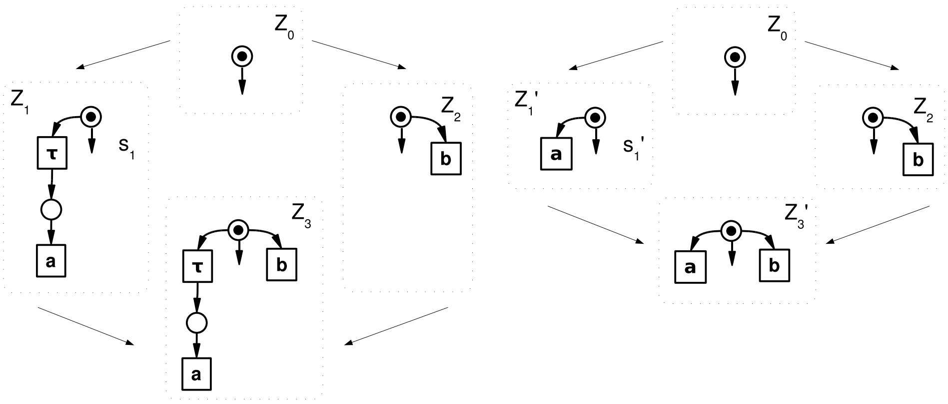


Fig. 11. Two pushouts for the comparison to CCS.