Binding bigraphs as symmetric monoidal closed theories

Tom Hirschowitz, Aurélien Pardon

Binding bigraphs as symmetric monoidal closed theories

Tom Hirschowitz[1] and Aurélien Pardon[2]

[1] CNRS, Université de Savoie `tom.hirschowitz@univ-savoie.fr`
[2] ENS Lyon `aurelien.pardon@ens-lyon.fr`

Abstract. Milner's bigraphs [1] are a general framework for reasoning about distributed and concurrent programming languages. Notably, it has been designed to encompass both the π-calculus [2] and the Ambient calculus [3].
This paper is only concerned with bigraphical syntax: given what we here call a *bigraphical* signature $\mathcal{K}$, Milner constructs a (pre-) category of bigraphs $\mathbf{Bbg}(\mathcal{K})$, whose main features are (1) the presence of *relative pushouts* (RPOs), which makes them well-behaved w.r.t. bisimulations, and that (2) the so-called *structural* equations become equalities. Examples of the latter are, e.g., in π and Ambients, renaming of bound variables, associativity and commutativity of parallel composition, or scope extrusion for ν-bound names. Also, bigraphs follow a scoping discipline ensuring that, roughly, bound variables never escape their scope.
Here, we reconstruct bigraphs using a standard categorical tool: *symmetric monoidal closed* (SMC) theories. Our theory enforces the same scoping discipline as bigraphs, as a direct property of SMC structure. Furthermore, it elucidates the slightly mysterious status of so-called *edges* in bigraphs. Finally, our category is also considerably larger than the category of bigraphs, notably encompassing in the same framework terms and a flexible form of higher-order contexts.

1 Overview

A central object of study in universal algebra is a *many-sorted algebraic theory.* It is specified by first giving a *signature*—a set of sorts X and a set Σ of operations with arities—together with a set of *equations* over that signature. For example, the theory for monoids is specified by taking only one sort x, and operations $m : x \times x \to x$ and $e : 1 \to x$, together with the usual associativity and unitality equations. We may equally well view this signature as given by a graph

$$(x \times x) \xrightarrow{m} x \xleftarrow{e} 1 \tag{1}$$

with vertices in the free monoid generated by X, which happen to be exactly the objects of the free category with finite products generated by X.

In this paper, we follow the same route, but replacing from the start finite products with SMC structure. Thus, an SMC signature is given by a set of sorts

X together with a graph whose vertices are objects of the free SMC category generated by X. Instead of cartesian product, we have the logical connectives of Girard's [4,5] Intuitionistic Multiplicative Linear Logic (henceforth IMLL): a tensor product $\otimes$, its right adjoint $\multimap$, and its unit I.

Here, we translate any bigraphical signature $\mathcal{K}$ into an SMC theory $\mathcal{T}_{\mathcal{K}}$, and then construct a faithful functor $\mathsf{T} : \mathbf{Bbg}(\mathcal{K}) \to S(\mathcal{T}_{\mathcal{K}})$, where $S(\mathcal{T}_{\mathcal{K}})$ is the free (or initial) SMC category generated by $\mathcal{T}_{\mathcal{K}}$. This functor is moreover essentially injective on objects (i.e., two objects with the same image are isomorphic).

In the category $S(\mathcal{T}_{\mathcal{K}})$, whose construction is essentially due to Trimble [6], morphisms are very much like intuitionistic multiplicative linear logic *proof nets* [4,5]: they are kind of graphs, whose *correctness* is checked by (a mild generalisation of) the well-known Danos-Regnier criterion [7]. And this criterion turns out to precisely enforce the same scoping discipline as bigraphs: our functor T induces an isomorphism on closed terms, i.e., of hom-sets

$$S(\mathcal{T}_{\mathcal{K}})(I, t) \cong \mathbf{Bbg}(\mathcal{K})(I, t),$$

where I is the unit of tensor product, and t is a particular object representing terms[3]. Our construction thus provides a logical explanation for the treatment of scope in bigraphs. Even more: our functor is not full, which means that SMC structure locally relaxes Milner's constraints on scope, preserving the overall discipline (i.e., the closed terms).

Furthermore, the status of so-called *edges* in bigraphs is fully elucidated: we translate differently *bound* edges (used for name restriction, much like ν in the π-calculus) and *free* edges (used for linking so-called *binding ports* to their peers). In the former case, we translate the edge into a ν node (which may also be understood as representing the private name in question); in the latter case, we simply remove the edge, and rely on our use of directed graphs to represent the flow from the binding port to its peers.

Finally, our category is also considerably larger than the category of bigraphs. Notably, it contains both the equivalent of terms and a kind of multi-hole, higher-order, binding contexts, all happily cohabiting in the same category $S(\mathcal{T}_{\mathcal{K}})$.

Related work The construction of the free SMC category generated by an SMC theory is essentially due to Trimble [6], followed by others [8,9,10,11]. The construction we use is a variant of Hughes' [10] construction, defined in our joint work with Richard Garner [12]. It was known that SMC (or cartesian closed) structure precisely represents various kinds of variable binding [13,14,15,16].

Damgaard and Birkedal [17] precisely axiomatise the category of bigraphs as an equational theory over a term language with variable binding. Our work may be seen as an essentially algebraic counterpart of theirs (which relies on α-equivalence). But our translation also provides a new viewpoint, tracing a path from the initial ingredients of distributed and concurrent languages to the graphical representation.

[3] We cheat a little here, see the actual result Lemma 3.

Future work On the down side, we address the category of *abstract* bigraphs, not the *precategory* of concrete bigraphs (only the latter having RPOs). Briefly, the morphisms of our category $S(\mathcal{T}_{\mathcal{K}})$ are actually equivalence classes of proof nets modulo Trimble's [6] *rewiring* relation. We briefly discuss this in Section 4.4.

Another natural research direction from this paper concerns the dynamics of bigraphs. Our hope is that Bruni et al.'s [15] very modular approach to dynamics may be revived, and work better with SMC structure than with cartesian closed structure. Specifically, with SMC structure, there is no duplication at the static level, which might simplify matters.

Finally, it might be fruitful to play with the categorical structure we use, to better explain the use of SMC structure, or possibly find better structures. In particular, Section 2.3 optimises the presentation by inlining the commutative monoid object structure of the sort t. It seems useful to look for a similar simplification on the sort v for variables, inlining its cocommutative comonoid. Indeed, this is likely to occur in many applications of our framework, those where variables are not restricted to a linear usage. But then, it might be useful to build this into the categorical structure. In other words, if we eventually give up linearity for v by adding new operations, why not accept this from the start and tune the categorical structure accordingly? In the same vein, it might be instructive to investigate (analogues of) RPOs in our setting, and search for those categorical structures having them.

Structure of the paper In Section 2, we introduce our variant of SMC theories, and their associated free SMC categories. We include a specialisation to the case where a sort is equipped with a commutative monoid object structure. In Section 3, we recall the definition of bigraphs, defining along the way our translation of bigraphical signatures. In Section 4, we construct our functor from bigraphs to the corresponding free SMC category, and show that it is an isomorphism on closed terms. Finally, we sketch the variant dealing with concrete bigraphs.

2 Symmetric monoidal closed theories

This section reviews the graphical presentation of SMC theories and models; for a more detailed version with proofs, we refer to our note [12].

2.1 The free symmetric monoidal closed category over a set

First, recall that a *monoidal* category is a category $\mathcal{C}$ with a functor $\mathcal{C} \times \mathcal{C} \xrightarrow{\otimes} \mathcal{C}$ called *tensor* product (and written in infix notation) and an object I, equipped with associativity and unit natural isomorphisms

$$A \otimes (B \otimes C) \xrightarrow{\alpha} (A \otimes B) \otimes C \qquad A \otimes I \xrightarrow{\rho} A \qquad I \otimes A \xrightarrow{\lambda} A$$

satisfying so-called coherence conditions [18]. It is *symmetric monoidal* when it is furthermore equipped with a natural isomorphism $(A \otimes B) \xrightarrow{\gamma} (B \otimes A)$ such that $\gamma^{-1}_{A,B} = \gamma_{B,A}$. It is finally *symmetric monoidal closed* (SMC) when for every object A, tensor product with A (e.g., on the right $- \otimes A$) has a right-adjoint, usually denoted by $A \multimap -$. This means that for each A, there is a natural bijection $\mathcal{C}(B \otimes A, C) \cong \mathcal{C}(B, A \multimap C)$.

A (strict) SMC functor $\mathcal{C} \to \mathcal{D}$ between two SMC categories is a functor (strictly) preserving these data. This defines a category SMCCat, which has a forgetful functor U to the category Set of sets and functions: it sends each SMC category to its set of objects, and each SMC functor to the corresponding function on objects. Trimble [6], followed by others, has shown (as a particular case, see below) that U has a left adjoint S sending each set X to an SMC category $S(X)$, free (or initial) in the sense that for any other SMC category $\mathcal{C}$ and function $X \xrightarrow{f} U(\mathcal{C})$, there is a unique SMC functor $S(X) \xrightarrow{f^*} \mathcal{C}$ such that f decomposes as $X \xrightarrow{\eta} US(X) \xrightarrow{U(f^*)} U(\mathcal{C})$.

How does $S(X)$ look like? Among its various characterisations, we find Hughes' easiest to grasp: it has objects the formulae of Girard's Intuitionnistic Multiplicative Linear Logic [4,5] (henceforth IMLL), described by the grammar

$$A, B, \ldots ::= x \mid I \mid A \otimes B \mid A \multimap B \qquad\qquad (\text{where } x \in X),$$

and morphisms $A \to B$ special graphs linking the leaf occurrences of A and B, which we call *ports*. Specifically, each such port is polarised, according to the number of times it goes left in a $\multimap$. Equivalently, writing $\overline{A}$ for the classical MLL formula equal to A, i.e., written using only $\otimes$, $⅋$, 1, $\perp$, and negation on atoms in X, polarity of a port in A is the polarity of the corresponding atom in $\overline{A}$. A morphism is then a function $A^+ + B^- \to A^- + B^+$, so that ports in $A^+ + B^-$ are *globally* negative, the others being globally positive. Our function is then required to link (i) each globally negative port labeled $x \in X$ (or x port) to a globally positive x port, all this bijectively, (ii) and additionally, each globally negative I port to any globally positive port.

But, crucially, not all such graphs qualify as morphisms of $S(X)$: they have to satisfy the Danos-Regnier [7] criterion, which goes as follows. For a classical MLL formula A, a switching is a graph obtained by removing in its abstract syntax tree exactly one premise edge of each $⅋$. Now, for a candidate morphism $A \xrightarrow{f} B$, a *switching* is a graph obtained by gluing f (seen as an undirected graph) along the ports with (i) a switching of the dual[4] $\overline{A}^{\perp}$ of $\overline{A}$, and (ii) a switching of $\overline{B}$. A candidate graph is a morphism, or is *correct* iff all its switchings are trees.

Finally, morphisms are quotiented by Trimble *rewiring*: a morphism *rewires* to another by changing the target of an edge from some globally negative I port, as soon as this preserves correctness. Rewiring is the smallest equivalence relation generated by this relation.

We now gradually extend the construction of $S(X)$ to signatures, then arbitrary theories.

---

[4] This is in the sense of De Morgan *duality* between $\otimes$ and $⅋$, I and $\perp$, and x and $x^{\perp}$.

2.2 Symmetric monoidal closed theories

Symmetric monoidal closed signatures An SMC *signature* is a pair (X, Σ) of a set X of sorts and a graph Σ with vertices in $S(X)$, as recalled in the previous section. A morphism of signatures $(X, \Sigma) \to (Y, \Sigma')$ is given by a function $f : X \to Y$, together with a morphism $\Sigma \to \Sigma'$ of graphs, whose vertex component is $S(f)$.

We then mimick Lawvere [19] in defining a *model* of such a signature to consist of an SMC category $\mathcal{C}$, with a function $M : X \to \mathrm{ob}\,\mathcal{C}$, and for each operation $f : A \to B$ a morphism $M_f : A \to B$ in $\mathcal{C}$. (A and B actually denote their images under the free extension $M^* : S(X) \to \mathcal{C}$ of M.) Morphisms of models are defined in the expected way, as strict SMC functors preserving the operations.

As in the case of a set X, there turns out to be a free model for each SMC signature. The construction is essentially due to Trimble [6], but we will use a variant, based on Hughes' [10] construction of $S(X)$, whose detailed exposition may be found in our note [12]. Here, we only briefly sketch this, and then directly put it to use to recover bigraphs.

The category $S(X, \Sigma)$ is much like $S(X)$, except that morphisms may contain *cells*, with one kind of cell per edge in Σ. In morphisms, each cell offers new ports to connect, one per leaf occurrence of its *type*, i.e., $A \multimap B$ for an edge $A \to B$. But this $A \multimap B$ counts as the domain of the morphism, which is thus a function $A^+ + C^+ + B^- \to A^- + C^- + B^+$, with C the set of cells.

The Danos-Regnier criterion extends by decreeing that a switching of a morphism is as above, but replacing each cell $A \to B$ with a a switching of $\overline{A \multimap B}^{\perp}$. Moreover, since we have cells, morphisms are both considered equivalent modulo Trimble rewiring and modulo the choice of cells.

Taking $X = \{x, y\}$ and $\Sigma = \{\alpha : x \to x \otimes y, \beta : y \otimes (x \multimap y) \to y\}$, a correct morphism of $S(X, \Sigma)$ is pictured in Fig. 1. The dotted link from the globally negative I port can be rewired to any globally positive port without violating the Danos-Regnier criterion; the morphims obtained are thus equivalent by Trimble rewiring.

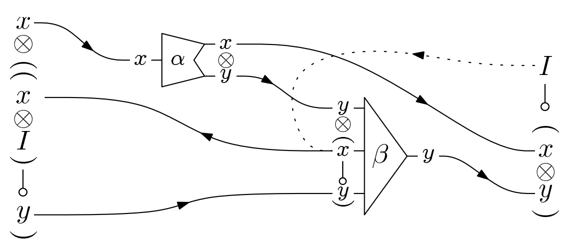


Fig. 1. A morphism of $S(X, \Sigma)$.

Symmetric monoidal closed theories Finally, we extend the construction to SMC theories: define a theory $\mathcal{T}$ to be given by a signature (X, Σ), together with a set $E_{A,B}$ of equations between morphisms in $S(X, \Sigma)(A, B)$, for each IMLL formulae A, B. The free SMC category $S(\mathcal{T})$ generated by such a theory is defined in our note [12] to be the quotient of $S(X, \Sigma)$ by the equations. Constructing the quotient graphically is more direct than could have been feared: we first define the binary predicate $f_1 \sim f_2$ relating two morphisms $C \overset{f_1,f_2}{\rightrightarrows} D$ in $S(X, \Sigma)$ as soon as each f_i decomposes as

$$C \xrightarrow{\cong} I \otimes C \xrightarrow{\ulcorner g_i \urcorner \otimes C} (A \multimap B) \otimes C \xrightarrow{f} D$$

with a common f, with $(g_1, g_2) \in E_{A,B}$ and where $\ulcorner g \urcorner$ is the currying of g. Then, we take the smallest generated equivalence relation, prove it stable under composition, and quotient $S(X, \Sigma)$ accordingly, which yields the free SMC category $S(\mathcal{T})$ generated by the theory $\mathcal{T} = (X, \Sigma, E)$.

2.3 Commutative monoid objects

We now slightly extend the results of our note [12] to better handle the special case of commutative monoids objects. This will be useful in our translation of bigraphs, where parallel composition and **0** have a commutative monoid object structure. Assume a theory (X, Σ, E) where a sort t is equipped with two operations m and e as in (1), with equations making it into a commutative monoid object (m is associative and commutative, e is its unit). Further assume that m and e do not occur in other equations.

Let Σ' be the result of removing the operations m and e in Σ. We define a relaxed version of our morphisms where each globally negative t port is connected to a globally positive one, but not necessarily bijectively. This defines a category isomorphic to $S(X, \Sigma)$, in which the operations m and e are built into the linking. The isomorphism is pictured in Fig. 2.

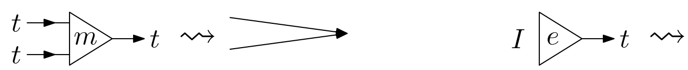


Fig. 2. Contracting m cells and deleting e cells.

3 Binding bigraphs and the translation of signatures

We now proceed to recall (a mild variant of) some definitions from Milner [1], along which we give our translation of bigraphical signatures $\mathcal{K}$ into SMC theories $\mathcal{T}_{\mathcal{K}}$. We then turn to our translation from the corresponding category of bigraphs to the free model $S(\mathcal{T}_{\mathcal{K}})$.

3.1 Signatures

Definition 1. *A* bigraphical (binding) signature *is a 4-uple* $(\mathcal{K}, B, F, A)$ *where* $\mathcal{K}$ *is a set of* controls*,* $B, F : \mathcal{K} \to \mathbb{N}$ *are maps providing a* binding *and a* free *arity for each control and* $A \subseteq \mathcal{K}$ *is a set of* atomic *controls.*

We fix such a bigraphical signature $\mathcal{K}$ for the rest of the paper. This signature can be translated into a SMC signature $\Sigma_{\mathcal{K}}$ over two sorts $\{t, v\}$. It consists of the following *structural* operations, accounting for the built-in structure of bigraphs:

$$\begin{tikzcd}
 & & & v \arrow[dl, "w"'] \arrow[dr, "c"] & \\
t \otimes t \arrow[dr, "|"] & & I \arrow[dl, "\mathbf{0}"'] \arrow[dr, "\nu"] & & v \otimes v \\
 & t & & v &
\end{tikzcd}$$

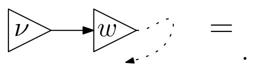
plus, for all controls k, a *logical* operation

$$(v^{\otimes B(k)} \multimap x) \otimes v^{\otimes F(k)} \xrightarrow{K_k} t$$

where $x = I$ if k is atomic and $x = t$ otherwise.

We call $\mathcal{T}_{\mathcal{K}}$ the theory consisting of the operations in $\Sigma_{\mathcal{K}}$, with the equations making

- (v, c, w) into a cocommutative comonoid object (c is coassociative, cocommutative, and w is its unit),
- $(t, |, \mathbf{0})$ into a commutative monoid object, and
- ν and w annihilate each other, as in



We now proceed to describe the category $\mathbf{Bbg}(\mathcal{K})$ of *abstract binding bigraphs* over $\mathcal{K}$, which we relate in Section 4 to the free model $S(\mathcal{T}_{\mathcal{K}})$ of $\mathcal{T}_{\mathcal{K}}$.

3.2 Interfaces

We now assume an infinite and totally ordered set $\mathcal{X}$ of *names*.

Definition 2. *A* bigraphical (binding) interface *is a triple* (n, X, loc) *where* n *is a finite ordinal,* X *a finite set of names and* $loc : X \to n + \{\bot\}$ *a function called* locality map*.*

A name x is said *global* if $loc(x) = \bot$ and *local* or *located at* i when $loc(x) = i \in n$.

Bigraphical interfaces are the objects of the category $\mathbf{Bbg}(\mathcal{K})$. We define a function T from these objects to IMLL formulas, i.e., the objects of $\mathcal{L}$, by:

$$\mathsf{T} : (n, X, loc) \mapsto v^{\otimes n_g} \multimap \bigotimes_{i \in n} (v^{\otimes n_i} \multimap t)$$

where $n_g = |loc^{-1}(\bot)|$ and for all $i \in n$, $n_i = |loc^{-1}(i)|$. The ordering of $\mathcal{X}$ induces a bijection between X and v leaves in the formula.

In [20], Milner presents a slight generalisation of binding bigraphs, where names have multiple locality. Some interfaces cannot be simply translated into IMLL formulas as before, e.g., if x is located in 0 and 1 and y in 1 and 2, this dependency cannot be expressed directly in an IMLL formula.

3.3 Place graph

Let n and m be two finite ordinals.

Definition 3. *A* place graph $(V, \mathit{ctrl}, \mathit{prnt}) : n \to m$ *is a pair where:*

- V *is a finite set of* nodes*,*
- $\mathit{ctrl} : V \to \mathcal{K}$ *is a function called* control map *and*
- $\mathit{prnt} : n + V \to V + m$ *is an acyclic function called* parent map *whose image does not contain any atomic node.*

The ordinals n and m index respectively the *sites* and *roots*. A node is said *barren* if it has no preimage under the parent map (atomic nodes are thus necessarily barren).

The relation $\prec$ over sites, roots and nodes defined by:

$$x \prec y \iff \exists k > 0\,,\ \mathit{prnt}^k(x) = y$$

is a (strict) partial order. The maximal elements of $\prec$ are the roots; the minimal elements are the barren nodes (including atomic nodes) and the sites.

3.4 Link graph

Let X and Y be two finite sets of names.

Definition 4. *A* link graph $(V, E, \mathit{ctrl}, \mathit{link}) : X \to Y$ *is a tuple where:*

- V *is a finite set of* nodes*,*
- E *is a finite set of* edges*,*
- $\mathit{ctrl} : V \to \mathcal{K}$ *is a* control map *and*
- $\mathit{link} : P + X \to E + Y$ *is a function called the* link map

with P being the set of ports*, i.e., the coproduct of binding ports defined by* $P_B = \coprod_{v \in V} B(\mathit{ctrl}(v))$ *and free ports* $P_F = \coprod_{v \in V} F(\mathit{ctrl}(v))$*. Moreover, link must satisfy the* binding rule*:*

For all binding ports $p \in P_B$*,* $\mathit{link}(p) \notin Y$*.*

We define the *binders* of N to be the local names of Y (located at a root) and the binding ports (located at a node) P_B.

Two distinct *points* (i.e., two elements of $P + X$) x and y are *peers* when $\mathit{link}(x) = \mathit{link}(y)$. An edge is *idle* when it has no preimage under the link map.

3.5 Abstract binding bigraphs

Let $U = (n, X, \mathit{loc})$ and $W = (m, Y, \mathit{loc}')$ be two bigraphical interfaces.

Definition 5. *A bigraph* $G = (V, E, \mathit{ctrl}, \mathit{prnt}, \mathit{link}) : U \to W$ *is a tuple where:*

- $(V, \mathit{ctrl}, \mathit{prnt}) : n \to m$ *is a place graph,*

– $(V, E, \mathit{ctrl}, \mathit{link}) : X \to Y$ *is a link graph,*
– G *satisfies the* scope rule*:*

If p is a binder located at w, then each of its peers is located at some $w' \prec w$.

The scope rule ensures that no binding port p is peer of a name in Y, hence $\mathit{link}(p)$ has to be an edge. Moreover, by acyclicity of prnt, no two binding ports may be peers, hence edges are linked to at most one binding port. The set of edges may thus be decomposed into a set of *free* edges E_F (without binding port) and a set of *bound* edges E_B in one-to-one correspondence with P_B by the link map: $E = E_F \uplus E_B \cong E_F + P_B$.

Finally, two bigraphs are *lean-support* equivalent when after discarding their idle edges, there is an isomorphism between their sets of nodes and edges preserving the structure.

Definition 6. *The category* $\mathbf{Bbg}(\mathcal{K})$ *of* abstract binding bigraphs over $\mathcal{K}$ *has bigraphical interfaces as objects and lean-support equivalence classes of bigraphs as morphisms.*

The composition of two bigraphs $U_1 \to U_2 \to U_3$ is defined by taking the coproduct of their nodes, edges and control maps and the composition of parent and link maps (modulo some bijections on sets), forgetting the roots/sites from U_2. Acyclicity of the parent map, and the binding and scope rules are preserved by composition.

4 Translation

We now want to show how a binding bigraph $G = (V, E, \mathit{ctrl}, \mathit{prnt}, \mathit{link}) : U \to W$ over $\mathcal{K}$ can be translated into a morphism $\mathsf{T}(G) : \mathsf{T}(U) \to \mathsf{T}(W)$ in the free model $\mathcal{L} = S(\mathcal{T}_{\mathcal{K}})$ of the SMC theory $\mathcal{T}_{\mathcal{K}}$.

Let $U = (n, X, \mathit{loc})$ and $W = (m, Y, \mathit{loc}')$. We will define the support C of $\mathsf{T}(G)$ as the disjoint union of:

– a *logical* support C containing a K_k cell for every node whose control is k and a ν cell for every free edge in E_F, and
– a *structural* support C' consisting of c and w cells, which we define below.

We then specify the graph $\mathsf{T}(G)$ for each sort in $\{t, v\}$ separately, and for I.

4.1 Places

First, since $(t, |, \mathbf{0})$ has a commutative monoid object structure, the representation of Section 2.3 applies: we just have to define a function from globally negative t ports to globally positive ones. Now, for any set X labeled in formulae, denote by X^+_t its set of positive t ports, and similarly for $X^{+,-}_{t,v,I}$. Now, we have:

- $C_t^+ \cong V$, because each type of cell K_k has one positive t port,
- $C_t^- \cong V_{na} \hookrightarrow V$, where V_{na} is the set of non-atomic nodes, because there is one globally negative t port for each non-atomic cell,
- $\mathsf{T}(U)_t^+ \cong n$, because for each $i \in n$ there is a positive t port in $\mathsf{T}(U)$,
- similarly, $\mathsf{T}(W)_t^+ \cong m$, and finally
- $\mathsf{T}(W)_t^- \cong \mathsf{T}(U)_t^+ \cong \emptyset$.

Our morphism $\mathsf{T}(G)$ is thus defined on the sort t by the function f_t:

$$\begin{tikzcd}
\mathsf{T}(U)_t^+ + C_t^+ + \mathsf{T}(W)_t^- \arrow[r, "\cong"] \arrow[d, dashed, "f_t"'] & \mathsf{T}(U)_t^+ + C_t^+ \arrow[r, "\cong"] & n + V \arrow[d, "prnt"] \\
\mathsf{T}(U)_t^- + C_t^- + \mathsf{T}(W)_t^+ & \mathsf{T}(V)_t^+ + C_t^- \arrow[l, "\cong"'] & m + V_{na}. \arrow[l, "\cong"']
\end{tikzcd}$$

4.2 Links

The function f_v for v requires more work, and involves defining the structural support C'. Recall that the data is the function $link : P \uplus X \to E \uplus Y$.

We start with an informal description of f_v based on Fig. 3, in which bold arrows come from binders. First, we deal with points sent to edges. There are two kinds of edges.

First, we understand each free edge e as the creation of a fresh name, and each free point p in $P_F \uplus X$ sent to e as an occurrence of this free name. Accordingly, e is replaced by its ν cell in C, and each p becomes a v port in $\mathsf{T}(U)^- + C^-$. We hence link the v port of the ν cell to each corresponding p, through a tree of c and w cells, as depicted in the bottom row.

Second, we understand each bound edge e as an indirection to its binding port $p_0 \in P_B$, itself understood as a bound name. We further understand each free peer $p \in P_F \uplus X$ of p_0 as an occurrence of the bound name. Accordingly, we completely forget about e, p_0 becomes a v port in C^+, and each p becomes a v port in $C^- + \mathsf{T}(U)^+$, hence we link p_0 to each corresponding p, again through a tree of c and w cells.

Finally, points p not sent to an edge are sent to some name $y \in Y$. But each such p becomes a v port in $\mathsf{T}(U)^- + C^-$ and each such y becomes a v port in $\mathsf{T}(W)^-$, hence we link y to each p, again using c and w cells. This determines the structural support C', as well as f_v. Finally, for the I part f_I, each globally negative I port arises from a structural w cell. But in the above each cell is generated by *one* v port (the fresh name or the binder). In the former case, we may safely link our I port to any valid t port. In the latter, the binder occurs to the left of a $\multimap$, whose right-hand side is a t port, to which we safely link our I port.

More formally, observe from our translation of signatures and interfaces, plus the logical support C defined above, that:

- each free edge in E_F corresponds to one ν cell, hence to one port in C_v^+,

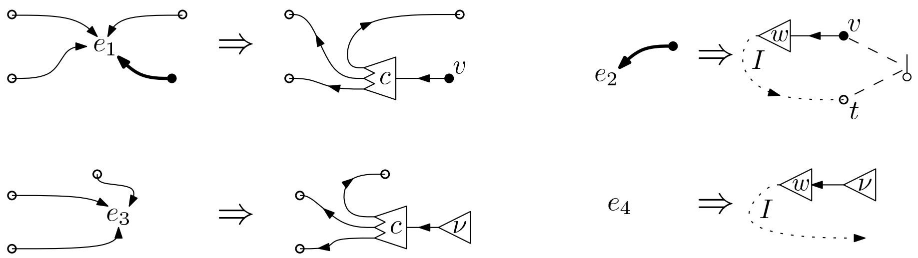


Fig. 3. Translation of *link*.

- each binding port in P_B corresponds to one negative occurrence of v in the domain of some cell in C, hence to one port in C_v^+,
- each local name in Y corresponds one port in $\mathsf{T}(W)_v^-$.

Thus, we have an isomorphism $E_F + P_B + Y \cong C_v^+ + \mathsf{T}(W)_v^-$. Similarly, free points in $P_F + X$ correspond to ports in $C_v^- + \mathsf{T}(U)_v^+$, i.e., $P_F + X \cong C_v^- + \mathsf{T}(U)_v^+$.

We may thus define a first function *link'* by:

$$\begin{tikzcd} C_v^- + \mathsf{T}(U)_v^+ \arrow[r, "\cong"] \arrow[d, dashed, "link'"'] & P_F + X \arrow[r, hook] & P_B + P_F + X \arrow[d, "link"] \\ C_v^+ + \mathsf{T}(W)_v^- & E_F + P_B + Y \arrow[l, "\cong"'] & E + Y. \arrow[l, "\cong"'] \end{tikzcd}$$

We then encode this function by a forest of c and w cells C' (as pictured in Fig. 4), to obtain a function $C_v^+ + C_v'^+ + \mathsf{T}(W)_v^- \xrightarrow{f_v} C_v^- + C_v'^- + +\mathsf{T}(U)_v^+$, which qualifies as the v part of our morphism. The rest follows similarly.

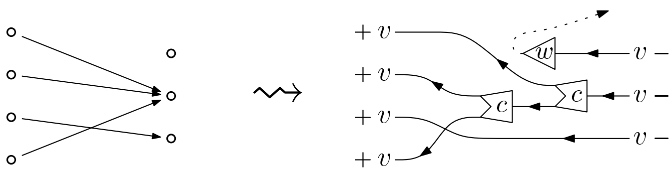


Fig. 4. Translation of a function using w and c cells.

This defines a function from bigraphs to candidate proof structures (respecting domain and codomain). We now show that it extends to a functor.

4.3 The functor

First, we prove that the image of a bigraph is correct, i.e., is a proper morphism.

Lemma 1. *All switchings of* $\mathsf{T}(G)$ *are connected.*

Proof. Consider a switching of $\mathsf{T}(G)$.

Given a site or a node p, we denote by $\mathsf{T}(p)$ the globally negative t port corresponding to it in the switching. If p is a root, then $\mathsf{T}(p)$ denotes the globally positive t port of its image.

Free ports of a node p (resp. local names of a site p') have their image (a globally positive v port) connected to $\mathsf{T}(p)$ (resp. $\mathsf{T}(p')$) as shown in Fig. 5. Moreover, either one globally negative v port (corresponding to a binding port) or the globally positive t port of the cell p is connected to $\mathsf{T}(p)$ by the switched formula.

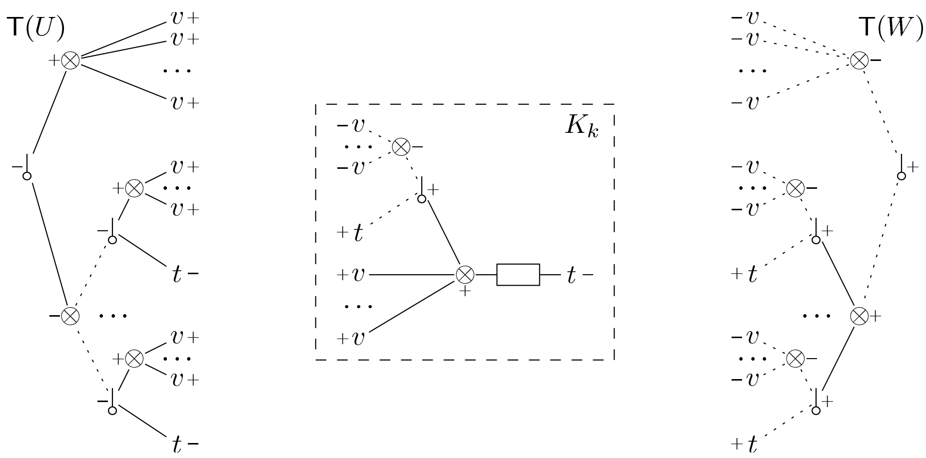


Fig. 5. Domain, codomain and a node of a switching.

We now prove by induction that all binding ports (located at a node or a root p) have their image connected to $\mathsf{T}(p)$. Let b be a binding port, and $\mathsf{T}(b)$ its image by T (a globally negative v port).

If b has no peers (this is necessarily the case if p is a barren node), then $\mathsf{T}(b)$ is connected to a w cell whose I port is connected to $\mathsf{T}(p)$.

If b has peers, then $\mathsf{T}(b)$ is connected, in the morphism, to their translations through a tree of c cells. But this tree is heavily switched and only connects $\mathsf{T}(b)$ to one globally positive v port f (whose preimage is) located, thanks to the scope rule, to a site or a node $p' \prec p$.

By induction f is connected to $\mathsf{T}(p')$ and $\mathsf{T}(p')$ is connected to $\mathsf{T}(p)$ through the (unswitched) parent map. Indeed, the parent map connects the t ports of cells between p' and p, and these cells have their t ports connected thanks to the induction hypothesis. The port $\mathsf{T}(b)$ and $\mathsf{T}(p)$ are thus connected.

Finally, we remark that:

– roots are connected to each other in the codomain's formula (by their t or v ports, see Fig 5),
– global variables of the domain are connected to a site (by the domain's formula, see Fig. 5) and
– remaining globally negative v ports (global variable of the codomain and ν cells) are connected to the other globally positive ports by a switched tree of c cells or a w cell.

We can conclude that all ports of our switching are connected.

The following seems known [21]:

Lemma 2. *Any switching of a morphism in $S(\mathcal{T}_{\mathcal{K}})$ is acyclic iff it is connected.*

Proof (sketch). One proves by induction on the domain and codomain formulae that the graph induced by the switching has one more vertex than it has edges.

Proposition 1. *The map $\mathsf{T} : \mathbf{Bbg}(\mathcal{K}) \to \mathcal{L}$ is a functor.*

Proof. T sends lean-support equivalent bigraphs to equivalent morphisms (in particular discarding idle edges corresponds to annihilation of ν and w), which we have just proved correct. The identity property is easily verified. The equations of $\mathcal{T}_{\mathcal{K}}$ defined in Section 3.1 ensure that T behaves well w.r.t. composition.

One sees at once that T is not full. For example, the morphism in Fig. 6 has no preimage – any such preimage would violate the scope rule. This example

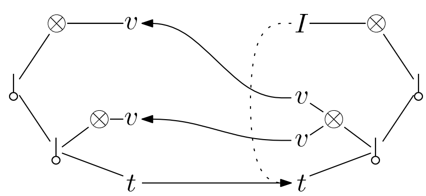


Fig. 6. A correct morphism violating the scope rule.

reflects that it is not necessary to distinguish global and local variables in a bigraph with only one site. Nevertheless, the notion of scope is preserved by T because closed morphisms can actually be translated into bigraphs. In $\mathbf{Bbg}(\mathcal{K})$, define the interfaces $I = (0, \emptyset, \widehat{\emptyset})$ and $t = (1, \emptyset, \widehat{\emptyset})$.

Lemma 3. *The functor T induces an isomorphism $S(\mathcal{T}_{\mathcal{K}})(I, t) \cong \mathbf{Bbg}(\mathcal{K})(I, t)$.*

Proof. Consider any $f : I \to t$. We have $\mathsf{T}(I) = (I \multimap I) \cong I$ and $\mathsf{T}(t) = I \multimap (I \multimap t) \cong t$, which justifies our "induces" above. We now define $G = (V, E, \mathit{ctrl}, \mathit{prnt}, \mathit{link}) : I \to t$ such that $\mathsf{T}(G) = f$.

Let the set of nodes V be the set of logical cells in f; the control map ctrl sends each K_k cell to $k \in \mathcal{K}$.

The set of edges is the coproduct of binding v ports in the support of f and of ν cells (where a v port is binding when it occurs to the left of a $\multimap$, e.g., a cell of type $((v \otimes v) \multimap t) \otimes v \to t$ has two binding ports).

The parent map $prnt : 0 + V \to 1 + V$ is exactly the restriction of f to t ports. The link map $link : P_B + P_F + \emptyset \to E + \emptyset$ is obtained from the restriction of f to v ports as follows. From any v port p, following the tree of contractions towards its root leads to a maximal globally positive v port in the support, which may be either a port from a ν cell, or a binding port of a logical cell. In each case, there is a corresponding edge e_p. Our link map sends each port p to e_p. Since in each tree there is only one root, the binding rule is respected.

We then prove that G is correct. Suppose that the parent map contains a cycle, then any switching where, for all cells of the cycle, the two t ports are connected contains this cycle. Suppose that the scope rule is not satisfied for a binder p and one of its peers p'. Then, in f, p is the root of a contraction tree with p' as a leaf: among the switchings connecting them, choose again one that connects both t ports of each logical cell: every logical cell then has a path to the root r (the t port in the codomain), which forms a cycle involving p, p', and r, hence contradicting correctness of f. The binding rule is automatically satisfied because the codomain has no name. An atomic node has no antecedent in the parent map because the corresponding cell in f has no globally positive t port.

All in all, we have

Theorem 1. *The functor* $\mathsf{T} : \mathbf{Bbg}(\mathcal{K}) \to \mathcal{L}$ *is faithful, essentially injective on objects, and surjective on* $\mathcal{L}(I, t)$.

It is however not full and far from surjective on objects.

4.4 Tuning the presentation

We end the paper with a brief discussion of concrete vs. abstract bigraphs. We could hope to recover Milner's concrete bigraphs, which differ from abstract bigraphs mainly in that they are not considered equivalent modulo the choice of cells. In our setting, we will want to quotient by Trimble rewiring and structural equations, which seems problematic, because neutrality of w w.r.t. contraction adds (or removes) a cell. We would thus have to consider our morphisms equivalent modulo choice of w cells, which is not that satisfactory.

Finally, within the full subcategory with objects the images of bigraphical interfaces, we observe that edges from (globally negative) occurrences of I may be rewired to any globally positive port without breaking correctness. Hence, such edges may be safely omitted. This brings us even closer to Milner's representation.

References

1. Jensen, O.H., Milner, R.: Bigraphs and mobile processes (revised). Technical Report TR580, University of Cambridge (2004)

2. Milner, R., Parrow, J., Walker, D.: A calculus of mobile processes. Information and Computation **100**(1) (1992) 1–40
3. Cardelli, L., Gordon, A.: Mobile ambients. In Nivat, M., ed.: Foundations of Software Science and Computational Structures. Volume 1378 of Lecture Notes in Computer Science., Springer (1998) 140–155
4. Girard, J.Y.: Linear logic. Theoretical Computer Science **50** (1987) 1–102
5. Girard, J.Y.: Linear logic: a survey. In: Proc. International Summer School of Marktoberdorf. F94, NATO Advanced Science Institute (1993) 63–112
6. Trimble, T.: Linear logic, bimodules, and full coherence for autonomous categories. PhD thesis, Rutgers University (1994)
7. Danos, V., Regnier, L.: The structure of multiplicatives. Archive for Mathematical Logic **28** (1989) 181–203
8. Blute, R., Cockett, J.R.B., Seely, R.A.G., Trimble, T.H.: Natural deduction and coherence for weakly distributive categories. Journal of Pure and Applied Algebra **13**(3) (1996) 229–296
9. Tan, A.: Full Completeness for Models of Linear Logic. PhD thesis, University of Cambridge (1997)
10. Hughes, D.J.D.: Simple free star-autonomous categories and full coherence. ArXiv Mathematics e-prints, math/0506521 (June 2005)
11. Lamarche, F., Strassburger, L.: From proof nets to the free *-autonomous category. Logical Methods in Computer Science **2**(4) (2006)
12. Garner, R., Hirschowitz, T., Pardon, A.: Symmetric monoidal closed theories. Available from `www.lama.univ-savoie.fr/~{}hirschowitz/papers/smcc.pdf`
13. Barber, A.: Linear Type Theories, Semantics and Action calculi. PhD thesis, University of Edinburgh (1997)
14. Barber, A., Gardner, P., Hasegawa, M., Plotkin, G.: From action calculi to linear logic. In: Annual Conference of the European Association for Computer Science Logic (CSL'97), Aarhus, August 1997, Selected Papers. Volume 1414 of Lecture Notes in Computer Science., Springer (1998) 78–97
15. Bruni, R., Montanari, U.: Cartesian closed double categories, their lambda-notation, and the pi-calculus. In: LICS '99: Proc. 14th Annual IEEE Symposium on Logic in Computer Science, IEEE Computer Society (1999) 246
16. Coccia, M., Gadducci, F., Montanari, U.: GS·Λ theories: A syntax for higher-order graphs. In: CTCS'02, Category Theory and Computer Science. Volume 69 of Electronic Notes in Theoretical Computer Science., Elsevier (2003) 83–100
17. Damgaard, T., Birkedal, L.: Axiomatizing binding bigraphs. Nordic Journal of Computing **13**(1–2) (2006) 58–77
18. Mac Lane, S.: Categories for the Working Mathematician. 2 edn. Number 5 in Graduate Texts in Mathematics. Springer (1998)
19. Lawvere, F.W.: Functorial semantics of algebraic theories. PhD thesis, Columbia University (1963)
20. Milner, R.: Bigraphs whose names have multiple locality. Technical Report TR603, University of Cambridge (2004)
21. Soloviev, S.: Connectedness and acyclicity in IMLL proof nets (June 2008) Message to the TYPES forum.