A Calculus for Graphs with Complement

Renata de Freitas[1], Paulo A.S. Veloso[2], Sheila R.M. Veloso[3], and Petrucio Viana[1]

[1] Institute of Mathematics, UFF: Universidade Federal Fluminense; Niterói, Brazil
[2] Systems and Computer Engin. Program, COPPE, UFRJ: Universidade Federal do Rio de Janeiro; Rio de Janeiro, Brazil
[3] Systems and Computer Engin. Dept., Faculty of Engineering, UERJ: Universidade do Estado do Rio de Janeiro; Rio de Janeiro, Brazil

Abstract. We present a system for deriving inclusions between graphs from a set of inclusions between graphs taken as hypotheses. The novel features are the extended notion of graph with an explicitly representation of complement, the more involved definition of the system, and its completeness proof due to the embedding of complements. This is an improvement on former work, where complement was introduced by definition. Our calculus provides a basis on which one can construct a wide range of graph calculi for several algebras of relations.

Keywords: Reasoning with Diagrams, Graph Calculus, Complement, Completeness, Algebras of Relations.

1 Introduction

Our understanding of basic *reasoning with diagrams* can be better grasped by observing Figure 1 (a). In this paper, we show how the basic idea underlying reasoning with diagrams can be nicely adapted to the case of "formulas" being "terms from a relation algebraic language" and "implies" meaning "the relation defined by term t_1 is a sub-relation of the one defined by term t_2, under a set Σ of hypotheses". In this case, certainly due to the conceptual proximity between *binary relations* and *graphs*, the diagrams that appear as the most appropriate to deal with are the *2-pointed labeled directed graphs* [1], and the picture in Figure 1 (a) converts to that in Figure 1 (b).

Graphs in the sense above mentioned have been used or, rather, have been proposed to be used as a tool in the investigation of relation algebraic formalisms from a long time ago. Usually, the graphs in general — not only those obtained from terms — have bigger expressive and proof powers and are reputed to be easier to use than the algebraic terms. Hence, by using graphs instead of terms, one can obtain results on graphs that remain true when restricted to the terms. Examples of this strategy in use and some of its developments can be found in the papers [1,6,4,7,5]. One can say that all these works, except [6], use graphs as auxiliary tools into the investigation of subsystems of relation algebras [15,20].

A.K. Goel, M. Jamnik, and N.H. Narayanan (Eds.): Diagrams 2010, LNAI 6170, pp. 84–98, 2010.
© Springer-Verlag Berlin Heidelberg 2010

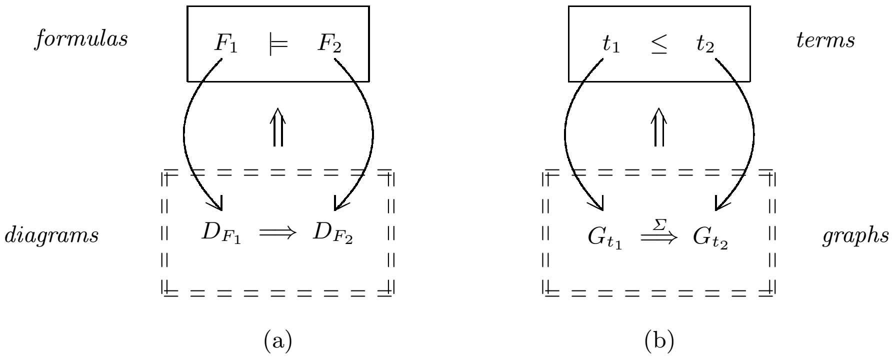

Fig. 1. Reasoning (a) with diagrams and (b) with graphs

Differently, Curtis and Lowe [6] suggest to investigate graph systems themselves as ordinary formal systems. Their main idea is to use the 2-pointed labeled directed graphs machinery as a formal system whose language has graphs as formulas and whose inference rules are applied to transform a graph into another. Under some general conditions, these transformation rules can be used to put the graphs into a certain normal form and the inclusions of graphs in normal form can be tested via an adequate notion of homomorphism between graphs. They also give hints of how to adapt their ideas to deal with a whole class of algebras built on the top of a lattice equipped with an associative binary operation compatible with the lattice operations.

We decided to investigate Curtis and Lowe ideas deeply and to develop logical systems having graphs as terms and inferences on graphs. In the previous papers [10,9,12,13] we investigate the use of graphs to decide the validities of languages representing just positive information. The next natural step was taken in [11] where we considered reasoning from hypotheses in a language where complementation is introduced by definition.

Here we present a system which improves the earlier ones on reasoning from hypotheses, having an explicit representation of complement. The basic intuitions are quite simple, leading to a playful and powerful system for deriving inclusions between graphs that are consequences of a set of inclusions between graphs taken as hypotheses. The novel features are the extended notion of graph with arcs labeled by boxes (to represent complement), the more involved definition of the system, and its completeness proof due to the embedding of complements. We leave for further investigation the use of our system to simplify previous reasoning about algebras of relations as well as to adapt our system to deal with algebras having a structured domain (cf. [11,12] for preliminary results in this direction).

The approach to reasoning with graphs we adopt here may be called the *logic systematic* approach: pictures are considered as ordinary terms of a (non-orthodox) logical system and a set of inference rules is provided for deriving pictures from pictures. This approach emphasizes notions of homomorphism for pictures, which are used to prove the inclusions and equalities. With no intention of being exhaustive, we would like to mention two other approaches in using pictures as a tool to help investigating and applying relational formalisms. The approach based on the *theory of allegories* [2,3,16] views pictures as arrows in a (unitary pretabular) allegory [8] and uses laws directly associated to the valid allegorical identities for transforming pictures. Results of the theory of allegories are used to show that two pictures can be proved equal by using the laws on pictures iff they represent the same relation. The approach based on the *rewriting systems* [17,18,19] endows pictures with a relational semantics, which allows them to be interpreted as terms of an algebraic language. A rewriting mechanism for pictures is built as a variant of the algebraic approach to graph rewriting. The way one can use rewriting sequences as proofs leads to a general and flexible tool for the proof of relational algebraic identities.

The structure of this paper is as follows. In Section 2, we present some intuitive ideas and examples to motivate the graph language and rules discussed in the other sections. In Section 3, we introduce more precisely our graphs, presenting its syntax and semantics, and defining the notion of consequence we deal with. We also present two schemes of axioms and two rules which can be used to transform a graph into another and prove soundness of the system. In Section 4, we indicate how one can characterize graph consequence by means of the axioms and rules, obtaining completeness of the calculus. Due to lack of space some minor details in the proofs were omitted.

2 Basic Ideas

We begin with some intuitive ideas behind our graph calculus, describing the aspect our 2-pointed labeled directed graphs have, how they can be used to represent relations, and how they can be transformed to represent inferences on relations. A graph is a finite set of slices. A slice consists of nodes, labeled arcs between nodes, and exactly two distinguished nodes we call input and output and represented by $-$ and $+$, respectively. A label is a relation symbol or a box. A box is a figure of the form $\boxed{G}$, where G is a graph. A box should be considered as a black-box, that means, the nodes, arcs, labels, and distinguished nodes of the graph it encloses do not count as nodes, arcs, labels, or distinguished nodes of the graph in which it occurs as a label. We identify a graph $\{S\}$, consisting of just one slice S, with the slice S.

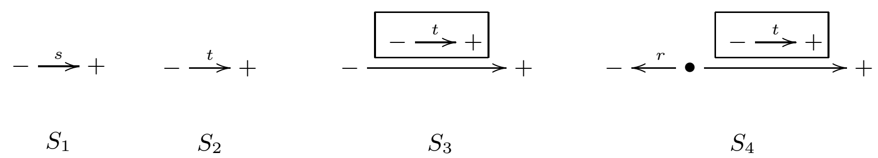

Fig. 2. Slices

Figure 2 shows slices S_1, S_2, S_3, and S_4. Slices S_1 and S_2 have single arcs, labeled by the relation symbols s and t, respectively. Slice S_3 has a single arc labeled by the box $\boxed{S_2}$. Slice S_4 has two consecutive opposite arcs, one labeled by the relation symbol r and the other labeled by the box $\boxed{S_2}$.

Figure 3 shows a single graph consisting of two slices, S_5 and S_6. Slice S_5 has two parallel paths from the non-distinguished node $\bullet$ to node the output $+$. One is the path $\bullet \xrightarrow{r} - \xrightarrow{s} +$ and the other is the arc $\bullet \xrightarrow{\boxed{S_2}} +$. Slice S_6 is like S_5, with arc $- \xrightarrow{\boxed{S_1}} +$ in place of arc $- \xrightarrow{s} +$.

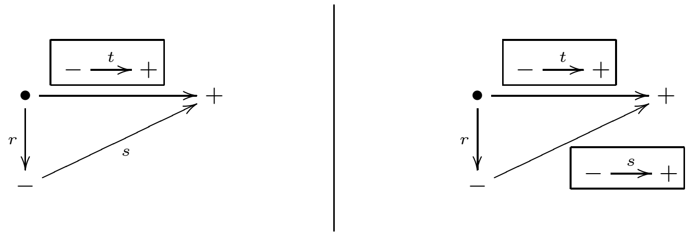

Fig. 3. Graph with two alternative slices S_5 and S_6, obtained from S_4

Now, we describe the way graphs, slices, and labels represent binary relations.

Given an arbitrary set M, a relation symbol represents an arbitrary binary relation on M. So, considering that the relation symbols r, s and t, referred above, represent binary relations on a base set M, the slices and graph depicted in Figures 2 and 3 also represent binary relations on M, according to the following ideas.

A labeled arc from a node u to a node v with label L represents a restriction imposed to u and v, namely that u and v should be related by the relation L. A slice represents the set of pairs satisfying the restrictions imposed to its input-output nodes. Thus, S_1 represents the set of pairs related by s, i.e. the relation s. Analogously, S_2 represents relation t.

A box represents the complement of the relation the graph it encloses represents. Hence, since S_3 represents the set of pairs related by $\boxed{S_2}$, we have that S_3 represents the relation t^{C}, the complement of t.

Consecutive arcs represent a concatenation of the restrictions each arc imposes on their input and output nodes, i.e. the "serialization" of the restrictions. Slice S_4 represents the set of pairs having an intermediate point $\bullet$ with which the input node is related by r and that is related to the output node by $\boxed{S_2}$. Hence, S_4 represents the relation $r^{\mathsf{T}} \circ t^{\mathsf{C}}$, the conposition of the transpose of r with the complement of t.

Parallel paths sharing the same extreme points represent simultaneous restrictions their extreme points should satisfy, i.e. the "parallelization" of the restrictions. So, S_5 represents the set of pairs related by both the relations $r^{\mathsf{T}} \circ t^{\mathsf{C}}$ and s, i.e. S_5 represents the intersection $(r^{\mathsf{T}} \circ t^{\mathsf{C}}) \cap s$. Similarly, S_6 represents the intersection $(r^{\mathsf{T}} \circ t^{\mathsf{C}}) \cap s^{\mathsf{C}}$.

In general, a slice imposes a set of restrictions on its input and output nodes that any pair of points in M should satisfy to be in the relation the slice represents.

Finally, a graph represents the relation which is the union of the relations represented by its slices. Thus, the graph in Figure 3 represents the union $\{(r^{\mathsf{T}} \circ t^{\mathsf{C}}) \cap s\} \cup \{(r^{\mathsf{T}} \circ t^{\mathsf{C}}) \cap s^{\mathsf{C}}\}$.

Now, we describe, by way of an example, how our graphs can be manipulated to represent inferences on relations. The ideas here presented give the intuitions behind our graph calculus and will lead us to a set of inference rules which will characterize it.

It is known that, for all relations r, s and t, the inclusion $r^{\mathsf{T}} \circ t^{\mathsf{C}} \subseteq s^{\mathsf{C}}$ follows from the inclusion $r \circ s \subseteq t$. Figure 4 is a proof of this fact in our graph calculus. It consists of a sequence $\langle G_1, G_2, G_3, G_4, G_5 \rangle$ of five graphs.

The graph G_1 is the single slice graph S_4 of Figure 2, that represents the relation $r^{\mathsf{T}} \circ t^{\mathsf{C}}$, i.e. G_1 represents the left-hand side of the inclusion we want to prove.

The graph G_2 is the two slices graph $\{S_5, S_6\}$ of Figure 3, obtained from graph G_1 by expanding slice S_4. Notice that slice S_5 and slice S_6 are obtained from slice S_4 by adding to it a new arc from x_4 to y_4, labeled by s and the box $\boxed{S_1}$, respectively. To show that this passage from S_4 to $\{S_5, S_6\}$ is justified, we consider that, in general, a graph represents alternatives a pair of points may satisfy in order to belong to the relation defined by the graph. Also, as usual, any pair of points is related by a given relation or by its complement. Thus, slice S_4 and the graph $\{S_5, S_6\}$ impose the same restrictions to any pair of nodes, representing the same relation.

The graph G_3 is also a two slices graph, $\{S_7, S_6\}$, obtained this time from the graph $\{S_5, S_6\}$, according to the following idea that allow us to use the hypothesis $r \circ s \subseteq t$ to transform the slice S_5. Since slice S_5 has a path from node $\bullet$ to the output node $+$ through the input node $-$ that represents the relation $r \circ s$ and since, by hypothesis, $r \circ s \subseteq t$, we are allowed to transform slice S_5 into slice S_7 by adding an arc labeled t from node $\bullet$ to the output node $+$.

Now, observe that slice S_7 of graph G_3 has two parallel arcs linking node $\bullet$ to the output node $+$, one labeled t and the other labeled $\boxed{S_2}$. So, according to our conventions on parallel paths, the points $\bullet$ and $+$ are simultaneously in the relations t and t^{C}, so that slice S_7 represents the empty relation. Thus, we can erase slice S_7 from the graph G_3 obtaining the single slice graph G_4 which consists of the remaining slice S_6.

Finally, note that, inside graph G_4 we can locate a copy of the slice S_8, as shown in Figure 5. This means that S_8 imposes no more restrictions than S_6 in defining a relation and so, S_6 may be considered to represent a sub-relation of S_8. Thus, we finally move from graph G_4 to graph G_5 in constructing our graph

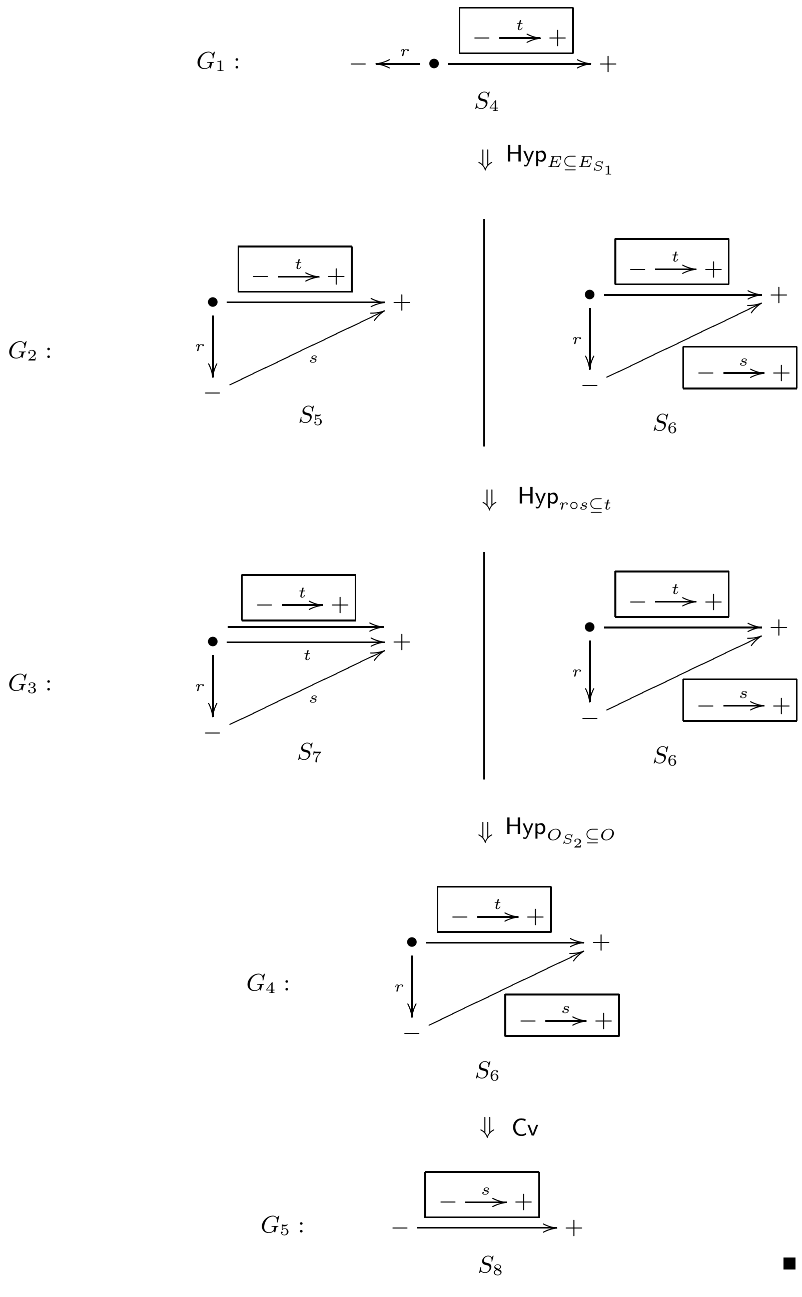

Fig. 4. Graph proof of $r \circ s \subseteq t \;\vdash\; r^{\mathsf{T}} \circ t^{\mathsf{C}} \subseteq s^{\mathsf{C}}$

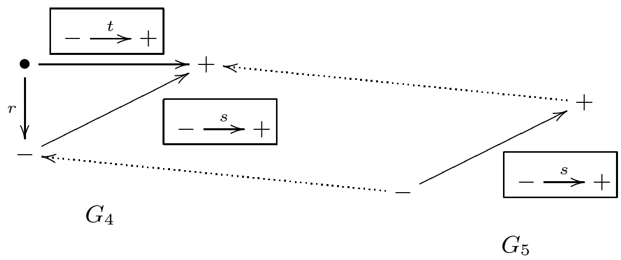

Fig. 5. Mapping slice S_8 to slice S_6

proof. The latter consists of a boxed slice which represents the relation s^{C}, i.e. G_5 represents the right-hand side of the implication we want to prove.

Summarizing, we have ilustrated above how graphs can be used to prove a valid inference involving two relational inclusions, the latter having occurrences of complement. The usual setting where formal reasoning on relations is performed is *equational logic*. There, all the statements are equalities between relational terms, and the only primitive inference rule is the high-school rule of replacing equals by equals. To emphasize the playful aspect of our approach, we present below an equational proof of the above inference, based on the usual equational axioms for relations.

Proposition 1. *$r \circ s \subseteq t$ implies $r^{\mathsf{T}} \circ t^{\mathsf{C}} \subseteq s^{\mathsf{C}}$, for all relations r, s, and t.*

Proof. We use the equational reasoning (Con), which assures that $=$ is a congruence relation with respect to the operations on relations; we apply some usual Boolean properties (BA), the distributivity of $\circ$ on $\cup$ (Dis), and the awckward axiom $(r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}) \cup s^{\mathsf{C}} = s^{\mathsf{C}}$ (Ax), which plays an important role in the axiomatization of relation algebras [20].

1.	$(r \circ s) \cap t = r \circ s$	(Hyp)
2.	$((r \circ s) \cap t)^{\mathsf{C}} = (r \circ s)^{\mathsf{C}}$	(1, Con)
3.	$(r \circ s)^{\mathsf{C}} \cup t^{\mathsf{C}} = (r \circ s)^{\mathsf{C}}$	(2, BA)
4.	$r^{\mathsf{T}} \circ ((r \circ s)^{\mathsf{C}} \cup t^{\mathsf{C}}) = r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}$	(3, Con)
5.	$(r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}) \cup (r^{\mathsf{T}} \circ t^{\mathsf{C}}) = r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}$	(4, Dis)
6.	$((r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}) \cup (r^{\mathsf{T}} \circ t^{\mathsf{C}})) \cup s^{\mathsf{C}} = (r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}) \cup s^{\mathsf{C}}$	(5, Con)
7.	$((r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}) \cup s^{\mathsf{C}}) \cup (r^{\mathsf{T}} \circ t^{\mathsf{C}}) = (r^{\mathsf{T}} \circ (r \circ s)^{\mathsf{C}}) \cup s^{\mathsf{C}}$	(6, BA)
8.	$s^{\mathsf{C}} \cup (r^{\mathsf{T}} \circ t^{\mathsf{C}}) = s^{\mathsf{C}}$	(7, Ax)

In the proof above we used the fact that, for relations, any inclusion $r \subseteq s$ is equivalent to the equalities $r \cap s = s$ and $s \cup r = r$. ∎

3 Graph Calculus

Now we formalize the intuitive ideas presented in Section 2 to define our graph calculus.

We start by presenting syntax. Our system has sets of 2-pointed labeled directed graphs as "terms" and inclusions between graphs as "formulas".

Nodes and labeled arcs are the building blocks of graphs. Hence, we first consider the sets INOD of individual *nodes* and RSYM of *relational symbols*, which we will keep fixed throughout.

A *slice*, typically denoted by S or T, is a structure (N, A, x, y), where N is a finite nonempty set of nodes; $A \subseteq N \times \mathfrak{L} \times N$ is a finite set of labeled *arcs* ($\mathfrak{L}$ is the set of all labels); x (*input*) and y (*output*) are, not necessarily distinct, distinguished nodes in N. An *arc* of A is a triple, denoted by uLv, with $u, v \in N$ and L being a label. A *label* is a relational symbol or a *box* $\boxed{G}$, where G is a concrete graph.

Concrete graphs are sets of slices defined by the following grammar.

$$G ::= \{S_L\} \mid E \mid I \mid O \mid G^{\mathsf{T}} \mid G \circ G \mid G \sqcap G \mid G \sqcup G,$$

where

$S_L = (\{x, y\}, \{xLy\}, x, y)$, with $x, y \in$ INOD and L being a label,
$E = \{(\{x, y\}, \emptyset, x, y)\}$, with $x, y \in$ INOD and $x \neq y$,
$I = \{(\{x\}, \emptyset, x, x)\}$, with $x \in$ INOD,
$O = \emptyset$.

The operations on concrete graphs are defined based on their analogous to slices, except for union. Given slices $S = (N, A, x, y)$, $S_1 = (N_1, A_1, x_1, y_1)$, and $S_2 = (N_2, A_2, x_2, y_2)$, we define

$S^{\mathsf{T}} = (N, A, y, x)$, the *transposition* of S,
$S_1 \circ S_2 = (N_1 \uplus N_2, A_1 \uplus A_2, x_1, y_2)\frac{x_2}{y_1}$, the *composition* of S_1 and S_2,
$S_1 \sqcap S_2 = (N_1 \uplus N_2, A_1 \uplus A_2, x_1, y_1)\frac{x_1}{x_2}\frac{y_1}{y_2}$, the *intersection* of S_1 and S_2.

Here, we use the *node substitution* notation $\frac{u}{v}$ for replacing u by v, which we extend naturally to sets as well as to tuples, e.g., for a set A of arcs, we put $A\frac{u}{v} = \{w\frac{u}{v}Lz\frac{u}{v} : wLz \in A\}$.

Given concrete graphs $G = \{S_i : i \in I\}$ and $H = \{T_j : j \in J\}$, we define

$G^{\mathsf{T}} = \{{S_i}^{\mathsf{T}} : i \in I\}$, the *transposition* of G,
$G \circ H = \{S_i \circ T_j : i \in I, j \in J\}$, the *composition* of G and H,
$G \sqcap H = \{S_i \sqcap T_j : i \in I, j \in J\}$, the *intersection* of G and H,
$G \sqcup H = G \cup H$, the *union* of G and H.

Slices $S_1 = (N_1, A_1, x_1, y_1)$ and $S_2 = (N_2, A_2, x_2, y_2)$ are *isomorphic* if there are bijections $f : N_1 \to N_2$ and $g : A_1 \to A_2$ such that

1. for all $urv \in A_1$, $g(urv) = furfv$,
2. for all $u\boxed{G}v \in A_1$, $g(u\boxed{G}v) = fu\boxed{G'}fv$ and G and G' are isomorphic,
3. $fx_1 = x_2$ and $fy_1 = y_2$.

Concrete graphs G and H are *isomorphic* if there is a bijection $h : G \to H$ such that $h(S)$ is isomorphic to S, for all $S \in G$. The usual identification of isomorphic concrete graphs is reflected in our figures by the representation of every non-distinguished node by $\bullet$, of every input node by $-$, and of every output node by $+$.

A *graph* is an equivalence class of isomorphic concrete graphs. In what follows, a graph is identified with each of the concrete graphs that represents the equivalence class.

A *graph inclusion* is an expression of the form $G \sqsubseteq H$.

We now move on to semantics. Given a base set M, the labels, slices, and graphs will denote binary relations on M. To define this in a proper way we need the notions of a model and an assignment of individual nodes.

A *model*, typically denoted by $\mathfrak{M}$, is a structure $\langle M, \{r^{\mathfrak{M}} : r \in \text{RSYM}\}\rangle$, where $M \neq \emptyset$ is the *universe* of $\mathfrak{M}$ and $r^{\mathfrak{M}} \subseteq M \times M$, for every $r \in$ RSYM. The *meaning* of a label L in a model $\mathfrak{M}$, denoted by $[\![L]\!]_{\mathfrak{M}}$, is defined by $[\![r]\!]_{\mathfrak{M}} = r^{\mathfrak{M}}$ and $\left[\!\left[\boxed{\boxed{\text{G}}}\right]\!\right]_{\mathfrak{M}} = [\![G]\!]^{\complement}_{\mathfrak{M}}$, the complement of $\left[\!\left[\boxed{\boxed{\text{G}}}\right]\!\right]_{\mathfrak{M}}$. The *meaning* of a graph G in a model $\mathfrak{M}$, denoted by $[\![G]\!]_{\mathfrak{M}}$, is defined in Table 1, where R^{-1} is the *transpose* of relation R, i.e. $R^{-1} = \{(a,b) \in M \times M : (b,a) \in R\}$, and $R_1 \mid R_2$ is the *composition* of relations R_1 and R_2, i.e $R_1 \mid R_2 = \{(a,b) \in M \times M : (a,c) \in R_1 \text{ and } (c,b) \in R_2, \text{ for some } c \in M\}$.

Table 1. Meaning of graphs

$[\![\{S_L\}]\!]_{\mathfrak{M}} = [\![L]\!]_{\mathfrak{M}}$	$[\![G^{\top}]\!]_{\mathfrak{M}} = [\![G]\!]_{\mathfrak{M}}{}^{-1}$
$[\![E]\!]_{\mathfrak{M}} = M \times M$	$[\![G \circ H]\!]_{\mathfrak{M}} = [\![G]\!]_{\mathfrak{M}} \mid [\![H]\!]$
$[\![O]\!]_{\mathfrak{M}} = \emptyset$	$[\![G \sqcap H]\!]_{\mathfrak{M}} = [\![G]\!]_{\mathfrak{M}} \cap [\![H]\!]$
$[\![I]\!]_{\mathfrak{M}} = \{(a,b) \in M \times M : a = b\}$	$[\![G \sqcup H]\!]_{\mathfrak{M}} = [\![G]\!]_{\mathfrak{M}} \cup [\![H]\!]$

As usual, we introduce a notion of consequence between a set of graph inclusions and a graph inclusion based on meaning. We say that a model $\mathfrak{M}$ *verifies* a graph inclusion $G \sqsubseteq H$, denoted by $\mathfrak{M} \models G \sqsubseteq H$, iff $[\![G]\!]_{\mathfrak{M}} \subseteq [\![H]\!]_{\mathfrak{M}}$. We say that a graph inclusion is *valid*, denoted by $\models G \sqsubseteq H$, iff it is verified by any model. We say that a model $\mathfrak{M}$ *verifies* a set Γ of graph inclusions, denoted by $\mathfrak{M} \models \Gamma$, iff $\mathfrak{M}$ verifies every graph inclusion in Γ. We say that a graph inclusion $G \sqsubseteq H$ is a *consequence* of a set Γ of graph inclusions, denoted by $\Gamma \models G \sqsubseteq H$, iff $\mathfrak{M} \models G \sqsubseteq H$ whenever $\mathfrak{M} \models \Gamma$, for every model $\mathfrak{M}$. As usual, we have $\models G \sqsubseteq H$ iff $\emptyset \models G \sqsubseteq H$.

Now we present a set of valid inclusions and a set of rules to transform a graph into another. These rules will preserve meaning when applied to graphs.

We first introduce two families of valid inclusions. These will play the role of axioms in our graph calculus. Recall $O = \emptyset$, the empty graph, and $E = \{(\{x,y\}, \emptyset, x, y)\}$, the graph consisting of one arcless slice with two distinct nodes, input x and output y. Given a slice $S = (N_S, A_S, x_S, y_S)$, we define two graphs as follows. The graph $O_S = \{(N_S, A_S \cup \{x_S \boxed{S}\, y_S\}, x_S, y_S)\}$ is obtained from S by adding to it a new arc from the input of S to the output of S labeled by $\boxed{S}$. The graph $E_S = \{S, (\{x, y\}, \{x \boxed{S}\, y\}, x, y)\}$ is obtained from S by adjoining to S a new slice with two distinct nodes, input x and output y, and a single arc $x \boxed{S}\, y$ (Figure 6).

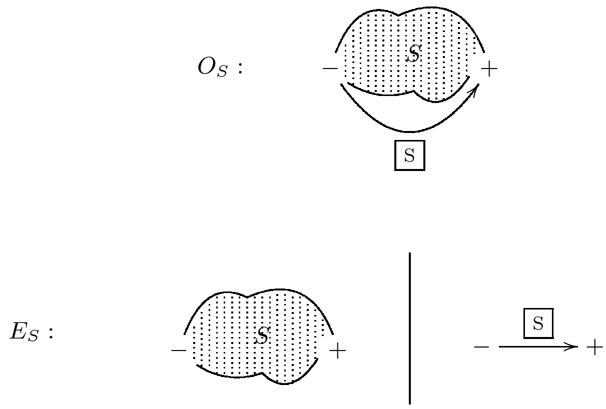

Fig. 6. Graphs O_S and E_S

We shall take as schemes of axioms of the graph calculus the inclusions $O_S \sqsubseteq O$ and $E \sqsubseteq E_S$ for any every S (Table 2). It follows immediately from the definitions that these inclusions are valid.

Table 2. Axioms

$O_S \sqsubseteq O$	and	$E \sqsubseteq E_S$

Lemma 1. $\mathfrak{M} \models \{O_S \sqsubseteq O, E \sqsubseteq E_S\}$*, for every model* $\mathfrak{M}$ *and every slice* S*.*

The rules of our graph calculus are Graph Cover rule, Hypothesis rule and Box rule. Graph Cover rule is used to compare graphs with respect to inclusion, Hypothesis rule, to transform graphs according to the set of inclusions taken as hypotheses, and Box rule, to simplify the inner structure of box labels.

To define our first transformation rule, the concepts of homomorphism from a slice to another and that of a graph covering another will be crucial. Given slices $S = (N_S, A_S, x_S, y_S)$ and $T = (N_T, A_T, x_T, y_T)$, by a *slice homomorphism from* T *to* S we mean a function $\phi : N_T \to N_S$, denoted by $\phi : T \to S$, that preserves input, output, and arcs, i.e. $\phi x_T = x_S$, $\phi y_T = y_S$, and if $uLv \in A_T$ then $\phi uL\phi v \in A_S$. Given graphs G and H, we say that H *covers* G or G *is covered by* H, denoted by $G \leftarrow H$, iff for each slice $S \in G$ there exist a slice $T \in H$ and a slice homomorphism $\phi : T \to S$.

Rule Cv (Table 3) allows us to replace a graph by another one that covers it. The next result, showing that covering preserves meaning, i.e. that rule Cv is sound, follows from the fact that a slice homomorphism transfers assignments by composition.

Table 3. Graph Cover rule

$$\mathsf{Cv}\ \frac{G}{H} \quad \text{if } \ G \leftarrow H$$

Lemma 2. *If* $G \leftarrow H$*, then* $[\![G]\!]_{\mathfrak{M}} \subseteq [\![H]\!]_{\mathfrak{M}}$*, for every model* $\mathfrak{M}$*.*

We now introduce the concepts of gluing slices and draft homomorphism between slices, which will be central in applying a graph inclusion to transform a graph into another.

Intuitively, we glue slice T onto slice S by adding to S a copy of T and identifying designated nodes u, v of S to the input and output of T. More precisely, given slices $S = (N_S, A_S, x_S, y_S)$ and $T = (N_T, A_T, x_T, y_T)$, as well as designated nodes $u, v \in N_S$, the result of *gluing* T onto S via u, v is the slice defined by $\mathsf{glue}_{(u,v)}(T, S) = (N_S \uplus N_T, A_S \uplus A_T, x_S, y_S)\frac{x_T}{u}\frac{y_T}{v}$. We *glue* a graph H onto a slice S, via nodes u, v of S, by gluing its slices to S, i.e. $\mathsf{glue}_{(u,v)}(H, S) = \{\mathsf{glue}_{(u,v)}(T, S) : T \in H\}$ (Figure 7).

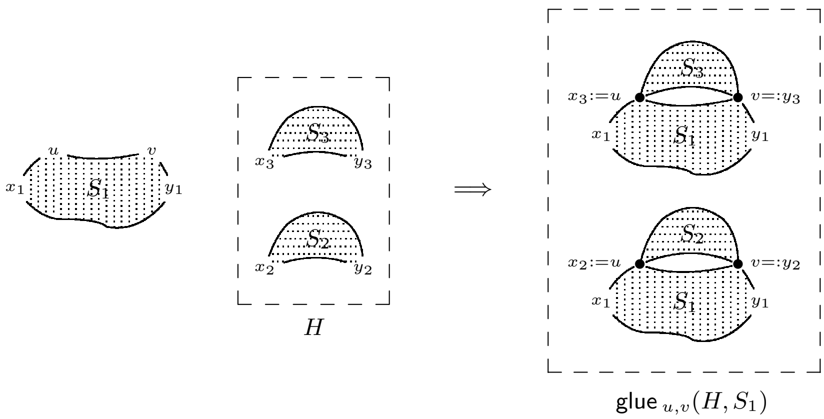

Fig. 7. Gluing H in S_1 via u, v

Given slices $S = (N, A, x, y)$ and $S' = (N', A', x', y')$, by a *draft homomorphism from S' to S* we mean a function $\theta : N' \to N$, denoted by $\theta : S' \dashrightarrow S$, that preserves arcs. Now, given slices S and S' as before, a draft homomorphism $\theta : S' \dashrightarrow S$, and graph H, we set $\mathsf{glue}_\theta(H, S) = \mathsf{glue}_{(\theta x', \theta y')}(H, S)$. Rule Hyp_Γ (Table 4) allows us to glue a graph H onto a slice S of a graph under a draft homomorphism $\theta : S' \dashrightarrow S$ when $G' \cup \{S'\} \sqsubseteq H$ is a hypothesis in Γ or is an axiom.

Table 4. Hypothesis rule

$$\mathsf{Hyp}_\Gamma \quad \frac{G \cup \{S\}}{G \cup \mathsf{glue}_\theta(H, S)}$$

if $\theta : S' \dashrightarrow S$ and $G' \cup \{S'\} \sqsubseteq H$ is in Γ or is an axiom

The next result, showing that gluing preserves meaning, i.e. that rule Hyp_Γ is sound, follows from the fact that draft homomorphisms transfer assignments by composition.

Lemma 3. *For all slices S and S' and draft homomorphism $\theta : S' \dashrightarrow S$, if $\mathfrak{M} \models \{S'\} \sqsubseteq H$, then $\mathfrak{M} \models \{S\} \sqsubseteq \mathsf{glue}_\theta(H, S)$, for every model $\mathfrak{M}$.*

The Box rule (Table 5) is a two-way rule, i.e. it can be applied in the top-down and in the bottom-up directions. In the top-down direction, the Box rule allows us to replace an arc labeled by a box $\boxed{H}$ having a graph $H = \{S_i : i \in I\}$ inside of it, by a set $\{u \boxed{S_i} v : i \in I\}$ of parallel arcs, each one labeled by a box with the unary slice graph S_i inside of it, and vice-versa, for the bottom-up direction. In the case $I = \emptyset$, the Box rule allows us to erase (top-down) or to add (bottom-up) an arc labeled by a box with the empty graph O inside of it. Our calculus is heavily based on slice homomorphism, but a graph is a finite set of slices. Thus the De Morgan's laws expressed by our Box rule, in the top-down direction, allows one to obtain graphs where boxes have singleton graphs, preparing the application of the Cover rule.

Soundness of the Box rule follows immediately from the definitions of meaning of box label and meaning of graphs.

Table 5. Box rule

$$\mathsf{Box} \ \frac{G \cup \{(N, A \cup \{u \boxed{\{S_i : i \in I\}} v\}, x, y)\}}{G \cup \{(N, A \cup \{u \boxed{S_i} v : i \in I\}, x, y)\}}$$

Lemma 4. *For every model $\mathfrak{M}$, it follows that*

$$[\![(N, A \cup \{u \boxed{\{S_i : i \in I\}} v\}, x, y)]\!]_{\mathfrak{M}} = [\![\{(N, A \cup \{u \boxed{S_i} v : i \in I\}, x, y)\}]\!]_{\mathfrak{M}}.$$

The notion of derivation is standard as in the rewriting systems for equational proofs. A proof of a graph inclusion $G \sqsubseteq H$ from a set of hypotheses is obtained starting with G and applying our derivation rules and the hypotheses for rewriting G until we obtain H. Given a set of graph inclusions Γ, by a *derivation from Γ*, or simply a *Γ-derivation*, we mean a sequence $(G_0, \dots, G_n)$ of graphs such that each graph G_i, for $i \in \{1, \dots, n\}$, is obtained from graph G_{i-1} by an application of one of the rules Cv, Hyp_Γ, or Box. A graph H is *derivable from a graph G using Γ*, or simply *H is Γ-derivable from G*, denoted by $\Gamma \vdash G \sqsubseteq H$, when there is a Γ-derivation $(G_0, \dots, G_n)$ such that $G_0 = G$ and $G_n = H$. An inclusion $G \sqsubseteq H$ is a *theorem*, denoted by $\vdash G \sqsubseteq H$, when H is derivable from G using the empty set of hypotheses.

The Box rule gives us a normal form for graphs.

Lemma 5. *For all graph G, there is a graph $\mathsf{NF}G$ such that $\vdash G \sqsubseteq \mathsf{NF}G$, $\vdash \mathsf{NF}G \sqsubseteq G$, and every box label occurring in $\mathsf{NF}G$ encloses a singleton graph.*

4 Completeness of the Graph Calculus with Complement

Soundness of our graph calculus follows from Lemmas 1, 2, 3, and 4. We will give the general idea of the completeness proof . We may assume that every box label have a unary graph inside, based on Lemma 5.

Given a set of graph inclusions Γ and a graph inclusion $G \sqsubseteq H$, follow the procedure.

> Step 0. $G_0 := G$.
> Step $i+1$. Either $G_i \leftarrow H$, then stop, or else there is a slice $S \in G_i$ such that $\{S\} \not\leftarrow H$. Write $G_i = G' \cup \{S\}$. Take $G_i := G' \cup \mathsf{glue}_S(\theta, H')$, where $\theta : S' \dashrightarrow S$ and $G' \cup \{S'\} \sqsubseteq H' \in \Gamma$ or is an axiom.

If the procedure ever stops, the sequence $(G_0, \dots, G_n)$ is a Γ-derivation of $G \sqsubseteq H$. Otherwise, we have a directed chain of slices $(S_n)_{n \in \mathbb{N}}$ such that, for all $i \in \mathbb{N}$, $\{S_i\}$ is not covered by H, and there is a slice homomorphism from S_i to S_{i+1}. In fact, for each $i \in \mathbb{N}$, there is a slice $S_i \in G_i$ with $\{S\} \not\leftarrow H$ and a slice $S_{i+1} \in G_{i+1}$ with $\{S_{i+1}\} \not\leftarrow H$, for $G_i = G' \cup \{S_i\}$, $G_{i+1} = G' \cup \mathsf{glue}_{S_i}(\theta, H')$ and $S_{i+1} \in \mathsf{glue}_{S_i}(\theta, H')$. Since $S_{i+1} \in \mathsf{glue}_{S_i}(\theta, H')$, there is a slice homomorphism $\phi : S_i \to S_{i+1}$.

The *canonical model* $\mathfrak{M}^c = \langle \tilde{N}_*, \{r^{\mathfrak{M}^c} : r \in \textsc{Rsym}\} \rangle$ is obtained by a direct limit on slice chain $(S_n)_{n \in \mathbb{N}}$. More explicitly, form the set N_* of all nodes occurring in the slices of the chain, and define the (equivalence) relation $\sim$ on it as follows: given nodes $u \in S_i$ and $v \in S_j$, set $u \sim v$ iff, for some $k \geq i, j$, $\phi^i_k u = \phi^j_k v$. Take $\tilde{N}_*$, the quotient set of N_* by $\sim$, with the natural quotient map $\nu : N_* \to \tilde{N}_*$.

Now, interpret each relation symbol naturally by setting $(\tilde{u}, \tilde{v}) \in r^{\mathfrak{M}^c}$ iff there exist $n \in \mathbb{N}$, $u' \sim u$, $v' \sim v$, and an arc $u'rv' \in A_{S_n}$.

We then have the Satisfiability Lemma, whose proof will be omitted, by the lack of space.

Lemma 6 (Satisfiability). *Consider the canonical model $\mathfrak{M}^c$ for the graph inclusion $G \sqsubseteq H$ with set of hypotheses Γ. Hence, (1) $\mathfrak{M}^c \models \Gamma$ and (2) $\mathfrak{M}^c \not\models G \sqsubseteq H$.*

We thus have completeness of our graph calculus.

Theorem 1 (Completeness). *If $\Gamma \models G \sqsubseteq H$, then $\Gamma \vdash G \sqsubseteq H$.*

Proof. Suppose $\Gamma \models G \sqsubseteq H$. Hence, there is no (counter-)model $\mathfrak{M}$ such that $\mathfrak{M} \models \Gamma$ and $\mathfrak{M} \not\models G \sqsubseteq H$. Hence, the procedure stops. Hence, $\Gamma \vdash G \sqsubseteq H$. ∎

From the considerations above, we have that whenever $\Gamma \models G \sqsubseteq H$, there is a derivation in a normal form, i.e. a derivation consisting of a sequence of applications of Box followed by a sequence of applications of Hyp_Γ followed by a single application of Cv followed by a sequence of applications of Box.

Corollary 1 (Normal form of derivations). *Given a set $\Gamma \cup \{G \sqsubseteq H\}$ of graph inclusions, if $\Gamma \models G \sqsubseteq H$, then there are graphs G_1, G_2, G_3 with all box label having a unary graph inside such that $G \xRightarrow{\mathsf{Box}^*} G_1 \xRightarrow{\mathsf{Hyp}_\Gamma^*} G_2 \leftarrow G_3 \xRightarrow{\mathsf{Box}^*} H$.*

Our calculus can be adapted to be used with graphs whose labels are more complex, i.e. we can assume our labels are terms generated from relational symbols by application of relational algebraic operators as the Boolean, Peircean and others. For instance, the operations complement, union, intersection, composition, and reversion, whose meaning are the expected ones, as well as constant terms interpreted in a model whose universe is M as $M \times M$, the identity and the empty relations. For this, one has to provide meaning preserving transformation rules to simplify the labels of the slice to boxes (with slices inside) and atomic relations.

Acknowledgment. We are indebted to the anonymous referees for valuable comments which enhanced the quality of the paper. In particular, we would like to thank the first referee for the very interesting remarks he/she has risen.

References

1. Andréka, H., Bredikhin, D.A.: The equational theory of union-free algebras of relations. Algebra Universalis 33, 516–532 (1995)
2. Brown, C., Hutton, G.: Categories, allegories and circuit design. LICS 94, 372–381 (1994)
3. Brown, C., Jeffrey, A.: Allegories of circuits. In: Proc. Logical Foundations of Computer Science, pp. 56–68. St. Petersburg (1994)

4. Bredikhin, D.A.: On clones generated by primitive-positive operations of Tarskis relation algebras. Algebra Universalis 38, 165–174 (1997)
5. Cantone, D., Formisano, A., Omodeo, E.G., Zarba, C.G.: Compiling dyadic specifications into map algebra. Theoret. Comput. Sci. 293, 447–475 (2003)
6. Curtis, S., Lowe, G.: Proofs with graphs. Sci. Comput. Programming 26, 197–216 (1996)
7. Formisano, A., Omodeo, E.G., Simeoni, M.: A graphical approach to relational reasoning. ENTCS 44, 1–22 (2003)
8. Freyd, P.J., Scedrov, A.: Categories, Allegories. Elsevier, Amsterdam (1990)
9. de Freitas, R., Viana, P.: A note on proofs with graphs. Sci. Comput. Programming 73, 129–135 (2008)
10. de Freitas, R., Veloso, P.A.S., Veloso, S.R.M., Viana, P.: On positive relational calculi. Logic J. IGPL 15, 577–601 (2007)
11. de Freitas, R., Veloso, P.A.S., Veloso, S.R.M., Viana, P.: On a graph calculus for algebras of relations. In: Hodges, W., de Queiroz, R. (eds.) WoLLIC 2008. LNCS (LNAI), vol. 5110, pp. 298–312. Springer, Heidelberg (2008)
12. de Freitas, R., Veloso, P.A.S., Veloso, S.R.M., Viana, P.: Positive fork graph calculus. In: Artemov, S., Nerode, A. (eds.) LFCS 2009. LNCS, vol. 5407, pp. 152–163. Springer, Heidelberg (2008)
13. de Freitas, R., Veloso, P.A.S., Veloso, S.R.M., Viana, P.: On graph reasoning. Information and Computation 207, 1000–1014 (2009)
14. Grätzer, G.: Universal Algebra. Springer, New York (1979)
15. Hirsch, R., Hodkinson, I.: Relation Algebras by Games. Elsevier, Amsterdam (2002)
16. Hutton, G.: A relational derivation of a functional program. In: Lecture Notes of the STOP Summer School on Constructive Algorithms (1992)
17. Kahl, W.: Algebraic graph derivations for graphical calculi. In: D'Amore, F., Marchetti-Spaccamela, A., Franciosa, P.G. (eds.) WG 1996. LNCS, vol. 1197, pp. 224–238. Springer, Heidelberg (1997)
18. Kahl, W.: Relational treatment of term graphs with bound variables. Logic J. IGPL 6, 259–303 (1998)
19. Kahl, W.: Relational matching for graphical calculi of relations. Inform. Sciences 119, 253–273 (1999)
20. Maddux, R.: Relation Algebras. Elsevier, Amsterdam (2006)