Parametric Sheaves for modelling Store Locality

Hongseok Yang
Department of Computer Science
University of Illinois at Urbana-Champaign
hyang@cs.uiuc.edu

Uday S. Reddy
School of Computer Science
University of Birmingham
U.Reddy@cs.bham.ac.uk

July 14, 2000

Abstract

In this paper, we bring together two important ideas in the semantics of Algol-like imperative programming languages. One is that program phrases act on fixed sets of storage locations. The second is that the information of local variables is hidden from client programs. This involves combining sheaf theory and parametricity to produce new classes of sheaves. We define the semantics of an Algol-like language using such sheaves and discuss the reasoning principles validated by the semantics.

1 Introduction

A programmer working in an imperative programming language understands intuitively that every program phrase acts on a fixed collection of storage locations and, it will continue to act on the same set of locations even if invoked in a larger store with extra locations. Capturing this form of locality in a semantic model is the subject of this paper.

As an application of locality idea, consider a predicate of "independence." For program phrases M and N, $M \perp N$ is to mean that M and N act on disjoint sets of locations. The idea then is that one can intermix the computations of M and N freely, and there would be no unwanted interference. Using our intuitive understanding of the independence predicate, we can formulate a "decomposition" axiom:

$$M \perp N_1 \;\wedge\; M \perp N_2 \Longrightarrow M \perp (N_1, N_2)$$

In words, if locations accessed by M are disjoint from those accessed by N_1 and those accessed by N_2, then they should be disjoint from the locations accessed by the pair (N_1, N_2). Unfortunately, most semantic models found in the literature do not validate this reasoning principle. Hence, they fail to capture storage locality.

The independence predicate was first postulated by Reynolds [26] in formulating his system of Syntactic Control of Interference. The semantics of this system was studied by O'Hearn [13, 14], who successfully modelled the independence predicate by using pullback-preserving functors on location worlds. A similar model was used by Pitts and Stark [24], who also noted that the pullback-preserving functors form sheaves in the Shanuel topos. The support-based model of [12] is also related. Since most other models of Algol-like languages use functors [27, 21, 32, 18], this would seem like a small further step. But the categories of worlds used in these other models do not have pullbacks. It is still an open question whether these models have an interpretation for the independence predicate (with the decomposition axiom).

The next major step in the semantics of Algol-like languages is the realization that the "information hiding" features implicit in local variables can be modelled using relational parametricity [19, 16]. In Reddy [25], these ideas were applied to semantics of object-oriented programs. Reddy also formulated a programming logic using the independence predicate mentioned above, but the effort to give a semantics foundered because the existing tools were not adequate to model the combination of parametricity and the independence predicate.

In this paper, we formulate a semantic model that combines parametricity and the independence predicate by extending sheaf theory for reflexive graphs of categories. (The latter form the framework for modelling parametricity.) Sheaves, familiar from topology and geometry [11], capture the kind of locality property involved in imperative computations over locations. In fact, the original models of [14, 24] are sheaf models. The challenge for us is to extend theory of sheaves to cover parametricity aspects.

We believe that the ideal setting for modelling parametricity is to move from categories to reflexive graphs of categories [19]. Having made such a move, we face the question: what is a presheaf in the world of reflexive graphs. A standard answer might be reflexive graph-functors of type $\mathbf{C}^{\mathrm{op}} \to \mathbf{Rel}$, where $\mathbf{Rel}$ is the reflexive graph structure on $\mathbf{Set}$ with binary relations as edges and relation-preserving pairs of functions as edge morphisms. Certainly, this reflexive graph is pervasive in parametricity theory [29, 19], and seems to play a role similar to that of $\mathbf{Set}$ in category theory. However, the category of such functors, $\mathbf{Rel}^{\mathbf{C}^{\mathrm{op}}}$, does not form a topos. So, we work in $\mathbf{Span}^{\mathbf{C}^{\mathrm{op}}}$, where $\mathbf{Span}$ is the reflexive graph with spans in $\mathbf{Set}$ as edges. Since $\mathbf{Rel}$ embeds into $\mathbf{Span}$, this seems to be a good choice. But, $\mathbf{Span}$ is cumbersome to work with and our definitions end up with excessive detail and lack conviction. We close the circle by showing that our definitions match up with the standard definitions of sheaves internal to the topos $\mathbf{Span}^{\mathbf{C}^{\mathrm{op}}}$.

Related Work

After Reynolds' series of seminal papers on higher-order imperative languages such as Idealized Algol (in short, IA) and Syntactic Control of Interference (in short, SCI) [27, 26, 28], there have been several works to define the semantics of IA and SCI based on the presheaf category [21, 13, 14, 32, 18, 15, 12]. In

[19, 16], the relational parametricity has been used to obtain more abstract model which captures data abstractions and the model is extended in [25] to handle classes and objects. Atomic sheaves on the category of finite sets and injections (equivalently, pullback-preserving functors) are used to give a model of SCI in [14] and that of nu-calculus in [31, 24]. Such atomic sheaves also become the theoretical basis of the approach for handling binders in abstract syntax in [4].

The parametricity theory itself arose from an effort to get a model of polymorphic lambda calculus and to formalize logical relations in a categorical setting [29, 10, 30]. A structure known as reflexive graph has been used in the theory.

The category of sheaves has been used to give a semantics of a type theory or a first order logic in [9, 11, 22]. Goguen proposed a framework based on sheaves for defining semantics of concurrent object oriented languages [5]. Fiore and Simpson defined Grothendieck logical relations using ideas from sheaf theory to show lambda definability results [3]. In [6], Hilken extended the meaning of sheaves in order to obtain a model for intuitionistic modal logic. He defined "relational" sheaves for a "relational" topological space, which are essentially sheaves with respect to somewhat unusual Grothendieck topology on the category of open sets with one more condition: they are separated with respect to the usual Grothendieck topology. His definition as well as his motivation is very different from our work: neither subsumes the other and it is not clear that either of the definitions can be used to solve the main motivating problem in the other's approach.

2 Preliminaries

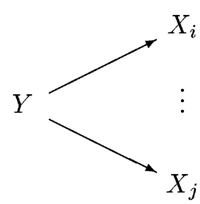
2.1 Sheaves

A *presheaf* is a **Set**-valued functor. Intuitively, a presheaf denotes sets parameterized over a category. For example, interpretations of Algol types, like *COMM* and *EXP*, denote presheaves that are parameterized by "store shapes." $COMM(X)$ denotes commands for modifying stores of shape X. $EXP(X)$ denotes expressions that read stores of shape X.

Sheaves are again sets parameterized by a category but satisfying a property that is "local" in an abstract sense. A standard motivating example is the property of topological continuity [11]. Intuitively, continuity is a local property in that a function on a space O is continuous just if it is continuous on each O_i belonging to one family of open sets $\{O_i\}_{i \in I}$ that cover O. Let $\mathbf{S}$ be the category of all subspaces of a space S, with inclusions as morphisms. Consider the set $CONT(O)$ of all continuous real-valued functions on O. We make two observations: First, whenever $O' \subseteq O$ is a subspace of O, we have a restriction operation $CONT(O) \to CONT(O')$. This just says that $CONT\colon \mathbf{S}^{\mathsf{op}} \to \mathbf{Set}$ is a presheaf. Second, if $\{O_i\}_{i \in I}$ is an open cover of O, and $f_i \in CONT(O_i)$ is a family of continuous functions that agree on all common elements of the O_i, then there is a unique $f \in CONT(O)$ such that each f_i is a restriction of f. This says that $CONT$ is a sheaf. However, note that the idea of sheaf depends on the notion of "open covers" which is not a part of the specification of the category $\mathbf{S}$. To model the covers, we assume that we are given a function J that assigns to each space O a set of covers of O (i.e., families of subspaces of O satisfying certain axioms). The pair $(\mathbf{S}, J)$ is called a *site* and $CONT$ is a sheaf over this site.

Coming to Algol semantics, regard store shapes as finite sets of locations. Form a category $\mathbf{W}$ of these store shapes with *reverse* inclusions as morphisms. We can treat parameterized sets $COMM(X)$ in the same fashion as $CONT(O)$ above. First, whenever there is an inclusion $X \hookrightarrow X'$ (so that X' is thought of as a "subspace" of X), there is an operation $COMM(X) \to COMM(X')$ that restates a command on X as one on X'. (It is better to think of this operation as an "expansion" rather than a "restriction."). Second, if $\{X_i\}_{i \in I}$ is a family of store shapes such that $\bigcap_{i \in I} X_i = X$, and $c_i \in COMM(X_i)$ is a family of commands that is consistent in an appropriate sense (see below), then there is a unique command $c \in COMM(X)$ whose expansions are the various c_i. This amounts to saying that $COMM$ is a sheaf on a site $(\mathbf{W}, J)$.

The notion of consistent family involved in both $CONT$ and $COMM$ can be characterized in a categorical fashion. To say that a family of elements $\{x_i \in X_i\}_{i \in I}$ is consistent is to say that for any cone of the form



the elements x_i of the X_i's involved in the cone restrict/expand to the same element in Y. Further, we can obtain economy of expression by including all such Y's in the cover $\{X_i\}_{i \in I}$ to start with. Then, consistency would just amount to preservation under restriction/expansion operations.

Formally, a *sieve* on an object K is a collections of morphisms S leading to K that is right-closed under composition:

$$\forall f \colon X \to K \in S, \forall g \colon Y \to X.\ f \circ g \in S$$

A *site* is a pair $(\mathbf{C}, J)$ where $\mathbf{C}$ is a category and J assigns to each object X of $\mathbf{C}$, a collection of sieves on X (which are then called *covers* of X) satisfying the conditions of a Grothendieck topology [11]. If $F \colon \mathbf{C}^{\mathsf{op}} \to \mathbf{Set}$ is a presheaf and S is a sieve on an object X of $\mathbf{C}$, a *matching family* over S for F is a family of elements $\{x_f \in F(\mathsf{dom}(f))\}_{f \in S}$ such that $x_{f \circ g} = F(g)(x_f)$. Given such a family, an element $x \in F(X)$ is called its *amalgamation* if $x_f = F(f)(x)$ for each $f \in S$. A *sheaf* over the site $(\mathbf{C}, J)$ is a presheaf such that every matching family for a covering sieve has a unique amalgamation. The category of sheaves over $(\mathbf{C}, J)$ with natural transformations as morphisms is denoted $\mathbf{Sh}(\mathbf{C}, J)$. A sheaf category $\mathbf{Sh}(\mathbf{C}, J)$ is always cartesian closed. So we can interpret typed lambda calculus-based programming languages in it. It also forms a topos, which allows us to model programming logics as well.

The thesis is that all Algol types are local in this abstract sense and, hence, they are sheaves for an appropriate Grothendieck topology on the category of store shapes.

2.2 Parametricity and Reflexive Graphs

Relational parametricity, as formulated by Reynolds [29] for the set-theoretic case, involves statements of the form:

$$\forall x_0, x_1.\, x_0 \, R \, x_1 \Longrightarrow f_0(x_0) \, S \, f_1(x_1)$$

where $R\colon A_0 \leftrightarrow A_1$ and $S\colon B_0 \leftrightarrow B_1$ are binary relations and $f_0\colon A_0 \to B_0$ and $f_1\colon A_1 \to B_1$ are functions. Intuitively, the idea is that f_0 and f_1 map R-related values to S-related values. We use the diagrammatic notation

$$\begin{array}{ccc} A_0 & \xrightarrow{f_0} & B_0 \\ R \updownarrow & & \updownarrow S \\ A_1 & \xrightarrow{f_1} & B_1 \end{array} \qquad (1)$$

to denote such a relationship between f_0 and f_1. These kinds of statements play the same role as commutative squares in category theory and form the basis of the uniformity concept captured by relational parametricity.

O'Hearn and Tennent [19] used reflexive graphs of categories to lift this picture to an abstract level, so that it could be applied to the semantics of Idealized Algol. (See also [8, 23, 30] for other applications.) Looking at the diagram (1) above, we notice two kinds of objects and morphisms:

vertices and vertex morphisms: A_i, B_i and $A_i \xrightarrow{f_i} B_i$

edges and edge morphisms: R, S and $\begin{array}{ccc} A_0 & \xrightarrow{f_0} & B_0 \\ R \updownarrow & & \updownarrow S \\ A_1 & \xrightarrow{f_1} & B_1 \end{array}$

These can be organized into two categories $\mathbf{C}_v$ and $\mathbf{C}_e$ together with two functors

$$\partial_0, \partial_1 \colon \mathbf{C}_e \to \mathbf{C}_v$$

that pick out the source and target of edges/edge morphisms in the vertical direction. In addition to these, we require a functor

$$\mathbf{I}\colon \mathbf{C}_v \to \mathbf{C}_e$$

which assigns to each vertex V a canonical edge $\mathbf{I}_V$ (the "identity edge") and similarly for each vertex morphism. The identity edges are used to capture the "identity extension" property of type constructors. All said and done, we have the following definition:

Definition

1. A *reflexive graph* of categories is a quintuple $\mathbf{C} = (\mathbf{C}_v, \mathbf{C}_e, \partial_0, \partial_1, \mathbf{I})$ where $\mathbf{C}_v$ and $\mathbf{C}_e$ are categories and the remaining components are functors such that:

$$\mathbf{C}_v \xrightarrow{\mathbf{I}} \mathbf{C}_e \xrightarrow{\partial_i} \mathbf{C}_v = \mathsf{Id}_{\mathbf{C}_v}$$

2. A *reflexive graph-functor* (RG-functor) $F\colon \mathbf{C} \to \mathbf{D}$ is a pair $F = (F_v\colon \mathbf{C}_v \to \mathbf{D}_v,\ F_e\colon \mathbf{C}_e \to \mathbf{D}_e)$ preserving the structure of reflexive graphs: $\partial_i \circ F_e = F_v \circ \partial_i$ and $\mathbf{I} \circ F_v = F_e \circ \mathbf{I}$ (The second condition is referred to as the *identity extension* property.)

3. Given two RG-functors $F, G\colon \mathbf{C} \to \mathbf{D}$, a *parametric natural transformation* $\eta\colon F \to G$ is a pair $\eta = (\eta_v\colon F_v \to G_v, \eta_e\colon F_e \to G_e)$ preserving the structure of reflexive graphs:

$$\partial_i \eta_e = \eta_v \partial_i \text{ and } \mathbf{I}\eta_v = \eta_e \mathbf{I}$$

When the collection of parametric natural transformations $F \to G$ is a set, it is denoted by $\mathsf{Par}[F, G]$.

(We omit the subscripts $_v$ and $_e$ whenever the context makes it clear which is meant.)

As remarked in [19], these definitions are nothing but those of indexed categories, indexed functors and indexed natural transformations for a particular situation. The term "parametric" in the part 3 of the definition signifies our notion of parametricity at the abstract level.

The canonical example of reflexive graphs, denoted **Rel**, captures Reynolds' parametricity theory [29]. The vertex category $\mathbf{Rel}_v$ is **Set**. The edge category $\mathbf{Rel}_e$ has binary relations $R\colon A \leftrightarrow B$ as objects and relation-preserving pairs of functions $(f_0, f_1) : R \to S$ as morphisms. The functors ∂_0 and ∂_1 are the evident ones. The functor $\mathbf{I}$ sends a set A to the diagonal relation on A and a function $f\colon A \to B$ to $(f, f)\colon \mathbf{I}_A \to \mathbf{I}_B$.

Note that products $\mathbf{C}_1 \times \mathbf{C}_2$ and duals $\mathbf{C}^{\mathsf{op}}$ of reflexive graphs are easily defined pointwise (separately for the vertex and edge parts).

Examples of RG-functors include the product and exponent functors:

$$\times : \mathbf{Rel} \times \mathbf{Rel} \to \mathbf{Rel} \quad \text{and} \quad \Rightarrow : \mathbf{Rel}^{\mathsf{op}} \times \mathbf{Rel} \to \mathbf{Rel}.$$

The edge parts of these RG-functors are the usual "logical relation" constructions $R \times S$ and $R \Rightarrow S$:

$$\begin{aligned} (x_0, y_0)\,(R \times S)\,(x_1, y_1) &\iff x_0\, R\, x_1 \wedge y_0\, S\, y_1. \\ f_0\,[R \Rightarrow S]\,f_1 &\iff \forall x_0, x_1.\, x_0\, R\, x_1 \Rightarrow f_0(x_0)\, S\, f_1(x_1). \end{aligned}$$

Examples of parametric natural transformations include

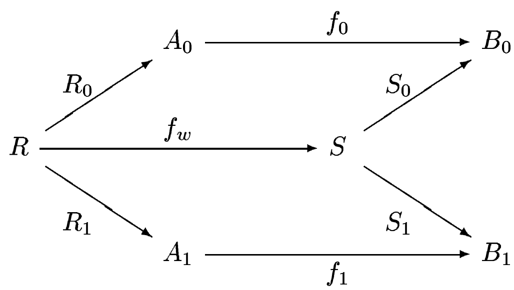
$$\begin{array}{rcl} \mathsf{fst}_{A,B} & : & A \times B \to A \\ \epsilon_{A,B} & : & (A \Rightarrow B) \times A \to B \end{array}$$

whose vertex parts are standard. For the edge parts, we have

$$\begin{array}{rcl} \mathsf{fst}_{R,S} & = & (\mathsf{fst}_{A_0,B_0}, \mathsf{fst}_{A_1,B_1}) \\ \epsilon_{R,S} & = & (\epsilon_{A_0,B_0}, \epsilon_{A_1,B_1}) \end{array}$$

The *existence* of these edge parts captures the parametricity condition of Reynolds.

A second example of reflexive graphs, which will play a significant role in this paper, is denoted **Span**. The vertex category is again **Set**. The edge category has spans in **Set** as objects. The edge morphisms are span-morphisms, i.e., triples of functions (f_0, f_w, f_1) making the following diagram commute:



The functors ∂_0 and ∂_1 send a span to its two tips respectively, and a span morphisms (f_0, f_w, f_1) to f_0 and f_1 respectively. The functor **I** sends a set A to the "identity span":

$$A \xleftarrow{\mathsf{id}_A} A \xrightarrow{\mathsf{id}_A} A$$

and a function f to (f, f, f). Spans are rather similar to relations, with the main difference being that we also get to encode "witnesses" for relating elements. Note that every relation can be regarded as a span $A_0 \leftarrow R \rightarrow A_1$ by using projections of $R \subseteq A_0 \times A_1$ as arrows. Thus, **Span** may be thought of as a "constructive" variant of **Rel**. There is a functor

$$\mathsf{wit} \colon \mathbf{Span}_e \to \mathbf{Set}$$

that sends a span $A_0 \overset{R_0}{\leftarrow} R \overset{R_1}{\rightarrow} A_1$ to the witness set R and a span morphism (f_0, f_w, f_1) to f_w. We often use the notation

$$\begin{array}{rl} F_w = \mathsf{wit} \circ F_e & \text{(for RG-functor } F \colon \mathbf{C} \to \mathbf{Span}) \\ \eta_w = \mathsf{wit}\, \eta_e & \text{(for parametric natural transformation } \eta \colon F \to G) \end{array}$$

for the functors and natural transformations that pick out the witness components of spans.

Both **Rel** and **Span** are cartesian closed reflexive graphs (in an appropriate sense). An important difference between them is that **Rel** is a "relational" reflexive graph, i.e., given edges R and S, for any pair of vertex morphisms (f_0, f_1) there is at most one edge morphism $\varphi\colon R \to S$ such that $\partial_0(\varphi) = f_0$ and $\partial_1(\varphi) = f_1$. This means that the edge morphisms $R \to S$ define a relation between vertex hom-sets. On the other hand, in **Span**, edge morphisms (f_0, f_w, f_1) have additional "witness" information in f_w. So, **Span** is not relational.

Other structures used for modelling parametricity, such as the categories $REL(K, B, F)$ of Ma and Reynolds [10], and the category of pers with saturated relations of Bainbridge at al. [1] can also be described as reflexive graphs.

The following property is satisfied by **Rel**, **Span** and most reflexive graphs of interest:

Definition A reflexive graph **C** is said to satisfy the *identity condition* if the functor **I** is full.[8]

Note that the functor **I** is already faithful for any reflexive graph. So, for a reflexive graph with identity condition, we have the isomorphism $\mathsf{Hom}_{\mathbf{C}_e}[\mathbf{I}_A, \mathbf{I}_B] \cong \mathsf{Hom}_{\mathbf{C}_v}[A, B]$.

3 RG-Presheaves

In the world of categories a presheaf is taken to be a functor of type $\mathbf{C}^{\mathsf{op}} \to \mathbf{Set}$. The category of presheaves $\mathbf{Set}^{\mathbf{C}^{\mathsf{op}}}$ forms a topos. Sheaves are then defined to be presheaves with special properties.

To duplicate these constructions in the world of reflexive graphs, we first need a canonical reflexive graph that plays the same role as **Set** does in the case of categories. One immediate candidate for the canonical reflexive graph is **Rel**, whose edges are binary relations between sets. Unfortunately, $\mathbf{Rel}^{\mathbf{G}^{\mathsf{op}}}$ is not in general a topos. So, this immediate choice has a shortcoming. In particular, this means that the Lawvere-Tierney notion of sheaves is not available to $\mathbf{Rel}^{\mathbf{G}^{\mathsf{op}}}$. A second candidate for the canonical reflexive graph is **Span**, whose edges are spans over **Set**. It turns out that $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ is always a topos and, therefore, serves our purpose. In this section, we examine the topos structure of this category.

Definition Given a reflexive graph with identity condition **G**, an *RG-presheaf* over **G** is an RG-functor $\mathbf{G}^{\mathsf{op}} \to \mathbf{Span}$. The category of RG-presheaves with parametric natural transformations as morphisms is denoted $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$.

From now on, we assume that the reflexive graph **G** is *small*, i.e., both $\mathbf{G}_v$ and $\mathbf{G}_e$ are small categories. If **G** is a small reflexive graph satisfying the identity condition, there is a Yoneda embedding of $\mathbf{G}_v$ in $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$, via a "hom-functor."

Define the RG-bifunctor:

$$\mathsf{Hom}\colon \mathbf{G}^{\mathsf{op}} \times \mathbf{G} \to \mathbf{Span}$$

as follows. The vertex part is, expectedly,

$$\mathsf{Hom}_v[U, V] = \mathsf{hom}_{\mathbf{G}_v}[U, V].$$

For the edge part, let $P\colon U_0 \leftrightarrow U_1$ and $R\colon V_0 \leftrightarrow V_1$ be edges of $\mathbf{G}$. Then $\mathsf{Hom}_e[P, R]$ is the span

$$\mathsf{Hom}_v[U_0, V_0] \xleftarrow{\partial_0} \mathsf{hom}_{\mathbf{G}_e}[P, R] \xrightarrow{\partial_1} \mathsf{Hom}_v[U_1, V_1]$$

If $\varphi\colon P' \to P$ and $\psi\colon R \to R'$ are edge morphisms with $\partial_0(\varphi) = f_0$, $\partial_1(\varphi) = f_1$, $\partial_0(\psi) = g_0$, $\partial_1(\psi) = g_1$, then $\mathsf{Hom}_e[\varphi, \psi]$ is the span-morphism

$$\begin{array}{ccccc}
 & \mathsf{Hom}_v[U_0, V_0] & \xrightarrow{\mathsf{Hom}_v[f_0, g_0]} & \mathsf{Hom}_v[U_0', V_0'] & \\
\nearrow^{\partial_0} & & & & \nearrow^{\partial_0} \\
\mathsf{hom}_{\mathbf{G}_e}[P, R] & & \xrightarrow{\mathsf{hom}_{\mathbf{G}_e}[\varphi, \psi]} & & \mathsf{hom}_{\mathbf{G}_e}[P', R'] \\
\searrow_{\partial_1} & & & & \searrow_{\partial_1} \\
 & \mathsf{Hom}_v[U_1, V_1] & \xrightarrow[\mathsf{Hom}_v[f_1, g_1]]{} & \mathsf{Hom}_v[U_1', V_1'] &
\end{array}$$

It is apparent that Hom preserves ∂_0 and ∂_1: $\mathsf{Hom}_v \circ (\partial_i \times \partial_i) = \partial_i \circ \mathsf{Hom}_e$. The identity extension property $\mathsf{Hom}_e \circ (\mathbf{I} \times \mathbf{I}) = \mathbf{I} \circ \mathsf{Hom}_v$ follows from the identity condition on $\mathbf{G}$.[1]

For a fixed vertex V of $\mathbf{G}$, we can specialize the hom-functor to an RG-presheaf $\mathsf{Hom}[-, V]\colon \mathbf{G}^{\mathsf{op}} \to \mathbf{Span}$ defined by

$$\mathsf{Hom}_v[-, V] = \mathsf{hom}_{\mathbf{G}_v}[-, V] \text{ and } \mathsf{Hom}_e[-, V] = \mathsf{Hom}_e[-, \mathbf{I}_V]$$

For any vertex morphism $f\colon V \to V'$ in $\mathbf{G}$, we have a parametric natural transformation $\mathsf{Hom}[-, f]\colon \mathsf{Hom}[-, V] \to \mathsf{Hom}[-, V']$ defined by:

$$\mathsf{Hom}_v[U, f] = f \circ - \text{ and } \mathsf{Hom}_e[P, f] = \mathbf{I}_f \circ -$$

This gives a Yoneda embedding of $\mathbf{G}_v$ in $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ which sends a vertex V to $\mathsf{Hom}[-, V]$ and a vertex morphism f to $\mathsf{Hom}[-, f]$.

There is also a functor $\mathcal{H}$ from the edge category $\mathbf{G}_e$ to $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$, for which a Yoneda lemma for edges holds, i.e., given an RG-functor F in $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$, $\mathsf{Par}[\mathcal{H}(-), F]$ is isomorphic to F_w in the presheaf category $\mathbf{Set}^{\mathbf{G}_e^{\mathsf{op}}}$. But this is nontrivial and does not follow the general categorical pattern used above.

[1] Strictly speaking, $\mathsf{Hom}_e \circ (\mathbf{I} \times \mathbf{I}) \cong \mathbf{I} \circ \mathsf{Hom}_v$ but not $\mathsf{Hom}_e \circ (\mathbf{I} \times \mathbf{I}) = \mathbf{I} \circ \mathsf{Hom}_v$. The equality can be obtained by modifying Hom using isomorphisms between $\mathsf{Hom}_e[\mathbf{I}_V, \mathbf{I}_V]$ and $\mathbf{I}_{\mathsf{Hom}_v(V)}$ for all vertices V. Whenever we present RG-functors to $\mathbf{Span}$, we leave such an appropriate modification of the RG-functors to the reader.

Since we will not explicitly need this functor for the results of this paper other than the proof of externalization in Section 4.1, we relegate it to the Appendix A and refer the interested reader to the discussion there. However we will need the following functor in the ensuing discussion, which is obtained by fixing the second argument of the Hom_e bifunctor at R:

$$\mathsf{Hom}_e[-, R]\colon \mathbf{G}_e^{\mathsf{op}} \to \mathbf{Span}_e$$

Lemma 1 If $\mathbf{G}$ is a small reflexive graph with identity condition, the corresponding RG-presheaf category $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ is cartesian closed.

The details are as in [19] but generalized to nonrelational reflexive graphs. Products and a terminal object are given pointwise:

$$(F \times G)(-) = F(-) \times G(-) \text{ and } 1(-) = 1$$

(for spans and span-morphisms, the products $F(R) \times G(R)$ and $F(\varphi) \times G(\varphi)$ are given pointwise.) Exponents are as follows:

- $(F \Rightarrow G)(V)$ is the set of *parametric* natural transformations $\mathsf{Par}[\mathsf{Hom}[-, V] \times F, G]$.
- $(F \Rightarrow G)(R)$, for an edge $R\colon V_0 \leftrightarrow V_1$, is a span whose witness set consists of all triples (η_0, ρ, η_1) where $\eta_i \in (F \Rightarrow G)(V_i)$ and $\rho\colon \mathsf{Hom}_e[-, R] \times F_e \to G_e$ such that
  1. $\partial_i \rho = (\eta_i)_v \partial_i$, and
  2. for every edge morphism of the form $\varphi : \mathbf{I}_U \to R$ (for some vertex U) and composable edge morphism $\psi\colon P \to \mathbf{I}_U$,

  $$\rho_P(\varphi \circ \psi, -) = \big((\eta_0)_e\big)_P(I_{\partial_0(\varphi)} \circ \psi, -) = \big((\eta_1)_e\big)_P(I_{\partial_1(\varphi)} \circ \psi, -)$$

  The arrows of the span send (η_0, ρ, η_1) to η_0 and η_1 respectively.
- $(F \Rightarrow G)(f)$, for a vertex morphism f, is $\mathsf{Par}[\mathsf{Hom}[-, f] \times F, G]$, which sends η to $\eta \circ (\mathsf{Hom}[-, f] \times \mathsf{id}_F)$,
- $(F \Rightarrow G)(\varphi)$, for an edge morphism $\varphi\colon R' \to R$ with $\partial_0(\varphi) = f_0$ and $\partial_1(\varphi) = f_1$, sends (η_0, ρ, η_1) to

  $$((F \Rightarrow G)(f_0)(\eta_0),\ \rho \circ (\mathsf{Hom}_e[-, \varphi] \times \mathsf{id}_{F_e}),\ (F \Rightarrow G)(f_1)(\eta_1))$$

The second condition in the definition of $(F \Rightarrow G)(R)$ is somewhat surprising. No such condition is involved for the exponents in $\mathbf{Rel}^{\mathbf{G}^{\mathsf{op}}}$ studied in [19]. More accurately, the condition is automatically satisfied in $\mathbf{Rel}^{\mathbf{G}^{\mathsf{op}}}$ because relations have unique witnesses for related pairs of elements. On the other hand, spans may have multiple witnesses in general. The condition is needed to ensure that $F \Rightarrow G$ has the identity extension property.

Theorem 2 If G is a small reflexive graph with identity condition, the corresponding RG-presheaf category $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ is a topos.

Since we have already argued that $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ is a CCC, all that is left is to define the subobject classifier $\Omega\colon \mathbf{G}^{\mathsf{op}} \to \mathbf{Span}$ (which must be an RG-functor). We define it using two kinds of sieves, called vertex-sieves and edge-sieves respectively.

Definition If $\mathbf{G}$ is a reflexive graph with identity condition and V is a vertex in $\mathbf{G}$, a *vertex-sieve* on V is a pair $S = (S_v, S_w)$ where

- S_v is a sieve on V in $\mathbf{G}_v$ and
- S_w is a sieve on $\mathbf{I}_V$ in $\mathbf{G}_e$

such that they are closed under the structure of the reflexive graph:

$$\forall \varphi \in S_w.\, \partial_0(\varphi), \partial_1(\varphi) \in S_v \quad \text{and} \quad \forall f \in S_v.\, \mathbf{I}_f \in S_w$$

A vertex-sieve can be regarded as an RG-subfunctor $S \hookrightarrow \mathsf{Hom}[-, V]$. Then $S(U) = \{f \in S_v \mid f\colon U \to V\}$ and $S(R\colon V_0 \leftrightarrow V_1) = S(V_0) \overset{\partial_0}{\leftarrow} \{\varphi \in S_w \mid \varphi\colon R \to \mathbf{I}_V\} \overset{\partial_1}{\to} S(V_1)$.

Definition If $\mathbf{G}$ is a reflexive graph with identity condition and $R\colon V_0 \leftrightarrow V_1$ is an edge in $\mathbf{G}$, an *edge-sieve* on R is a triple $\Sigma = (S_0, \Sigma_w, S_1)$ where

- S_0 and S_1 are vertex-sieves on V_0 and V_1 respectively, and
- Σ_w is a sieve on R in $\mathbf{G}_e$

such that

1. for every $\varphi \in \Sigma_w$, $\partial_0(\varphi) \in S_0$ and $\partial_1(\varphi) \in S_1$ and
2. for every morphism of the form $\tau\colon \mathbf{I}_U \to R$ with $\partial_0(\tau) = f_0$ and $\partial_1(\tau) = f_1$, we have[2]

$$\mathbf{I}^*_{f_0}((S_0)_w) = \mathbf{I}^*_{f_1}((S_1)_w) = \tau^*(\Sigma_w)$$

Using these definitions, we can show that the subobject classifier of $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ is as follows:

- $\Omega(V) =$ the set of vertex-sieves on V

[2] For any category $\mathbf{C}$, a morphism $f\colon C \to D$ induces a mapping of sieves on D to sieves on C, denoted f^*: $f^*(S) = \{g \mid f \circ g \in S\}$.

- $\Omega(f) = f^*$
- $\Omega(R) =$ the span with the set of edge-sieves on R as the witness set and projections sending each (S_0, Σ, S_1) to S_0 and S_1, respectively.
- $\Omega(\varphi) = (\Omega(\partial_0(\varphi)), \varphi^*, \Omega(\partial_1(\varphi)))$

The parametric natural transformation $T\colon 1 \to \Omega$, denoting the constant-true predicate, is defined by

$$T_V = (t_V, t_{\mathbf{I}_V})$$
$$T_R = (T_{V_0}, t_R, T_{V_1})$$

where t_X denotes the maximal sieve on X (in a category).

To get some intuition for these definitions, we consider the notion of a *matching family* (which will be used to define sheaves in the next section).

Definition Let $\mathbf{G}$ be a reflexive graph with identity condition and $F\colon \mathbf{G}^{\mathsf{op}} \to$ **Span** an RG-functor. If $S = (S_v, S_w)$ is a vertex-sieve on a vertex V, a *matching family* over S for F is a pair of families $(\{x_f \in F(\mathsf{dom}(f))\}_{f \in S_v}, \{s_\varphi \in F_w(\mathsf{dom}(\varphi))\}_{\varphi \in S_e})$ such that[3]

- $x_{f \circ g} = x_f \cdot g$;
- $s_{\varphi \circ \psi} = s_\varphi \cdot \psi$; and
- for all $\varphi\colon R \to \mathbf{I}_V$ in S_e with $F(R) = (F(U_0) \overset{\rho_0}{\leftarrow} F_w(R) \overset{\rho_1}{\to} F(U_1))$, $\rho_0(s_\varphi) = x_{\partial_0(\varphi)}$ and $\rho_1(s_\varphi) = x_{\partial_1(\varphi)}$.

More abstractly, a matching family is parametric natural transformation $(\mathbf{x}, \mathbf{s})\colon S \to F$ with S regarded as an RG-functor.

This definition shows the role played by edge-parts of sieves. The matching families must be consistent in that they must respect all the relationships denoted by the edges included in the sieve.

There is similarly a notion of matching families for edge-sieves.

Definition Let $\mathbf{G}$ be reflexive graph with identity condition and $F\colon \mathbf{G}^{\mathsf{op}} \to$ **Span** an RG-functor. If $\Sigma = (S_0, \Sigma_w, S_1)$ is an edge-sieve on an edge $R\colon V_0 \leftrightarrow V_1$ of $\mathbf{G}$, a *matching family* over Σ for F is a triple $((\mathbf{x}, \mathbf{s}), \mathbf{w}, (\mathbf{y}, \mathbf{t}))$ where

- $(\mathbf{x}, \mathbf{s})$ and $(\mathbf{y}, \mathbf{t})$ are matching families over S_0 and S_1 respectively, and
- $\mathbf{w} = \{w_\varphi \in F_w(\mathsf{dom}(\varphi))\}_{\varphi \in \Sigma_w}$

satisfying the following conditions:

1. $w_{\varphi \circ \varphi'} = w_\varphi \cdot \varphi'$

[3]For any RG-functor $F\colon \mathbf{G}^{\mathsf{op}} \to$ **Span**, we write $x \cdot f$ for $F(f)(x)$ and $w \cdot \varphi$ for $F_w(\varphi)(w)$.

2. for all $\varphi \in P \to R$ in Σ_w, with $F(P) = (F(V_0) \stackrel{\rho_0}{\leftarrow} F_w(P) \stackrel{\rho_1}{\rightarrow} F(V_1))$, $\rho_0(w_\varphi) = x_{\partial_0(\varphi)}$ and $\rho_1(w_\varphi) = x_{\partial_1(\varphi)}$

3. for every edge morphism of the form $P \stackrel{\psi}{\to} \mathbf{I}_U \stackrel{\varphi}{\to} R$ in Σ_w

$$w_{\varphi\circ\psi} = s_{\mathbf{I}_{\partial_0(\varphi)}\circ\psi} = t_{\mathbf{I}_{\partial_1(\varphi)}\circ\psi}$$

Relational Presheaves

Having given the definitions for the general case, we consider a special case of RG-presheaves which are encountered in Algol semantics. Note that we have a reflexive graph embedding $J\colon \mathbf{Rel} \hookrightarrow \mathbf{Span}$ which sends a relation $R\colon A_0 \leftrightarrow A_1$ to the span $A_0 \leftarrow R \rightarrow A_1$ with projections as arrows. We call an RG-presheaf $F\colon \mathbf{G}^{\mathsf{op}} \to \mathbf{Span}$ *relational* if it factors through this embedding upto isomorphism, i.e., $F \cong JF'$ for an RG-functor $F'\colon \mathbf{G}^{\mathsf{op}} \to \mathbf{Rel}$. So, we can regard $F_w(R)$, for any edge $R\colon V_0 \leftrightarrow V_1$ of G, as simply a relation $F(V_0) \leftrightarrow F(V_1)$. The subcategory of relational RG-presheaves in $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ is cartesian closed. However, it does not have a subobject classifier.

We show the notion of matching families for relational RG-presheaves because it is considerably simpler than the general case. If F is a relational RG-presheaf, a *matching family* over a vertex-sieve S for F is a family $\{x_f \in F(\mathsf{dom}(f))\}_{f\in S_v}$ such that

1. for all $f\colon U \to V$ in S_v and all vertex morphisms $g\colon U' \to U$, $x_{f\circ g} = x_f \cdot g$

2. for all $\varphi\colon R \to \mathbf{I}_V$ in S_e, $x_{\partial_0(\varphi)}\,[F_w(R)]\,x_{\partial_1(\varphi)}$.

We do not need a separate family of witnesses because the witnesses are unique whenever they exist. For an edge sieve $\Sigma = (S_0, \Sigma_w, S_1)$, a matching family is merely a pair of matching families $(\mathbf{x}, \mathbf{y})$ for S_0 and S_1 such that,

$$\text{for all } \varphi\colon P \to R \text{ in } \Sigma_w,\ x_{\partial_0(\varphi)}\,[F_w(P)]\,y_{\partial_1(\varphi)}.$$

4 RG-Sheaves

Sheaves can be defined internal to any topos using a Lawvere-Tierney topology [11]. Since $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ is a topos, we automatically have sheaves internal to it. We call them RG-sheaves.

Such an internal characterization is very abstract and hard to use in applications. In this section, we give an external characterization of RG-sheaves similar to the definition of sheaves in Section 2.1. We also show that it coincides with the internal characterization.

As in Section 2.1, we first need a Grothendieck topology J that assigns a collection of sieves to each vertex and edge of a reflexive graph $\mathbf{G}$.

Definition Let $\mathbf{G}$ be a small reflexive graph with identity condition. A *Grothendieck topology* on $\mathbf{G}$ is a pair $J = (J_v, J_w)$ where

- J_v assigns to each vertex of $\mathbf{G}$, a collection of vertex-sieves on V (called the "covers" of V); and
- J_w assigns to each edge $R\colon V_0 \leftrightarrow V_1$ of $\mathbf{G}$ a collection of edge-sieves on R (called the "covers" of R) such that the following conditions are satisfied:

 1. *Closed under reflexive graph structure*:
    - For each $(S_0, \Sigma_w, S_1) \in J(R)$, $S_0 \in J(V_0)$ and $S_1 \in J(V_1)$.
    - For each $S \in J(V)$, $(S, S_w, S) \in J(\mathbf{I}_V)$.
 2. *Maximality*: $T_V \in J(V)$ and $T_R \in J(R)$.
 3. *Stability*:
    - If $S \in J(V)$, then $\Omega(f)(S) \in J(V')$ for any $f'\colon V' \to V$.
    - If $\Sigma \in J(R)$, then $\Omega_w(\varphi)(\Sigma) \in J(R')$ for any $\varphi\colon R' \to R$.
 4. *Transitivity*:
    - If $S \in J(V)$ and S' is any vertex-sieve on V such that $\Omega(f)(S') \in J(U)$ for all $f\colon U \to V$ in S_v and $\Omega_w(\varphi)(S', S'_w, S') \in J(P)$ for all $\varphi\colon P \to \mathbf{I}_V$ in S_w, then $S' \in J(V)$.
    - If $\Sigma \in J(R)$ and $\Sigma' = (S'_0, \Sigma'_w, S'_1)$ on $R\colon V_0 \leftrightarrow V_1$ such that $S'_0 \in J(V_0)$, $S'_1 \in J(V_1)$ and $\Omega_w(\varphi)(\Sigma') \in J(P)$ for all $\varphi\colon P \to R$ in Σ_w, then $\Sigma' \in J(R)$.

 We call the pair $(\mathbf{G}, J)$ an *RG-site*.

The idea of covers, as mentioned in Section 2.1 is to identify certain sieves that are intended to force unique amalgamations. A Grothendieck topology is meant to pick out such covers. A simple example of a Grothendieck topology is called the *atomic topology*. (This will play a role in Algol semantics of Section 5.) Denoting it by the symbol $\mathcal{A}$, we have:

$\mathcal{A}_v(V) =$ the set of all vertex-sieves (S_v, S_w) such that S_v and S_w are nonempty.

$\mathcal{A}_w(R) =$ the set of all edge-sieves (S_0, Σ_w, S_1) such that $S_0 \in \mathcal{A}_v(V_0)$, $S_1 \in \mathcal{A}_v(V_1)$ and Σ_w is nonempty.

When for any two edge morphisms $\varphi\colon R' \to R$ and $\psi\colon R'' \to R$ in $\mathbf{G}$, there exist edge morphisms $\varphi'\colon P \to R'$ and $\psi'\colon P \to R''$ such that $\varphi \circ \varphi' = \psi \circ \psi'$, $\mathcal{A}$ satisfies all the axioms to be a Grothendieck topology on $\mathbf{G}$.

Matching families for vertex- and edge-sieves have already been defined in Section 3. So, we have the main definition:

Definition An *RG-sheaf* over an RG-site $(\mathbf{G}, J)$ is an RG-presheaf $F\colon \mathbf{G}^{\mathsf{op}} \to \mathbf{Span}$ such that

- for every vertex-sieve $S \in J(V)$ and matching family $(\mathbf{x}, \mathbf{s})$ over S, there is a unique amalgamation: an element $x \in F(V)$ such that $x_f = x \cdot f$ for all $f \in S_v$ and $s_\varphi = x \cdot \varphi$ for all $\varphi \in S_w$; and
- for every covering edge-sieve $\Sigma \in J(R\colon V_0 \leftrightarrow V_1)$ and matching family $((\mathbf{x}, \mathbf{s}), \mathbf{w}, (\mathbf{y}, \mathbf{t}))$ over Σ, there is a unique amalgamation: an element $w \in F_w(R)$, where $F(R) = F(V_0) \overset{\rho_0}{\leftarrow} F_w(R) \overset{\rho_1}{\to} F(V_1)$, such that $\rho_0(w)$ and $\rho_1(w)$ are the amalgamations of $(\mathbf{x}, \mathbf{s})$ and $(\mathbf{y}, \mathbf{t})$ respectively and $w_\varphi = w \cdot \varphi$ for all $\varphi \in \Sigma_w$.

It is instructive to consider the special case of relational RG-sheaves.

Lemma 3 Let $F\colon \mathbf{G}^{\mathsf{op}} \to \mathbf{Span}$ be a relational RG-presheaf. Then, it is an RG-sheaf iff

- for every $S \in J(V)$ and matching family $\mathbf{x}$ over S, there is a unique amalgamation: an element $x \in F(V)$ such that $x_f = x \cdot f$ for all $f \in S_v$.
- for every $\Sigma \in J(R\colon V_0 \leftrightarrow V_1)$ and matching family $(\mathbf{x}, \mathbf{y})$ over Σ, the amalgamations x of $\mathbf{x}$ and y of $\mathbf{y}$ satisfy $x\,[F_w(R)]\,y$.

In the next section, we will find several examples of relational RG-sheaves over atomic RG-sites.

It follows from the general results discussed in Section 4.1 that:

Theorem 4 An RG-sheaf category $\mathbf{Sh}(\mathbf{G}, J)$ is a topos (and, hence, cartesian closed.).

4.1 RG-Sheaves are Sheaves

The definition of RG-sheaves above involves excessive detail and it is hard to convince oneself that the definitions are right. In this section, we show that RG-sheaves are precisely the sheaves in the topos $\mathbf{Span}^{\mathbf{G}^{\mathsf{op}}}$ using Lawvere-Tierney topologies.

A Grothendieck topology $J = (J_v, J_w)$ can be easily turned into an RG-functor. By equipping each $J_w(R)$ with projections that send (S_0, Σ_w, S_1) to S_0 and S_1 respectively, we obtain a functor $J_e\colon \mathbf{G}_e^{\mathsf{op}} \to \mathbf{Span}_e$. (The morphism part comes from the stability condition.) The pair $J = (J_v, J_e)$ does form an RG-functor.

Since the sets involved in J are all sets of sieves, it is clear that $J \hookrightarrow \Omega$ is an RG-subfunctor.[4] The characteristic morphism of the subobject, $j\colon \Omega \to \Omega$,

[4]Given two RG-presheaves F and G, F is an RG-subfunctor of $\mathbf{G}$ if F_v and F_w are subfunctors of G_v and G_w respectively.

is a Lawvere-Tierney topology on $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$. Conversely, the kernel of every Lawvere-Tierney topology on $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$ is a Grothendieck topology (regarded as an RG-functor).

Lemma 5 There is one-to-one correspondence between Grothendieck topologies on a small reflexive graph $\mathbf{G}$ with identity condition and Lawvere-Tierney topologies on $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$.

Theorem 6 (Externalization) For any RG-site $(\mathbf{G}, J)$, an RG-functor $F\colon \mathbf{G}^{\mathsf{op}} \to \mathbf{Span}$ is an RG-sheaf iff it is a sheaf for $j = \mathsf{char}(J)$.

The whole structure of the proof for the theorem is similar to that for the corresponding theorem in the presheaf category case, shown in [11]. A straightforward extension of the proof in [11] shows only-if direction in $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$, but showing if-direction requires a somewhat new idea. We mainly focus on that direction.

Recall that F is a sheaf for j iff for all dense monomorphisms $\tau\colon G \to H$ and a morphism η, η can be uniquely extended along τ, i.e., there exists a unique η' such that $\eta = \tau \circ \eta'$. Dense monomorphisms in $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$ can be characterized in a manner very similar to the presheaf category case, which explains the reason that only-if direction can be shown easily by just extending the proof in the presheaf category. We have a characterization of monos in $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$:

Lemma 7 A parametric natural transformation $\eta\colon F \to G$ is monic in $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$ iff for any vertex V and edge R, $(\eta_v)_V$ and $(\eta_w)_R$ are injections.

For any monic parametric natural transformation, by the above lemma, we can assume that $(\eta_v)_V$ and $(\eta_w)_R$ are inclusions. The dense monomorphism has the following characterization:

Lemma 8 A monomorphism $\eta\colon F \hookrightarrow G$ in $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$ is dense for a Lawvere-Tierney topology j iff

1. for any vertex V in $\mathbf{G}_v$ and $x \in G(V)$,

$$(\{f\colon W \to V \mid x \cdot f \in F(W)\}, \{\varphi\colon R \to \mathbf{I}_V \mid x \cdot \varphi \in F_w(R)\}) \text{ is in } J(V)\text{; and}$$

2. for any edge $R\colon V_0 \leftrightarrow V_1$ in $\mathbf{G}_e$, $w \in G_w(R)$, $x \in G(V_0)$ and $y \in G(V_1)$ such that w is a witness for (x, y),

$$\begin{array}{c}\big((\{f\colon V' \to V_0 \mid x \cdot f \in F(V')\}, \{\varphi\colon R' \to \mathbf{I}_{V_0} \mid x \cdot \varphi \in F_w(R')\}\}), \\ \{\varphi\colon R' \to R \mid w \cdot \varphi \in F_w(R')\}\}, \\ (\{f\colon V' \to V_1 \mid y \cdot f \in F(V')\}, \{\varphi\colon R' \to \mathbf{I}_{V_1} \mid y \cdot \varphi \in F_w(R')\}\})\big) \\ \text{is in } J(R)\end{array}$$

The following lemma informally says that for any matching family for vertex- and edge-sieves, the existence of unique amalgamation can be equivalently expressed by the existence of an unique extension along a dense monomorphism, which implies if-direction of our main theorem. First, we deal with vertex-sieves:

Lemma 9 Let $(\mathbf{G}, J)$ be an RG-site.

1. A covering S on a vertex V induces a RG-subfunctor G of $\mathsf{Hom}[-, V]$, whose inclusion is dense. More precisely, G is defined by
   - $G_v(U) = \mathsf{Hom}[U, V] \cap S_v$
   - $G_w(R\colon V_0 \leftrightarrow V_1)) = \mathsf{Hom}[R, \mathbf{I}_V] \cap S_w$

2. A matching $(\mathbf{x}, \mathbf{s})$ of F over S has one-to-one correspondence with the parametric natural transformation η from G to F given by

   $$(\eta_v)_U(f) = x_f \text{ for all } f \in G_v(U) \text{ and } (\eta_w)_R(\varphi) = w_\varphi \text{ for all } \varphi \in G_w(R).$$

3. A matching family $(\mathbf{x}, \mathbf{s})$ of F has a unique amalgamation iff the corresponding parametric natural transformation η can be uniquely extended along the inclusion from G to $\mathsf{Hom}[-, V]$.

In the case of edge-sieves on an edge $R\colon V_0 \leftrightarrow V_1$, we need a more complicated RG-functor $\mathcal{H}(R)$, whose exact definition is shown in Appendix A. We only mention that $\mathcal{H}(R)_v(V)$ is defined a quotient of a disjoint union of $\hom_{\mathbf{G}_v}[V, V_0]$ and $\hom_{\mathbf{G}_v}[V, V_1]$ by an appropriate equivalence relation and $\mathcal{H}(R)_w(P)$ a quotient of a disjoint union of $\hom_{\mathbf{G}_e}[P, \mathbf{I}_{V_0}]$, $\hom_{\mathbf{G}_e}[P, \mathbf{I}_{V_1}]$ and $\hom_{\mathbf{G}_e}[P, R]$ by an appropriate equivalence relation. In fact, the very existence of such an RG-functor $\mathsf{Hom}[-, R]$ is the most interesting part of this whole proof.

Lemma 10 Let $(\mathbf{G}, J)$ be an RG-site.

1. A covering Σ on an edge R induces a RG-subfunctor G of $\mathcal{H}(R)$, whose inclusion is dense. More precisely, G is defined by
   - $G_v(U) = \{[f] \mid [f] \in \mathcal{H}(R)_v(U) \wedge (f \in (S_0)_v \vee f \in (S_1)_v)\}$
   - $G_w(P\colon U_0 \leftrightarrow U_1)) = \{[\varphi] \mid [\varphi] \in \mathcal{H}(R)_w(P) \wedge (\varphi \in (S_0)_w \vee \varphi \in (S_1)_w \vee \varphi \in \Sigma_w)\}$

2. A matching $((\mathbf{x}, \mathbf{s}), \mathbf{w}, (\mathbf{y}, \mathbf{t}))$ of F over Σ has one-to-one correspondence with the parametric natural transformation η from G to F given by

   $$(\eta_v)_U([f]) = \begin{cases} x_f & \text{if } f \in (S_0)_v \\ y_f & \text{if } f \in (S_1)_v \end{cases} \quad \text{and}$$
   $$(\eta_w)_R([\varphi]) = \begin{cases} s_\varphi & \text{if } \varphi \in (S_0)_w \\ w_\varphi & \text{if } \varphi \in \Sigma_w \\ t_\varphi & \text{if } \varphi \in (S_1)_w \end{cases}$$

3. A matching family $((\mathbf{x}, \mathbf{s}), \mathbf{w}, (\mathbf{y}, \mathbf{t}))$ of F has a unique amalgamation iff the corresponding parametric natural transformation η can be uniquely extended along the inclusion from G to $\mathcal{H}(R)$.

5 Semantics of Idealized Algol and Independence Predicate

In this section, we define the meaning of independence predicate in the parametricity semantics of Idealized Algol. The results in the previous section about RG-sheaves become essential in showing the well-definedness of the semantics.

5.1 Idealized Algol and Independence Predicate

Reynolds proposed Idealized Algol as a framework for higher order imperative languages just as PCF is so for functional languages. IA is a call-by-name typed lambda calculus with imperative base types such as com and exp and constant terms such as assignments and sequencing. Types are stratified into two layers: *data types* for storable values and *phrase types* for program terms. Usually, only integer and boolean values are regarded as storable entities so that only int and bool become data types. Phrase types are usual types in the lambda calculus sense, obtained from the command type (com), expression type (exp[int], exp[bool]) and variable type (var[int], var[bool]) by applying function and product type constructors. Basic imperative actions, such as assignment and sequencing, are incorporated as constant terms.

To simplify the presentation, we assume that only integers can be stored in variables, i.e., int is the only data type. We use exp and var to denote integer expressions and integer variables, respectively. Basic constants are also chosen according to this simplification. The recursion operator Y is not considered. The whole syntax of IA that we are using is shown in Table 1, whose meaning should be evident from its name. Note that the conditional statement can only be used with expressions or commands.[5]

Two terms are independent when they use disjoint resources, in particular, different memory locations. We write $M \perp N$ to denote that M and N are independent. The independence predicate is expected to satisfy the following basic principles [26, 28, 13]:

1. if $P \perp M$ and $N \perp M$, then $P(N) \perp M$;
2. if $N \perp i$ for all free identifiers i of M, then $N \perp M$; and
3. a non-local entity M and a local variable x are independent.

In [25], Reddy defined semantics of independence predicate in a parametricity model. As pointed out in the paper, he couldn't show that the second principle, which is called *decomposition axiom*, is valid in his semantics. In the next section, we present a semantics of IA and the independence predicate where all three principles are valid. Our focus is mainly on the second principle.

[5] Therefore, we don't allow bad variables.

Types:

data type	δ	$::=$	int
phrase type	θ	$::=$	exp | com | var | $\theta_1 \times \theta_2$ | $\theta_1 \to \theta_2$

Constant Terms:

$$\begin{array}{rcl} 0 & : & \mathsf{exp} \\ \mathsf{succ} & : & \mathsf{exp} \to \mathsf{exp} \\ \mathsf{pred} & : & \mathsf{exp} \to \mathsf{exp} \\ \mathsf{skip} & : & \mathsf{com} \\ \mathsf{seq} & : & \mathsf{com} \times \mathsf{com} \to \mathsf{com} \\ \mathsf{ifz} & : & \mathsf{exp} \times \beta \times \beta \to \beta \\ \mathsf{get} & : & \mathsf{var} \to \mathsf{exp} \\ := & : & \mathsf{var} \times \mathsf{exp} \to \mathsf{com} \\ \mathsf{new} & : & (\mathsf{var} \to \mathsf{com}) \to \mathsf{com} \end{array}$$

where $\beta = \mathsf{exp}$ or com.

Table 1: Syntax of Idealized Algol

5.2 RG-Sheaf Semantics

In the semantics in this section, IA types are interpreted as relational RG-sheaves on a certain RG-site. And the meaning of each terms is given as a parametric natural transformation. The semantics of IA described in this section is, in essence, similar to [19, 17, 25] with one minor difference, which is that each world denotes not an abstract set of states but a finite set of locations currently allocated on the memory. The main significant difference lies in showing that the independence predicate can be defined based on "finite support," as in [14], even in parametricity semantics of IA, thereby validating the decomposition principle. The theory of RG-sheaves in Section 4 is the key to obtaining the well-definedness of the definition.

Let $\mathbf{W}$ be the reflexive graph which parameterizes all interpretations of types by requiring them to be relational RG-functors from $\mathbf{W}^{\mathsf{op}}$ to **Span**. The two categories in the reflexive graph $\mathbf{W}$ model two different aspects. Objects and morphisms in the vertex category represent store shapes and changes of shapes by new variable allocations, and those in the edge category represent relations between states in two possibly different shapes of the store and preservations of those relations. The vertex category $\mathbf{W}_v$ is defined as follows:

- objects in $\mathbf{W}_v$ are finite sets, whose elements are thought of as "locations."
- morphisms $f : W \to V$ in $\mathbf{W}_v$: an injection from V to W.

Each object in the vertex category is called a *world*. Let $S(W)$ be $[W \to \mathbf{Int}]$, which intuitively denotes possible states at a world W. As shown in [17], each morphism $f\colon W \to V$ in $\mathbf{W}_v$ induces a projection map $\varphi_f\colon S(W) \to S(V)$ and a replacement map $\rho_f\colon S(W) \times S(V) \to S(W)$. They are defined as follows:

- $\varphi_f(w)(l) = w(f(l))$
- $\rho_f(w,v)(l) = \begin{cases} v(l') & \text{if } l = f(l') \text{ for some } l' \text{ in } V \\ w(l) & \text{otherwise} \end{cases}$

Intuitively, φ_f projects a smaller state from a larger state, and $\rho_f(w,v)$ replaces the V part of w by v.

The edge category $\mathbf{W}_e$ is defined as follows:

- objects are triples (W_0, R, W_1) where W_0 and W_1 are worlds (objects in the vertex category $\mathbf{W}_v$) and R is a nonempty relation between $S(W_0)$ and $S(W_1)$, that is, a subset of $S(W_0) \times S(W_1)$.
- morphisms from (W_0, R, W_1) to (X_0, S, X_1) are pairs $(f\colon W_0 \to X_0, g\colon W_1 \to X_1)$ of morphisms in the $\mathbf{W}_v$ such that
  1. $\rho_f[R \times S \to R]\rho_g$; and
  2. $\varphi_f[R \to S]\varphi_g$.

The Grothendieck RG-topology on $\mathbf{W}$, which we use to give the semantics of IA, is an atomic topology $\mathcal{A}$ explained in Section 4. Each type in IA is interpreted as a relational RG-sheaf on a site $(\mathbf{W}, \mathcal{A})$. Since relational RG-sheaves always form a sub-CCC of $\mathbf{Span}^{\mathbf{W}^{\mathrm{op}}}$, it is enough to define base types as relational RG-sheaves. We use the fact that each type is relational to simplify presentation: we regard $[\![\theta]\!]_w(R\colon V_0 \leftrightarrow V_1)$ as a relation $F(V_0) \leftrightarrow F(V_1)$ and omit the description of $[\![\theta]\!]_e$ other than presenting what relation $[\![\theta]\!]_w(R)$ is for all edges R because $[\![\theta]\!]_e$ can be recovered upto isomorphism from all other existing data [19].

- command:
  - $[\![\mathsf{com}]\!](W) = S(W) \to S(W)$
  - $[\![\mathsf{com}]\!](f\colon W \to V) = \lambda c\colon [\![\mathsf{com}]\!](V).\, \lambda s\colon S(W).\, \rho_f(s, c(\varphi_f(s)))$
  - $[\![\mathsf{com}]\!](R\colon W_0 \leftrightarrow W_1) = [R \to R]$
- expression:
  - $[\![\mathsf{exp}]\!](W) = S(W) \to \mathbf{Int}$
  - $[\![\mathsf{exp}]\!](f\colon W \to V) = \lambda e\colon [\![\mathsf{exp}]\!](V).\, \lambda s\colon S(W).\, e(\varphi_f(s))$
  - $[\![\mathsf{exp}]\!](R) = [R \to \Delta_{\mathbf{Int}}]$
- variable:
  - $[\![\mathsf{var}]\!](W) = (\mathbf{Int} \to [\![\mathsf{com}]\!](W)) \times [\![\mathsf{exp}]\!](W)$
  - $[\![\mathsf{var}]\!](f\colon W \to V) =$
    $\lambda v\colon [\![\mathsf{var}]\!](V).\, ([\![\mathsf{com}]\!](f) \circ \mathsf{fst}(v),\ [\![\mathsf{exp}]\!](f)(\mathsf{snd}(v)))$

– $[\![\mathsf{var}]\!](R) = [\Delta_{\mathbf{Int}} \to [\![\mathsf{com}]\!](R)] \times [\![\mathsf{exp}]\!](R)$

In Section 4, the condition for being RG-sheaves is expressed by the existence of a unique amalgamation for every matching family. The following lemma states equivalent characterization for the RG-site $(\mathbf{W}, \mathcal{A})$:

Lemma 11 For any relational RG-functor F in $\mathbf{Span}^{\mathbf{W}^{\mathrm{op}}}$, F is an RG-sheaf on $(\mathbf{W}, J)$ iff

- for any vertex morphism $f\colon V \to W$ in $\mathbf{W}$ and any $y \in F(V)$, if $y \cdot g = y \cdot h$ for all diagrams

$$U \underset{h}{\overset{g}{\rightrightarrows}} V \xrightarrow{\;f\;} W$$

  with $f \circ g = f \circ h$, then $y = x \cdot f$ for a unique $x \in F(W)$ (this is equivalent to requiring that F_v is an atomic sheaf [11]); and

- for any edge morphism $\varphi\colon P \to R$ to a non-identity edge R in $\mathbf{W}$, if $x \cdot \partial_0(\varphi)\,[F_w(P)]\,y \cdot \partial_0(\varphi)$, then $x\,[F_w(R)]\,y$.

The first condition is equivalent to saying that every matching family for any covering vertex-sieve in $\mathcal{A}(W)$ has a unique amalgamation. To see the equivalence, notice two facts: f and y in the hypothesis can generate a vertex-sieve in $\mathcal{A}(W)$ and a matching family over it respectively, and the condition implies that $F(f)$ is injective. Intuitively, the condition says that if renaming locations in $V - f(W)$ in two different ways doesn't change the meaning of a program, the program must access only the locations in $f(W)$. And under the first condition, the second condition is also equivalent to requiring that all matching families for all edge-sieves in $\mathcal{A}(R)$ have unique amalgamations. The equivalence follows from a similar reason: an edge-sieve in $\mathcal{A}(R)$ and a matching family over it can be generated by φ and, $x \cdot \partial_0(\varphi)$ and $y \cdot \partial_1(\varphi)$, respectively. The second condition, combined with the fact that an RG-functor always satisfies the converse, i.e., $x\,[F_w(R)]\,y \Longrightarrow x \cdot \partial_0(\varphi)\,[F_w(P)]\,y \cdot \partial_0(\varphi)$, says that an expansion doesn't change being-relatedness.

Another important observation is that the vertex part of an RG-sheaf preserves pullbacks [7].

Lemma 12 For all RG-sheaves F over $(\mathbf{W}, \mathcal{A})$, F_v preserves pullbacks in $\mathbf{W}_v$.

Proof: In [7], it is shown that the atomic sheaves over $\mathbf{W}_v$ preserve pullbacks. The conclusion follows because F_v satisfies the first condition in Lemma 11, which is an equivalent condition for being an atomic sheaf over $\mathbf{W}_v$. □

In order to see the computational interpretation of pullback preservation, let W, V and X be worlds such that $V, X \subseteq W$. And consider the denotation w of a program phrase of type θ at a world W ($w \in [\![\theta]\!](W)$), which only accesses X and V: there exist $x \in [\![\theta]\!](X)$ and $y \in [\![\theta]\!](V)$ such that $x \cdot i = w$ and $y \cdot i' = w$

for the inclusion i of X and i' of V in W. The computational intuition says that w only accesses $X \cap V$. The pullback preservation captures this: there exists a unique $z \in [\![\theta]\!](X \cap V)$ such that $z \cdot k = w$ for the inclusion k of $X \cap V$ in W.

Proposition 13 $[\![\mathsf{com}]\!]$, $[\![\mathsf{exp}]\!]$ and $[\![\mathsf{var}]\!]$ are RG-sheaves over the RG-site $(\mathbf{W}, J)$.

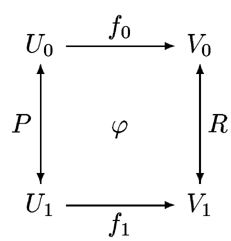
Proof: We show the conditions in Lemma 11 hold for $[\![\mathsf{exp}]\!]$. The case of $[\![\mathsf{com}]\!]$ is similar, and that of $[\![\mathsf{var}]\!]$ follows from the fact that $[\![\mathsf{var}]\!] \cong (\mathbf{Int} \Rightarrow [\![\mathsf{com}]\!]) \times [\![\mathsf{exp}]\!]$.

For the first condition in Lemma 11, let $y \in [\![\mathsf{exp}]\!](V)$ be such an expression. By the definition of $\mathbf{W}$, given $s \in S(W)$, there exists $s' \in S(V)$ such that $\varphi_f(s') = s$. Define $x \in [\![\mathsf{exp}]\!](W)$ by $x(s) = y(s')$ for some $s' \in S(V)$ such that $\varphi_f(s') = s$. The hypothesis on y implies that $x(s)$ doesn't depend on the choice of s'. More precisely, suppose $s', s'' \in S(V)$ such that $\varphi_f(s') = \varphi_f(s'')$. Then, there exist $g, h\colon U \to V$ and $r \in S(U)$ such that

$$(\varphi_g(r) = s') \wedge (\varphi_h(r) = s'') \wedge (g \circ f = h \circ f)$$

Then, by assumption, we have $y(s') = y(\varphi_g(r)) = y(\varphi_h(r)) = y(s'')$. It is now easy to show $x \cdot f = y$, which is the existence part. The uniqueness is straightforward.

For the second condition, let φ be an edge morphism in $\mathbf{G}$ of the form:



where R is a non-identity edge. We should show that for any $x \in [\![\mathsf{exp}]\!](V_0)$ and y in $[\![\mathsf{exp}]\!](V_1)$ such that $x \cdot f_0\, [[\![\mathsf{exp}]\!]_w(P)]\, y \cdot f_1$, we have $x\, [[\![\mathsf{exp}]\!]_w(R)]\, y$. First note that for any $s_0 \in S(V_0)$ and $s_1 \in S(V_1)$ such that $s_0\, R\, s_1$, there exists $r_0 \in S(U_0)$ and $r_1 \in S(U_1)$ such that $\varphi_{f_0}(r_0) = s_0$ and $\varphi_{f_1}(r_1) = s_1$ and $r_0\, P\, r_1$ by the nonemptiness of P and the definition of edge morphisms in $\mathbf{W}_e$. Since $x \cdot f_0\, [[\![\mathsf{exp}]\!]_w(P)]\, y \cdot f_1$, $(x \cdot f_0)(r_0) = (y \cdot f_1)(r_1)$. By unrolling the definition, $x(\varphi_{f_0}(r_0)) = y(\varphi_{f_1}(r_1))$, from which $x(s_0) = y(s_1)$ follows. □

Recall that intuitively, two program phrases do not interfere with each other when they use disjoint resources. Formalizing this requires us to define accurately what "used resources" means, which is done using the notion of *support*:

Definition For any RG-sheaf F over the site $(\mathbf{W}, \mathcal{A})$ and $a \in F(W)$, a subset W' of W is called the *support* of a if it is the least set with an element $a' \in F(W')$ such that $a' \cdot i = a$ where $i\colon W \to W'$ is an inclusion from W' to W.

For instance, the supports of the following commands at a world $\{l_0, l_1\}$ are as follows:

command	support
$\lambda s : S(\{l_0, l_1\}). s[l_0 \mapsto 1, l_1 \mapsto 1]$	$\{l_0, l_1\}$
$\lambda s : S(\{l_0, l_1\}). s[l_0 \mapsto 1]$	$\{l_0\}$
$\lambda s : S(\{l_0, l_1\}). s$	$\emptyset$

One might wonder when support exists. This is the place where the extra structure of being RG-sheaf plays a role.

Proposition 14 For any RG-sheaf F over the site $(\mathbf{W}, \mathcal{A})$, every $a \in F(W)$ has a support.

Proof: For any RG-sheaf F over the site $(\mathbf{W}, \mathcal{A})$, the vertex part F_v preserves pullbacks by Lemma 12. The existence of support follows from this pullback preservation of F_v [14]. □

We write $\mathsf{supp}(a)$ to denote the support of a. With the definition of support, the meaning of independence can easily be obtained by interpreting "disjoint" in the intuition literally.

Definition For any $a \in [\![\theta_1]\!](W)$ and $b \in [\![\theta_2]\!](W)$, a and b are independent ($a \perp b$) iff $\mathsf{supp}(a) \cap \mathsf{supp}(b) = \emptyset$.

The remaining question is whether this definition of independence is satisfactory in the sense that all the principles in Section 5.1 are valid with this semantics. This is, in fact, the case. The following lemma indicates why the second principle, the problematic one, is valid in the model:

Lemma 15 For any $p \in [\![\theta]\!](W)$, $p_1 \in [\![\theta_1]\!](W)$ and $p_2 \in [\![\theta_2]\!](W)$, if $p \perp p_1$ and $p \perp p_2$, then $p \perp (p_1, p_2)$.

Proof: Since $p \perp p_1$ and $p \perp p_2$, $\mathsf{supp}(p) \cap (\mathsf{supp}(p_1) \cup \mathsf{supp}(p_2)) = \emptyset$. Since $\mathsf{supp}((p_1, p_2)) = \mathsf{supp}(p_1) \cup \mathsf{supp}(p_2)$, the conclusion follows. □

6 Conclusion

Sheaves have been used extensively in geometry and logic to model various phenomena that are characterized as being "local." Their interest for programming is that they form toposes thereby allowing typed lambda calculi as well as programming logics to be interpreted.

However, in computer science applications one also encounters the phenomenon of "information hiding" which is best captured by relational (and other forms of) parametricity. Our contribution in this paper is to develop a theory of sheaves that accounts for parametricity. This fits into a broader program of generalizing categorical ideas to reflexive graphs so that a general axiomatization of concepts for semantics can be formulated.

The sheaves identified in this paper are those internal to the topos $\mathbf{Span}^{\mathbf{G}^{op}}$. It is possible that there are other notions of sheaves over reflexive graph. For instance, $\mathbf{Span}^{\mathbf{G}^{op}}$ is merely a category but also has a reflexive graph structure [2]. Examining this structure might give other notions of sheaves. We leave this question for future work.

Coming to the semantics of imperative programming, much of the work to date has focused on addressing one semantic feature at a time (examples being locality, noninterference, independence, state change irreversibility etc.). We are now beginning to see a consolidation, whereby multiple semantic features are integrated into a single model. For instance, O'Hearn and Reynolds [16] developed a linear parametricity model that combines locality and irreversibility. Ours is another such step: combining locality and independence.

We have considered a stripped down programming language for the sake of simplicity. Other feature can be added without much difficulty. For instance, adding classes as in [25] poses no problem. Recursion can be accommodated by using spans over **Cpo**, instead of spans over **Set**. Adding pointers (dynamically allocated data) is, however, nontrivial and pointers to higher-order objects could lead to as yet unexplored territory. We have these extensions to future work.

References

[1] E. S. Bainbridge, P. Freyd, A. Scedrov, and P. J. Scott. Functorial polymorphism. *Theoretical Comput. Sci.*, 70:35–64, 1990.

[2] B. P. Dunphy. Prelim proposal. Technical report, Department of Mathematics, University of Illinois at Urbana-Champaign, Jan 2000.

[3] M. P. Fiore and A. K. Simpson. Lambda definability with sums via grothendieck logical relations. In *Typed Lambda Calculi and Applications - TLCA '99*, volume 1581 of *LNCS*, pages 147–161. Springer-Verlag, 1999.

[4] M. J. Gabbay and A. M. Pitts. A new approach to abstract syntax involving binders. In *14th Annual Symposium on Logic in Computer Science*, pages 214–224. IEEE Computer Society Press, Washington, 1999.

[5] J. Goguen. Sheaf semantics for concurrent interacting objects. *Math. Struct. Comput. Sci.*, 2:159–191, 1992.

[6] B. P. Hilken. Relational sheaves and predicate intuitionistic modal logic. May. 1999.

[7] P. T. Johnstone. A topos-theorist looks at dilators. *Journal of Pure and Applied Algebra*, 58:235–249, 1989.

[8] Y. Kinoshita, P. W. O'Hearn, A. J. Power, M. Takeyama, and R. D. Tennent. An axiomatic approach to binary logical relations with applications to data refinement. Manuscript, Queen Mary and Westfield, London, 1997.

[9] J. Lambek and P. J. Scott. *Introduction to Higher Order Categorical Logic.* Cambridge University Press, Cambridge, 1986.

[10] QingMing Ma and J. C. Reynolds. Types, abstraction, and parametric polymorphism, part 2. In *Mathematical Foundations of Programming Semantics, 7th International Conference*, volume 598 of *LNCS*, pages 1–40. Springer-Verlag, 1992.

[11] S. MacLane and I. Moerdijk. *Sheaves in Geometry and Logic.* Springer-Verlag, 1991.

[12] A. R. Meyer and K. Sieber. Towards fully abstract semantics for local variables. In *Fifteenth Ann. ACM Symp. on Princ. of Program. Lang.*, pages 191–203. ACM, 1988. (Reprinted as Chapter 7 of [20]).

[13] P. W. O'Hearn. *Semantics of Noninterference: A Natural Approach.* PhD thesis, Queen's University, Kingston, Canada, 1990.

[14] P. W. O'Hearn. A model for syntactic control of interference. *Math. Struct. Comput. Sci.*, 3:435–465, 1993.

[15] P. W. O'Hearn, A. J. Power, M. Takeyama, and R. D. Tennent. Syntactic control of interference revisited. In S. Brookes, M. Main, A. Melton, and M. Mislove, editors, *Mathematical Foundations of Programming Semanatics: Eleventh Annual Conference*, volume 1 of *Electronic Notes in Theor. Comput. Sci.* Elsevier, 1995. (Reprinted as Chapter 18 of [20]).

[16] P. W. O'Hearn and J. C. Reynolds. From Algol to polymorphic linear lambda-calculus. Electronic manuscript, Queen Mary and Westfield, London, April 1997. URL http://www.qmw.ac.uk/ ohearn.

[17] P. W. O'Hearn and R. D. Tennent. Relational parametricity and local variables. In *Twentieth Ann. ACM Symp. on Princ. of Program. Lang.*, pages 171–184. ACM, 1993.

[18] P. W. O'Hearn and R. D. Tennent. Semantical analysis of specification logic, Part 2. *Inf. Comput.*, 107(1):25–57, 1993. (Reprinted as Chapter 14 of [20]).

[19] P. W. O'Hearn and R. D. Tennent. Parametricity and local variables. *J. ACM*, 42(3):658–709, 1995. (Reprinted as Chapter 16 of [20]).

[20] P. W. O'Hearn and R. D. Tennent. *Algol-like Languages (Two volumes).* Birkhäuser, Boston, 1997.

[21] F. J. Oles. *A Category-Theoretic Approach to the Semantics of Programming Languages.* PhD thesis, Syracuse University, 1982.

[22] E. Palmgren. Constructive sheaf semantics. *Mathematical Logic Quarterly*, 43:321–225, 1997.

[23] A. M. Pitts. Relational properties of domains. *Inf. Comput.*, 15(2):66, Jun 1996.

[24] A. M. Pitts and I. D. B. Stark. On the observable properties of higher-order functions that dynamically create local names. In *ACM SIGPLAN Workshop on State in Program. Lang.*, pages 31–45. Technical Report YALEU/DCS/RR-968, Yale University, New Haven, 1993.

[25] U. S. Reddy. Objects and classes in Algol-like languages. In *Fifth Intern. Workshop on Foundations of Object-oriented Languages.* electronic proceedings at http://pauillac.inria.fr/˜ remy/fool/proceedings.html, Jan 1998.

[26] J. C. Reynolds. Syntactic control of interference. In *ACM Symp. on Princ. of Program. Lang.*, pages 39–46. ACM, 1978. (Reprinted as Chapter 10 of [20]).

[27] J. C. Reynolds. The essence of Algol. In J. W. de Bakker and J. C. van Vliet, editors, *Algorithmic Languages*, pages 345–372. North-Holland, 1981. (Reprinted as Chapter 3 of [20]).

[28] J. C. Reynolds. Idealized Algol and its specification logic. In D. Neel, editor, *Tools and Notions for Program Construction*, pages 121–161. Cambridge Univ. Press, 1982. (Reprinted as Chapter 6 of [20]).

[29] J. C. Reynolds. Types, abstraction and parametric polymorphism. In R. E. A. Mason, editor, *Information Processing '83*, pages 513–523. North-Holland, Amsterdam, 1983.

[30] E. Robinson and G. Rosolini. Reflexive graphs and parametric polymorphism. In *Proceedings, Ninth Annual IEEE Symposium on Logic in Computer Science.* IEEE Computer Society Press, July 1994.

[31] Ian Stark. Categorical models for local names. *Lisp and Symbolic Computation*, 9(1):77–107, February 1996.

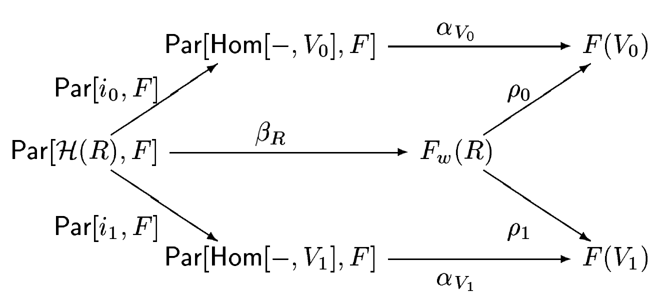
[32] R. D. Tennent. Semantical analysis of specification logic. *Inf. Comput.*, 85(2):135–162, 1990. (Reprinted as Chapter 13 of [20]).

A Yoneda Lemma in $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$

Given a small reflexive graph $\mathbf{G}$ with identity condition, there are two functors from $\mathbf{G}_v$ and $\mathbf{G}_e$ to $\mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$ respectively, for which a version of Yoneda lemma holds. Let Hom be defined as in Section 3. Then, the precise statement is as follows:

Lemma 16 Let $\mathbf{G}$ be a small reflexive graph with identity condition.

- For every vertex V in $\mathbf{G}$, there is a bijection $\alpha_V \colon \mathsf{Par}[\mathsf{Hom}[-,V],F] \cong F(V)$, which is natural in V.
- There exists $\mathcal{H} \colon \mathbf{G}_e \to \mathbf{Span}^{\mathbf{G}^{\mathrm{op}}}$ (whose definition is given below) such that for every edge R in $\mathbf{G}$, there is a bijection $\beta_R \colon \mathsf{Par}[\mathcal{H}(R),F] \cong F_w(R)$, which is natural in R.
- For all edges $R \colon V_0 \leftrightarrow V_1$ in $\mathbf{G}$, the above isomorphisms make the following diagram commute



where $F_e(R) = F(V_0) \overset{\rho_0}{\leftarrow} F_w(R) \overset{\rho_1}{\rightarrow} F(V_1)$ and $i_j \colon \mathsf{Hom}[-,V_j] \to \mathcal{H}(R)$ maps an element to an equivalent class containing it. The precise definition of i_j is given by:

– $\big((i_j)_v\big)_U(f) = [f]$

– $\big((i_j)_w\big)_P(\varphi) = [\varphi]$

The construction of $\mathcal{H}(R)$ is not as straightforward as $\mathsf{Hom}[-,V]$. It involves taking disjoint unions and quotients by some equivalence relations in order to make $\mathcal{H}(R)$ have the identity extension property. For any $R\colon V_0 \leftrightarrow V_1$, $\mathcal{H}(R)$ is defined as follows:

- $\mathcal{H}(R)_v(U) = (\mathsf{hom}_{\mathbf{G}_v}[U,V_0]{+}\mathsf{hom}_{\mathbf{G}_v}[U,V_1])\,/\approx_U$ where $\approx_U$ is the minimal equivalence relation containing $\sim_U$ which is defined as follows:

  $f_0 \sim_U f_1$ iff there exists $\varphi\colon \mathbf{I}_U \to R$ such that $\partial_0(\varphi) = f_0$ and $\partial_1(\varphi) = f_1$

- $\mathcal{H}(R)_v(g)[f] = [f \circ g]$
- $\mathcal{H}(R)_w(P\colon U_0 \leftrightarrow U_1) = (\mathsf{hom}_{\mathbf{G}_e}[P,\mathbf{I}_{V_0}]{+}\mathsf{hom}_{\mathbf{G}_e}[P,\mathbf{I}_{V_1}]{+}\mathsf{hom}_{\mathbf{G}_e}[P,R])\,/\approx_P$ where $\approx_P$ is the minimal equivalence relation containing $\sim_P$ which is defined as follows:

  $\varphi \sim_P \psi$ iff ψ is an edge morphism of the form $P \xrightarrow{\tau} \mathbf{I}_W \xrightarrow{\kappa} R$ for some vertex W such that $\varphi = \mathbf{I}_{\partial_0(\kappa)} \circ \tau$ or $\varphi = \mathbf{I}_{\partial_1(\kappa)} \circ \tau$

- the projections in $\mathcal{H}(R)_e$ send $[\varphi]$ to $[\partial_0(\varphi)]$ and $[\partial_1(\varphi)]$.
- $\mathcal{H}(R)_w(\varphi')[\psi] = [\psi \circ \varphi']$

The action on a morphism φ by $\mathcal{H}$ is defined by picking up an element in the equivalent class and composing it with one of $\partial_i(\varphi)$, $\mathbf{I}_{\partial_i(\varphi)}$ and φ:

- $\big(\mathcal{H}(\varphi)_v\big)_U[f] = \begin{cases} [\partial_0(\varphi) \circ f] & \text{if } f \in \mathsf{hom}_{\mathbf{G}_v}[U,V_0] \\ [\partial_1(\varphi) \circ f] & \text{if } f \in \mathsf{hom}_{\mathbf{G}_v}[U,V_1] \end{cases}$
- $\big(\mathcal{H}(\varphi)_w\big)_P[\psi] = \begin{cases} [\mathbf{I}_{\partial_0(\varphi)} \circ \psi] & \text{if } \psi \in \mathsf{hom}_{\mathbf{G}_e}[P,\mathbf{I}_{V_0}] \\ [\mathbf{I}_{\partial_1(\varphi)} \circ \psi] & \text{if } \psi \in \mathsf{hom}_{\mathbf{G}_e}[P,\mathbf{I}_{V_1}] \\ [\varphi \circ \psi] & \text{if } \psi \in \mathsf{hom}_{\mathbf{G}_e}[P,R] \end{cases}$

The Lemma 16 can used to simplify the presentation of concepts in the main text, such as vertex- and edge-sieves and matching families. We already noted that each vertex-sieve on V can be regarded as an RG-subfunctor of $\mathsf{Hom}[-,V]$. Similarly, edge-sieves on R can be thought of as RG-subfunctors of $\mathcal{H}(R)$, which is easily shown by Lemma 16. Recall that $\Omega_w(R)$ is defined to be a set of edge-sieves on R.

$$\begin{aligned} \Omega_w(R) &\cong \mathsf{Par}[\mathcal{H}(R),\Omega] \\ &\cong \mathsf{Sub}(\mathcal{H}(R)) \end{aligned}$$

In fact, the constructions of RG-functors from vertex-sieves in Lemma 9 and from edge-sieves in Lemma 10 show the isomorphisms directly. Also, via these isomorphisms, all matching families on vertex-sieves and edge-sieves can be considered as parametric natural transformations. The concrete construction was also shown in Lemma 9 and 10. This construction is bijective. That is, given a vertex-sieve S on V, the set of matching family for F over S is isomorphic to $\mathsf{Par}[S, F]$, and a similar result holds for edges.