Categorical Assertion Semantics in Toposes

Kawahara, Yasuo
Research Institute of Fundamental Information Science, Kyushu University

Mizoguchi, Yoshihiro
Deptartment of Control Engineering and Science, Kyushu Institute of Technology

出版情報：Advances in Software Science and Technology. 4, pp.137-150, 1992. Japan Society for Software Science and Technology
バージョン：

Categorical Assertion Semantics in Toposes

Yasuo Kawahara Yoshihiro Mizoguchi

Summary. A categorical interpretation of assertion(axiomatic) semantics of programming languages is proposed. All of the preconditions, postconditions and programs are interpreted as (binary) relations in toposes by making use of relational calculus, and several fundamental properties of Dijkstra's weakest preconditions are proved. Assertions in the semantics depend on the intuitionistic logic, so this is an extension of the assertion semantics due to E.G. Manes and M.A. Arbib.

1 Introduction

The objective of the paper concerns the mathematical foundation of program semantics from the standpoint of category theory [11, 12, 13]. Assertion (axiomatic) semantics is a methodology based on the use of assertions as preconditions and postconditions in program specifications associated with R. Floyd, C.A.R. Hoare, and E. Dijkstra. An assertion in the semantics is a statement about program states which is either true or false. For a program f and assertions p, q, we write a program with assertions as $\{p\}f\{q\}$ following to Hoare [6]. The program specification $\{p\}f\{q\}$ is correct if the satisfaction of precondition p about the input data guarantees the truth of postcondition q.

On the other hand, denotational semantics focuses on input/output behavior and ignores the intermediate states. That is, a program is considered as a function from input states to output states.

E.G. Manes and M.A. Arbib [11] have constructed a theory of program semantics reconciling assertion and denotational semantics with the use of category

ISBN 0–12–037103–0

theory. In their semantics an assertion is regarded as a guard function corresponding to a subset of a state set. A program is expressed by a function between state sets as in denotational semantics. They re-establish Hoare's notation $\{p\}f\{q\}$ with the composite of semantic functions and guard functions, and Dijkstra's weakest preconditions [4] are formulated using the notion of kernel-domain decompositions. Then the elementary properties of program semantics are demonstrated within the categorical framework.

It is an interesting problem to find a category in which assertions and programs will be naturally interpreted in terms of morphisms, and the fundamental properties of program semantics will be easily obtained. To this endManes [12] presented the concept of control categories. However they are imposed on many axioms, making it hard to verify whether a given category is a control category or not.

In this paper the authors investigate another categorical framework of assertion semantics of programming languages, slightly different from that of Manes and Arbib [11, 12]. We limit semantic categories to toposes [5, 7], which are the most typical and useful categories. Our background is relational calculus [9] in elementary toposes. All of the preconditions, postconditions and programs are interpreted as (binary) relations in toposes. The interpretation of conditional statements in our semantics is based on Heyting algebra [5], which is an algebra of intuitionistic (constructive) logic, and essentially different from the approach due to Manes and Arbib. Generally the truth value of a test expression q in a conditional statement **if** q **then** α **else** β depends on program executions which may not terminate. However, since interpretation with classical logic always imposes a value of true or false on test expressions unnaturally, our assertion semantics with intuitionistic logic is reasonable. In order to characterize Dijkstra's weakest preconditions without using the concept of kernel-domain decompositions, we introduce the notions of kernels and domains of relations in toposes. Then we reexamine partial and total correctness on Dijkstra's weakest and weakest liberal preconditions.

The results of the paper show that our interpretation is not only consistent, but also relational calculus in toposes is available for discussing mathematical foundations of programs.

2 Domains and Kernels of Relations

In this section we state a part of the relational calculus in elementary toposes, that is, some basic properties on (binary) relations and partial functions. For the details of relational calculus the reader is referred to M.S. Calenko [1, 2] and Y. Kawahara [8, 9].

Let $\mathbf{E}$ be an elementary topos [5, 7]. A morphism f of $\mathbf{E}$ is denoted by $f : X \to Y$ when it has domain X and codomain Y. The composite of two morphisms $f : X \to Y$ and $g : Y \to Z$ of $\mathbf{E}$ is written as $fg : X \to Z$. A span (f, g) in $\mathbf{E}$ is a pair of morphisms $x : Z \to X$ and $y : Z \to Y$ in $\mathbf{E}$ with a common domain. Given a span (f, g) in $\mathbf{E}$ there exists a unique morphism $h : Z \to X \times Y$ in $\mathbf{E}$ such that $hp = f$ and $hq = g$, where $p : X \times Y \to X$, $q : X \times Y \to Y$ are projections of a product $X \times Y$. We denote such a unique h by $x \top y$. By the basic property of toposes every morphism $f : X \to Y$ can be uniquely decomposed into a composite of an epimorphism $e(f) : X \to I$ followed by a monomorphism $m(f) : I \to Y$ up to isomorphisms. The subsequent arguments of this paper will be carried out in a fixed elementary topos $\mathbf{E}$.

A *relation* α from an object X to another object Y in $\mathbf{E}$, denoted by $\alpha : X \multimap Y$, is a subobject of the product $X \times Y$. Every span (f, g) with $f : R \to X$ and $g : R \to Y$ induces a relation $[m(f \top g)] : X \multimap Y$, which will be simply denoted by $[f, g]$. Recall that the subobject $[m(f \top g)]$ of $X \times Y$ presents an equivalence class of a monomorphism $m(f \top g)$. It is trivial that each relation α can be written as $\alpha = [f, g]$ with some span (f, g). We usually identify a morphism $f : X \to Y$ with a relation $[\mathrm{id}_X, f]$, where id_X is the identity morphism of X. Remark that $f = [\mathrm{id}_X, f] = [\mathrm{id}_X \top f]$ since $\mathrm{id}_X \top f$ is a monomorphism.

The composite $\alpha\beta (= \alpha \cdot \beta) : A \multimap C$ of a relation $\alpha : A \multimap B$ followed by a

relation $\beta : B \multimap C$ is defined as follows: Assume that $\alpha = [f,g]$ and $\beta = [h,k]$. Construct a pullback

$$\begin{array}{ccc} \cdot & \xrightarrow{h'} & \cdot \\ {\scriptstyle g'}\downarrow & & \downarrow{\scriptstyle g} \\ \cdot & \xrightarrow[h]{} & B \end{array}$$

and define $\alpha\beta = [h'f, g'k]$. Also the inverse $\alpha^\sharp : B \multimap A$ of α is defined by $\alpha^\sharp = [g,f]$. These definitions of the composition and the inverse of relations are well-defined. It is easy to see that $[f,g] = [f,\mathrm{id}][\mathrm{id},g] = f^\sharp g$.

These definitions are natural extensions of composite of functions and relations in the category **Set** of sets and functions.

Example 2.1 *A relation $\alpha : A \multimap B$ in* **Set** *is a subset of $A \times B$. The inverse relation $\alpha^\sharp : B \multimap A$ is a subset $\{(b,a) \in B \times A \,|\, (a,b) \in \alpha\}$. The composite $\alpha \cdot \beta : A \multimap C$ of two relations $\alpha : A \multimap B$ and $\beta : B \multimap C$ is a subset $\{(a,c) \,|\, (a,b) \in \alpha$ and $(b,c) \in \beta$ for some $b \in B\}$. A function $f : X \to Y$ is considered as a relation $\{(x,y) \in X \times Y \,|\, y = f(x)\}$ from X to Y.*

Since the set $\mathbf{Rel}(A,B)$ of all relations from A to B coincides with the set of all subobjects of $A \times B$, the ordering of relations in $\mathbf{Rel}(A,B)$ is the same as the ordering $\sqsubseteq$ of subobjects of $A \times B$. Note that $[f,g] \sqsubseteq [f',g']$ if and only if there exist an epimorphism e and a morphism s such that $ef = sf'$ and $eg = sg'$.

Let $f : X \to Y$ be a morphism of $\mathbf{E}$ and $xf = yf$ a pullback. Then $f^\sharp f = [f,f] \sqsubseteq [\mathrm{id}_Y, \mathrm{id}_Y] = \mathrm{id}_Y$ since $f \top f = f(\mathrm{id}_Y \top \mathrm{id}_Y)$, and $\mathrm{id}_X = [\mathrm{id}_X, \mathrm{id}_X] \sqsubseteq [x,y] = [\mathrm{id}_Y, f][\mathrm{id}_Y, f] = ff^\sharp$ since there exists a unique morphism z such that $\mathrm{id}_X \top \mathrm{id}_X = z(x \top y)$. We recall that a relation $\alpha : X \multimap Y$ satisfies $\alpha^\sharp\alpha \sqsubseteq \mathrm{id}_Y$ and $\mathrm{id}_X \sqsubseteq \alpha\alpha^\sharp$ if and only if there exists a unique morphism $f : X \to Y$ such that $\alpha = [\mathrm{id}_X, f]$.

The terminal object of $\mathbf{E}$ will be denoted by 1 and its initial object by 0. The maximum relation $\Theta_{XY} : X \multimap Y$ is $[\mathrm{id}_{X\times Y}]$ and the minimum relation $0_{XY} : X \multimap Y$ is $[¡_{X\times Y}](= [¡_X, ¡_Y])$, where $¡_X : 0 \to X$ is a unique morphism from initial object 0. In particular we write $\Omega_X = \Theta_{X1}$ (i.e. $\Omega_X = [\mathrm{id}_X, !_X](= !_X)$, where $!_X : X \to 1$

is unique morphism into terminal object 1).

We now state five fundamental properties of relations in an elementary topos without proof:

[**I-category**] Let $\alpha, \alpha' : A \multimap B$, $\beta, \beta' : B \multimap C$ and $\gamma : C \multimap D$ be relations. Then

(a) $(\alpha\beta)\gamma = \alpha(\beta\gamma)$ (associative),

(b) $\mathrm{id}_A\alpha = \alpha\mathrm{id}_B = \alpha$ (identity),

(c) $\alpha^{\sharp\sharp} = \alpha$, $(\alpha\beta)^\sharp = \beta^\sharp\alpha^\sharp$ (involutive),

(d) If $\alpha \sqsubseteq \alpha'$ and $\beta \sqsubseteq \beta'$, then $\alpha\beta \sqsubseteq \alpha'\beta'$ and $\alpha^\sharp \sqsubseteq \alpha'^\sharp$ (monotone).

[**Heyting Algebra**] $\mathbf{Rel}(A, B)$ is a Heyting algebra for all objects A and B. That is, it is a lattice with the minimum element $0_{A,B}$, the maximum element $\Theta_{A,B}$, and pseudo-complements.

The infimum and the supremum of relations $\alpha, \beta : X \multimap Y$ are written as $\alpha \sqcap \beta$ and $\alpha \sqcup \beta$, respectively. Also the pseudo-complement of $\alpha : X \multimap Y$ relative to $\beta : X \multimap Y$ is denoted by $\alpha \Rightarrow \beta$.

[**Law of Puppe-Calenko**] If $\alpha : A \multimap B, \beta : B \multimap C$ and $\gamma : A \multimap C$ are relations, then $\alpha\beta \sqcap \gamma \sqsubseteq \alpha(\beta \sqcap \alpha^\sharp\gamma)$ is valid.

[**Rationality**] For each relation $\alpha : A \multimap B$ there exists a pair of morphisms $f : X \to A$ and $g : X \to B$ such that $\alpha = f^\sharp g$ and $ff^\sharp \sqcap gg^\sharp = \mathrm{id}_X$.

[**Distributive Law**] Let Λ be a set. If $\mathbf{E}$ has coproducts of all Λ-indexed families of objects, then the distributive law $\alpha(\sqcup_{\lambda\in\Lambda}\beta_\lambda) = \sqcup_{\lambda\in\Lambda}\alpha\beta_\lambda$ holds for relations $\alpha : A \multimap B$ and $\beta_\lambda : B \multimap C$ $(\lambda \in \Lambda)$.

The following is an elementary result of relations deduced from the last fundamental properties.

Lemma 2.2 *Let $\alpha : X \multimap Y$, $\beta : X \multimap Z$ be relations in* **E** *and V,W objects of* **E** . *Then*

(a) $\alpha 0_{YV} = 0_{XV}$ *and* $0_{WX}\alpha = 0_{WY}$.

(b) $\Omega_X \Omega_Y^\sharp = \Theta_{XY}$.

(c) $\alpha\Omega_Y \sqsubseteq \beta\Omega_Z$ *if and only if* $\alpha \sqsubseteq \beta\beta^\sharp\alpha$.

(d) $\alpha \sqsubseteq \alpha\alpha^\sharp\alpha$.

(e) *If* $\alpha^\sharp\beta = 0_{YZ}$ *and* $\alpha\Omega_Y \sqsubseteq \beta\Omega_Z$, *then* $\alpha = 0_{XY}$.

(f) *If* $\alpha\Omega_Y = 0_{X1}$, *then* $\alpha = 0_{XY}$.

Proof. (a) Let $\alpha = [f,g]$. Since **E** is a cartesian closed category with finite limits, the square

$$\begin{array}{ccc} 0 & \xrightarrow{¡} & \cdot \\ {\scriptstyle \mathrm{id}_0}\downarrow & & \downarrow{\scriptstyle g} \\ 0 & \xrightarrow[¡_Y]{} & Y \end{array}$$

is a pullback. Then $\alpha 0_{YV} = [¡f, \mathrm{id}_0¡] = [¡_X, ¡_V] = 0_{XV}$.

(b) Let $p : X \times Y \to X$ and $q : X \times Y \to Y$ be projections. Then the square $p!_X = q!_Y$ is a pullback and so $\Omega_X\Omega_Y^\sharp = [p,q] = [\mathrm{id}_{X\times Y}] = \Theta_{XY}$.

(c) Firstly assume $\alpha \sqsubseteq \beta\beta^\sharp\alpha$. Then $\alpha\Omega_Y \sqsubseteq \beta\beta^\sharp\alpha\Omega_Y \sqsubseteq \beta\Omega_Z$. Secondly assume $\alpha\Omega_Y \sqsubseteq \beta\Omega_Z$. Then we have $\alpha = \alpha \sqcap \alpha\Theta_{YY} = \alpha \sqcap \alpha\Omega_X\Omega_Y^\sharp$ (by (b)) $\sqsubseteq \alpha \sqcap \beta\Omega_Z\Omega_Y^\sharp = \alpha \sqcap \beta\Theta_{ZY}$ (by (b)) $\sqsubseteq \beta(\beta^\sharp\alpha \sqcap \Theta_{ZY})$ (by Law of Puppe-Calenko) $= \beta\beta^\sharp\alpha$.

(d) is immediate from (c) since $\alpha\Omega_Y = \alpha\Omega_Y$.

(e) It follows from (c) that $\alpha \sqsubseteq \beta\beta^\sharp\alpha$. Thus the result is obvious.

(f) is a particular case of (b) when $\beta = 0_{XY}$. ■

A relation $f : X \multimap Y$ in **E** is called a *partial function* if it satisfies $f^\sharp f \sqsubseteq \mathrm{id}_Y$.

Proposition 2.3 *Let $f,g : X \to Y$ be partial functions in* **E** . *Then*

(a) $ff^\sharp f = f$.

(b) *If* $f \sqsubseteq g$ *and* $f\Omega_Y = g\Omega_Y$, *then* $f = g$.

Proof. (a) follows from $f \sqsubseteq ff^\sharp f$ (by 2.2(d)) $\sqsubseteq f$ since $f^\sharp f \sqsubseteq \mathrm{id}_Y$.

(b) Assume $f \sqsubseteq g$ and $f\Omega_Y = g\Omega_Y$. Then we have $g \sqsubseteq ff^\sharp g$ (by 2.2(c)) $\sqsubseteq fg^\sharp g$ (by $f \sqsubseteq g$) $\sqsubseteq f$ (by $g^\sharp g \sqsubseteq \mathrm{id}_Y$). ■

Let X be an object in $\mathbf{E}$. A relation $p : X \rightharpoonup X$ is called a *guard function* on X if it satisfies $p \sqsubseteq \mathrm{id}_X$. Let $\mathbf{G}(X)$ be the set of all guard functions of X, that is, $\mathbf{G}(X) = \{p : X \rightharpoonup X \mid p \sqsubseteq \mathrm{id}_X\}$. It is clear that $\mathbf{G}(X)$ is a Heyting algebra as a subalgebra of $\mathbf{Rel}(X, X)$.

Proposition 2.4 *Let $p, q : X \rightharpoonup X$ be guard functions on an object X of $\mathbf{E}$. Then*

(a) *$pp = p$, $pq = qp = p \sqcap q$ and $p^\sharp = p$.*

(b) *If $p\Omega_X \sqsubseteq q\Omega_X$, then $p \sqsubseteq q$.*

(c) *If $p\Omega_X = q\Omega_X$, then $p = q$.*

(d) *$p\Omega_X \sqcap q\Omega_X = (p \sqcap q)\Omega_X$.*

Proof. (a) simply follows from the following short computations: $pp \sqsubseteq p \sqsubseteq pp^\sharp p \sqsubseteq pp$, $pq \sqsubseteq p \sqcap q = (p \sqcap q)(p \sqcap q) \sqsubseteq pq$, and $p \sqsubseteq pp^\sharp p \sqsubseteq p^\sharp$.

(b) Assume $p\Omega_X \sqsubseteq q\Omega_X$. First note that a guard function is a partial function and $p \sqcup q$ is also a guard function. We have $(p \sqcup q)\Omega_X = p\Omega_X \sqcup q\Omega_X$ (by Distributive Law) $= q\Omega_X$ and hence $q = p \sqcup q$ follows from 2.3(b).

(c) is a corollary of (b).

(d) follows from $(p \sqcap q)\Omega_X \sqsubseteq p\Omega_X \sqcap q\Omega_X \sqsubseteq p(\Omega_X \sqcap p^\sharp q\Omega_X)$ (by Law of Puppe-Calenko) $= pp^\sharp q\Omega_X = pq\Omega_X = (p \sqcap q)\Omega_X$. ■

Throughout this paper the negation operator will be used only for guard functions. That is, for a guard function q on Y, $\neg q$ denotes the negation of q in $\mathbf{G}(Y)$, i.e. $\neg q = (q \Rightarrow 0_{YY}) \sqcap \mathrm{id}_Y$. For example, $\neg \mathrm{id}_Y = 0_{YY}$ and $\neg 0_{YY} = \mathrm{id}_Y$.

In the rest of the paper we will assume that all of the morphisms, relations, partial functions and guard functions are those in a fixed topos $\mathbf{E}$.

Lemma 2.5 *Let $\alpha : X \rightharpoonup Y$ be a relation and $q : Y \to Y$ a guard function. Then $\alpha q = 0_{XY}$ if and only if $\alpha(\neg q) = \alpha$.*

Proof. Assume $\alpha q = 0$. Then we have $\alpha^\sharp \alpha \sqcap \mathrm{id}_Y \sqsubseteq \neg q$ from $(\alpha^\sharp \alpha \sqcap \mathrm{id}_Y) \sqcap q \sqsubseteq \alpha^\sharp \alpha q = 0$. Hence $\alpha = \alpha \sqcap \alpha \sqsubseteq \alpha(\alpha^\sharp \alpha \sqcap \mathrm{id}_Y) \sqsubseteq \alpha(\neg q) \sqsubseteq \alpha$ and so $\alpha = \alpha(\neg q)$. Conversely assume $\alpha = \alpha(\neg q)$. Then it is immediate that $\alpha q = \alpha(\neg q) q = 0$. ■

For every relation $\alpha : X \rightharpoonup Y$, there exists a monomorphism $i : D \to X$ in $\mathbf{E}$ such that $i^\sharp \Omega_D = \alpha \Omega_Y$ $(= i^\sharp i \Omega_X)$ from Rationality. Such a monomorphism i is called a *domain monomorphism* of α. It is trivial that $i^\sharp i$ is a guard function on X, that is, $i^\sharp i \sqsubseteq \mathrm{id}_X$. We define the kernel $k(\alpha)$ of α to be a guard function $k(\alpha) = \neg(i^\sharp i)$ on X and the domain $d(\alpha)$ of α to be a guard function $d(\alpha) = \neg k(\alpha)$ on X. Note that the definitions of kernels and domains do not depend on the choice of domain monomorphisms, that is, $i^\sharp i = \alpha \alpha^\sharp \sqcap \mathrm{id}_X$.

Proposition 2.6 *Let $\alpha, \beta : X \rightharpoonup Y$ be relations. Then*

(a) $d(\alpha) = \neg k(\alpha)$, $k(\alpha) = \neg d(\alpha)$ *and* $k(\alpha) \sqcap d(\alpha) = 0_{XX}$.

(b) $k(\mathrm{id}_X) = 0_{XX}$, $d(\mathrm{id}_X) = \mathrm{id}_X$, $k(0_{XY}) = \mathrm{id}_X$ *and* $d(0_{XY}) = 0_{XX}$.

(c) *If p is a guard function on X, then $k(p) = \neg p$ and $d(p) = \neg\neg p$.*

(d) *If $f : X \to Y$ is a morphism in $\mathbf{E}$, then $k(f) = 0_{XX}$ and $d(f) = \mathrm{id}_X$.*

(e) *If $\alpha \sqsubseteq \beta$, then $k(\beta) \sqsubseteq k(\alpha)$ and $d(\alpha) \sqsubseteq d(\beta)$.*

Proof. The statements (a) - (d) are trivial.

(e) Let i and j be domain monomorphisms of α and β, respectively. Then $k(\alpha) = \neg(i^\sharp i)$ and $k(\beta) = \neg(j^\sharp j)$. The statement follows from $\alpha \Omega_Y \sqsubseteq \beta \Omega_Y \Rightarrow i^\sharp i \Omega_Y \sqsubseteq j^\sharp j \Omega_Y \Rightarrow i^\sharp i \sqsubseteq j^\sharp j$ (by 2.4(c)) $\Rightarrow k(\beta) \sqsubseteq k(\alpha) \Rightarrow d(\alpha) \sqsubseteq d(\beta)$. ■

Note that $k(\alpha) \sqcup d(\alpha) \neq \mathrm{id}_X$ in general, because of the basic properties of Heyting algebras.

Lemma 2.7 *Let $\alpha : X \rightharpoonup Y$ be a relation and $p : X \rightharpoonup X$ a guard function. Then the following statements are equivalent:*

(a) $p\alpha = 0_{XY}$.

(b) $(\neg p)\alpha = \alpha$.

(c) $p \sqsubseteq k(\alpha)$.

(d) $\neg\neg p \sqsubseteq k(\alpha)$.

(e) $(\neg\neg p)\alpha = 0_{XY}$.

Proof. Let i be a domain monomorphism of α. Then $\alpha\Omega_Y = i^\sharp i\Omega_X$. The proof follows from the following equivalences: $\alpha = (\neg p)\alpha \Leftrightarrow \alpha^\sharp = \alpha^\sharp(\neg p)$ (by applying $\sharp$ and 2.4(a)) $\Leftrightarrow \alpha^\sharp p = 0$ (by 2.5) $\Leftrightarrow$ (a) $p\alpha = 0 \Leftrightarrow pi^\sharp i\Omega_X = p\alpha\Omega_Y = 0$ (by 2.2(c)) $\Leftrightarrow p \sqcap i^\sharp i = pi^\sharp i = 0 \Leftrightarrow$ (c) $p \sqsubseteq \neg(i^\sharp i) = k(\alpha) \Leftrightarrow$ (d) $\neg\neg p \sqsubseteq \neg(i^\sharp i) = k(\alpha)$ (by $p \sqsubseteq \neg\neg p$) $\Leftrightarrow$ (e) $(\neg\neg p)\alpha = 0$. ■

Theorem 2.8 *Let $\alpha : X \rightharpoonup Y$ be a relation. Then*

(a) *$k(\alpha)\alpha = 0_{XY}$, that is, $k(\alpha)$ is the maximum element of a set $\{p \in G(X) \mid p\alpha = 0_{XY}\}$.*

(b) *$d(\alpha)\alpha = \alpha$.*

(c) *$k(\alpha) = \mathrm{id}_X$ ($ord(\alpha) = 0_{XX}$) if and only if $\alpha = 0$.*

(d) *For every relation $\tau : W \rightharpoonup X$, $\tau\alpha = 0_{WY}$ if and only if $\tau k(\alpha) = \tau$.*

Proof. (a) and (b) easily follow from 2.7.

(c) From 2.6(b) it suffices to show that $d(\alpha) = 0$ implies $\alpha = 0$. But if $d(\alpha) = 0$ then $\alpha = d(\alpha)\alpha$ (by (b)) $= 0$.

(d) Let i be a domain monomorphism of α. Then $k(\alpha) = \neg(i^\sharp i)$. Hence $\tau k(\alpha) = \tau \Leftrightarrow \tau(i^\sharp i) = 0$ (by 2.5 and $k(\alpha) = \neg(i^\sharp i)$) $\Leftrightarrow \tau\alpha\Omega_Y = \tau(i^\sharp i)\Omega_X = 0 \Leftrightarrow \tau\alpha = 0$. ■

The following corollary is a summary in order to interpret the program specification $\{p\}f\{q\}$ and define weakest and weakest liberal preconditions [11] in the subsequent sections.

Corollary 2.9 *Let $\alpha : X \rightharpoonup Y$ be a relation and $p : X \rightharpoonup X$, $q : Y \rightharpoonup Y$ guard functions. Then the following statements are equivalent:*

(a) *$p\alpha(\neg q) = 0_{XY}$.*

(b) *$(\neg p)\alpha(\neg q) = \alpha(\neg q)$.*

(c) $p\alpha(\neg\neg q) = p\alpha$.

(d) $(\neg\neg p)\alpha(\neg q) = 0_{XY}$.

(e) $(\neg\neg p)\alpha(\neg\neg q) = (\neg\neg p)\alpha$.

(f) $p \sqsubseteq k(\alpha(\neg q))$.

Proof. From 2.7 it is easy to see (a) $\Leftrightarrow$ (b) $\Leftrightarrow$ (d) $\Leftrightarrow$ (f), and 2.5 implies (a) $\Leftrightarrow$ (c) and (d) $\Leftrightarrow$ (e). ■

3 Assertion Semantics

Assertion (axiomatic) semantics is a methodology based on the use of assertions as preconditions and postconditions in program specifications associated with R. Floyd, C.A.R. Hoare, and E. Dijkstra. An assertion in this semantics is a statement about the program state which is either true or false. For a program α and assertions p, q, we write a program with assertions as $\{p\}\alpha\{q\}$ following to Hoare [6]. The program specification $\{p\}\alpha\{q\}$ is correct if the satisfaction of precondition p about the input data guarantees the truth of postcondition q.

This section is devoted to a categorical interpretation of assertion (axiomatic) semantics of programming language in particular categories, namely toposes. All of the preconditions, postconditions and programs are interpreted as (binary) relations in toposes by making use of relational calculus, and several fundamental properties of Dijkstra's weakest preconditions are proved. Assertions in the semantics depend on the intuitionistic (or constructive) logic, so our approach will be meaningful to the mathematical foundations of semantics.

We begin with our interpretation of Hoare's program specification and meanings of control statements in the framework of relational calculus.

Example 3.1 *We show a simple example of assertion semantics using relations in* **Set**. *Consider an assertion*

$$\{a^n = x * a^i \wedge i \geq 0\}\textbf{while } (i \neq 0)\textbf{do}(x := x * a; i := i - 1)\{a^n = x * a^i \wedge i = 0\},$$

and a state space $X = \mathbf{Z}^3$, *where* $\mathbf{Z}$ *is a set of integers. Logical formulae* $a^n = x * a^i \wedge i \geq 0$ *and* $a^n = x * a^i \wedge i = 0$ *are interpreted as the subsets*

$$P = \{(n,x,i) \in \mathbf{Z}^3 | a^n = x * a^i \wedge i \geq 0\} \text{ and } Q = \{(n,x,i) \in \mathbf{Z}^3 | a^n = x * a^i \wedge i = 0\}$$

of X, *respectively. The subsets* P *and* Q *correspond to guard functions*

$$p = \{(x,x) \in X \times X | x \in P\} : X \to X \text{ and } q = \{(x,x) \in X \times X | x \in Q\} : X \to X,$$

respectively. Then $\neg q$ *is expressed by a relation* $\{(x,x) \in X \times X | x \notin Q\}$. *A program* **while** $(i \neq 0)$ **do** $(x := x * a; i := i - 1)$ *is interpreted as a partial function (relation)* $f : X \to X$, *which is undefined on the states from which the program does not halt. In general Hoare's assertion* $\{p\}f\{q\}$ *is true if and only if the image* $f(P)$ *is included in* Q. *It is easy to ascertain that the assertion* $\{p\}f\{q\}$ *is equivalent to a relational equation* $pf(\neg q) = 0_{XX}$.

The last example leads to the following definition of assertion semantics in toposes:

Definition 3.2 *Let* $\alpha : X \rightharpoonup Y$ *be a relation, and* $p : X \rightharpoonup X$, $q : Y \rightharpoonup Y$ *guard functions. We write* $\{p\}\alpha\{q\}$ *if and only if one of the equivalent conditions in 2.9 holds.*

For example, $\{p\}\mathrm{id}_X\{q\}$ if and only if $p \sqcap \neg q = 0_{XX}$ for $p, q \in \mathbf{G}(X)$.

Assume that basic statements of programs are interpreted by relations between state objects. Define the meanings of control statements with relations as follows:

(a) $(\alpha;\beta) = \alpha \cdot \beta$.

(b) $(\mathbf{if}\ q\ \mathbf{then}\ \alpha\ \mathbf{else}\ \beta) = q\alpha \sqcup (\neg q)\beta$.

(c) $(\mathbf{while}\ p\ \mathbf{do}\ \alpha) = \bigsqcup_{n=0}^{\infty}(p\alpha)^n(\neg p)$.

For the above interpretation (c) we have to assume that $\mathbf{E}$ has countable coproducts. In this interpretation it is clear that $(\mathbf{if}\ q\ \mathbf{then}\ \mathrm{id}_Y\ \mathbf{else}\ \mathrm{id}_Y) = \mathrm{id}_Y$ does

not always hold. This is a one of the essential differences between the approach treating semantics with classical logic and our topos theoretic approach.

The following proof rules from 3.3 to 3.6 will be obtained by 2.9:

Proposition 3.3 (Consequence Rule) *If* $p \sqsubseteq p_1$, $q_1 \sqsubseteq q$ *and* $\{p_1\}\alpha\{q_1\}$, *then* $\{p\}\alpha\{q\}$.

Proof. We have $p \sqsubseteq p_1$, $\neg q \sqsubseteq \neg q_1$ and $p_1\alpha(\neg q_1) = 0$. Thus, $p\alpha(\neg q) \sqsubseteq p_1\alpha(\neg q_1) = 0$. ■

Proposition 3.4 (Composition Rule) *If* $\{p\}\alpha\{q\}$ *and* $\{q\}\beta\{r\}$, *then* $\{p\}(\alpha;\beta)\{r\}$.

Proof. We have $p\alpha = p\alpha(\neg\neg q)$ and $(\neg\neg q)\beta(\neg r) = 0$ by 2.9. Hence, $p\alpha\beta(\neg r) = p\alpha(\neg\neg q)\beta(\neg r) = 0$. ■

Proposition 3.5 (Conditional Rule) *If* $\{p \sqcap q\}\alpha\{r\}$ *and* $\{p \sqcap \neg q\}\beta\{r\}$, *then* $\{p\}$ (**if** q **then** α **else** β) $\{r\}$.

Proof. We have $(p \sqcap q)\alpha(\neg r) = 0$ and $(p \sqcap \neg q)\beta(\neg r) = 0$. Thus,

$$\begin{aligned} p(q\alpha \sqcup (\neg q)\beta)(\neg r) &= pq\alpha(\neg r) \sqcup p(\neg q)\beta(\neg r) \qquad \text{(by the distributive law)} \\ &= (p \sqcap q)\alpha(\neg r) \sqcup (p \sqcap \neg q)\beta(\neg r) \\ &= 0 \sqcup 0 \\ &= 0. \end{aligned}$$

■

Proposition 3.6 (Iteration Rule) *If* $\{p \sqcap q\}\alpha\{q\}$, *then* $\{q\}$ (**while** p **do** α) $\{(\neg p) \sqcap q\}$.

Proof. We have $pq\alpha(\neg q) = 0$ and $(\neg\neg q)p\alpha(\neg\neg q) = (\neg\neg q)p\alpha$. First we see by induction on n that $q(p\alpha)^n(\neg\neg q) = q(p\alpha)^n$ for $n \geq 0$. If $n = 0$, then $q(\neg\neg q) = q$ since $q \sqsubseteq \neg\neg q$. Assume $q(p\alpha)^n(\neg\neg q) = q(p\alpha)^n$ for $n \geq 0$. Then $q(p\alpha)^{n+1}(\neg\neg q) = q(p\alpha)^n p\alpha(\neg\neg q) = q(p\alpha)^n(\neg\neg q)p\alpha(\neg\neg q) = q(p\alpha)^n(\neg\neg q)p\alpha = q(p\alpha)^n p\alpha = q(p\alpha)^{n+1}$.

Hence $q(p\alpha)^n(\neg q) = 0 \quad (n \geq 0)$ by 2.9. On the other hand, $\neg p \sqcap \neg(\neg p \sqcap q) \sqsubseteq \neg q$ since $(\neg p \sqcap q) \sqcap \neg(\neg p \sqcap q) = 0$. Therefore,

$$q\{\bigsqcup_{n=0}^{\infty}(p\alpha)^n(\neg p)\}\{\neg(\neg p \sqcap q)\} = \bigsqcup_{n=0}^{\infty}\{q(p\alpha)^n(\neg p)\}\{\neg(\neg p \sqcap q)\} \sqsubseteq \bigsqcup_{n=0}^{\infty} q(p\alpha)^n(\neg q) = 0.$$

■

The following examples (Cf. [11]) are verified by simple computations of relations:

Example 3.7 *If $\alpha : X \rightharpoonup Y$ is a relation and $p : X \rightharpoonup X$ is a guard function, then* **while** p **do** α = **while** p **do** (**while** p **do** α).

Example 3.8 *If $\alpha : X \rightharpoonup X$, $\beta : X \rightharpoonup Y$ are relations and $p : X \rightharpoonup X$ is a guard function, then* (**while** p **do** $\alpha)\beta$ = **if** p **then** (**while** p **do** $\alpha)\beta$ **else** β.

Example 3.9 *Let $\alpha : X \rightharpoonup X$, $\gamma, \delta : X \rightharpoonup Y$ and $\beta, \xi : Y \rightharpoonup Y$ be relations, and $p \in \mathbf{G}(X)$, $q \in \mathbf{G}(Y)$. If $p\delta = \delta q$, $(\neg p)\delta = \delta(\neg q)$ and* (**if** p **then** $\alpha\delta$ **else** γ) = δ(**if** q **then** β **else** ξ), *then* (**while** p **do** $\alpha)\gamma = \delta$(**while** q **do** β)ξ.

4 Weakest Preconditions

In this section we discuss on the weakest conditions [11, 12, 13] in our framework and prove the partial correctness theorem. At the end of the section we briefly state a relationship between Wagner's approach [13] and ours.

The weakest liberal precondition wlp(α, q) for a relation $\alpha : X \rightharpoonup Y$ and a guard function $q \in \mathbf{G}(Y)$ is satisfied by any initial state with the property that, if execution of α terminates, then the program state upon termination satisfies q. Thus wlp(α, q) also holds for all initial states from which α does not terminate. The weakest precondition wp(α, q) strengthens the liberal precondition wlp(α, q) by guaranteeing that computation of α will terminate. We are now ready to define the weakest conditions in toposes.

Definition 4.1 *Let $\alpha : X \multimap Y$ be a relation and $q : Y \multimap Y$ a guard function. The weakest liberal precondition is defined by* $\mathrm{wlp}(\alpha, q) = k(\alpha(\neg q))$, *and the weakest precondition by* $\mathrm{wp}(\alpha, q) = \mathrm{wlp}(\alpha, q) \sqcap d(\alpha)$.

The following proposition is easily obtained from the above definitions:

Proposition 4.2 *Let $\alpha : X \multimap Y$ be a relation and $q : Y \multimap Y$ a guard function. Then*

(a) $\mathrm{wlp}(\alpha, \mathrm{id}_Y) = \mathrm{id}_X$ *and* $\mathrm{wp}(\alpha, \mathrm{id}_Y) = d(\alpha)$.

(b) $\mathrm{wlp}(\alpha, 0_{YY}) = k(\alpha)$ *and* $\mathrm{wp}(\alpha, 0_{YY}) = 0_{XX}$

(c) $\mathrm{wlp}(\mathrm{id}_X, q) = \neg\neg q$ *and* $\mathrm{wp}(\mathrm{id}_X, q) = \neg\neg q$.

(d) $\mathrm{wlp}(0_{XY}, q) = \mathrm{id}_X$ *and* $\mathrm{wp}(0_{XY}, q) = 0_{XX}$.

(e) *If* $\alpha' \sqsubseteq \alpha$ *and* $q \sqsubseteq q'$, *then* $\mathrm{wlp}(\alpha, q) \sqsubseteq \mathrm{wlp}(\alpha', q')$.

(f) $\mathrm{wlp}(\alpha, q) = \mathrm{wlp}(\alpha, \neg\neg q)$ *and* $\mathrm{wp}(\alpha, q) = \mathrm{wp}(\alpha, \neg\neg q)$.

Proof. (a) $\mathrm{wlp}(\alpha, \mathrm{id}_X) = k(\alpha(\neg \mathrm{id}_Y)) = k(0_{XY}) = \mathrm{id}_X$ and $\mathrm{wp}(\alpha, \mathrm{id}_Y) = d(\alpha) \sqcap \mathrm{id}_X = d(\alpha)$.

(b) $\mathrm{wlp}(\alpha, 0) = k(\alpha(\neg 0)) = k(\alpha \mathrm{id}_Y) = k(\alpha)$ and $\mathrm{wp}(\alpha, 0) = d(\alpha) \sqcap \mathrm{wlp}(\alpha, 0) = d(\alpha) \sqcap k(\alpha) = 0$.

The other statements (c) - (f) are trivial. ■

Theorem 4.3 (Composition Rule for Partial Correctness) *If $\alpha : X \multimap Y$, $\beta : Y \multimap Z$ are relations and $r : Z \multimap Z$ is a guard function, then* $\mathrm{wlp}(\alpha, \mathrm{wlp}(\beta, r)) = \mathrm{wlp}(\alpha\beta, r)$.

Proof. Set $q' = \mathrm{wlp}(\beta, r)$, $p = \mathrm{wlp}(\alpha\beta, r)$ and $p' = \mathrm{wlp}(\alpha, \mathrm{wlp}(\beta, r)) = \mathrm{wlp}(\alpha, q')$. Then $\{p'\}\alpha\{q'\}$ and $\{q'\}\beta\{r\}$. Hence by the Composition Rule 3.4 we have $\{p'\}\alpha\beta\{r\}$ and so $p' \sqsubseteq \mathrm{wlp}(\alpha\beta, r)$. Also the converse relation $p' \sqsubseteq p$ comes from

$$
\begin{aligned}
& p = \mathrm{wlp}(\alpha\beta, r) \\
\Rightarrow\ & p\alpha\beta(\neg r) = 0 \\
\Leftrightarrow\ & p\alpha k(\beta(\neg r)) = p\alpha \qquad (2.8(\mathrm{d}))
\end{aligned}
$$

$$\Leftrightarrow\ p\alpha\mathrm{wlp}(\beta,r) = p\alpha$$

$$\Leftrightarrow\ p\alpha(\neg\neg q') = p\alpha \qquad (\text{by } q' = \mathrm{wlp}(\beta,r) = \neg(i'^{\sharp}i'))$$

$$\Leftrightarrow\ p\alpha(\neg q') = 0$$

$$\Leftrightarrow\ p \sqsubseteq \mathrm{wlp}(\alpha,q') = p'.$$

■

Note that the Composition Rule 3.4 is equivalent to $\mathrm{wlp}(\alpha,\mathrm{wlp}(\beta,r)) \sqsubseteq \mathrm{wlp}(\alpha\beta,r)$. The following propositions are also fundamental properties of the weakest liberal preconditions:

Proposition 4.4 *If $\alpha_1,\alpha_2 : X \rightharpoonup Y$ are relations and if $q : Y \rightharpoonup Y$ is a guard function, then* $\mathrm{wlp}(\alpha_1 \sqcup \alpha_2, q) = \mathrm{wlp}(\alpha_1,q) \sqcap \mathrm{wlp}(\alpha_2,q)$.

Proof. It is clear that $\mathrm{wlp}(\alpha_1 \sqcup \alpha_2,q) \sqsubseteq \mathrm{wlp}(\alpha_1,q) \sqcap \mathrm{wlp}(\alpha_2,q)$ by 4.2(e). Set $p = \mathrm{wlp}(\alpha_1,q) \sqcap \mathrm{wlp}(\alpha_2,q)$. Then $p \sqsubseteq \mathrm{wlp}(\alpha_i,q)$ and so $p\alpha_i(\neg\neg q) = p\alpha_i$ $(i = 0,1)$ by 2.9. Hence we have $p(\alpha_1 \sqcup \alpha_2)(\neg\neg q) = p(\alpha_1 \sqcup \alpha_2)$, which indicates $p \sqsubseteq \mathrm{wlp}(\alpha_1 \sqcup \alpha_2,q)$.

■

Proposition 4.5 *Let $\alpha : X \rightharpoonup Y$ be a relation and $q_1,q_2 : Y \rightharpoonup Y$ guard functions. If $q_1 = \neg\neg q_1$ and $q_2 = \neg\neg q_2$, then* $\mathrm{wlp}(\alpha, q_1 \sqcap q_2) = \mathrm{wlp}(\alpha,q_1) \sqcap \mathrm{wlp}(\alpha,q_2)$.

Proof. It is clear that $\mathrm{wlp}(\alpha,q_1 \sqcap q_2) \sqsubseteq \mathrm{wlp}(\alpha,q_1) \sqcap \mathrm{wlp}(\alpha,q_2)$. Set $p = \mathrm{wlp}(\alpha,q_1) \sqcap \mathrm{wlp}(\alpha,q_2)$. Then $p \sqsubseteq \mathrm{wlp}(\alpha,q_i)$ and so $p\alpha q_i = p\alpha$ $(i = 0,1)$ by $q_i = \neg\neg q_i$ and 2.9. Hence we have $p\alpha(q_1 \sqcap q_2) = p\alpha q_1 q_2 = p\alpha q_2 = p\alpha$, which shows $p \sqsubseteq \mathrm{wlp}(\alpha, q_1 \sqcap q_2)$.

■

Proposition 4.6 *If $\mathbf{G}(X)$ is a boolean algebra, that is, $\neg\neg p = p$ for all $p \in \mathbf{G}(X)$) and $f : X \to Y$ is a partial function, then* $\mathrm{wlp}(f, q_1 \sqcup q_2) = \mathrm{wlp}(f,q_1) \sqcup \mathrm{wlp}(f,q_2)$ *for all guard functions $q_1,q_2 \in \mathbf{G}(Y)$.*

Proof. Let i_1, i_2 and i be domain monomorphisms of $f(\neg q_1)$, $f(\neg q_2)$ and $f(\neg(q_1 \sqcup q_2))$, respectively. Then $k(f(\neg q_1)) = \neg(i_1^{\sharp} i_1)$, $k(f(\neg q_2)) = \neg(i_2^{\sharp} i_2)$ and $k(f(\neg(q_1 \sqcup$

$q_2))) = \neg(i^\sharp i)$. Hence,

$$\begin{aligned} i^\sharp i\Omega_X &= f(\neg(q_1 \sqcup q_2))\Omega_Y \\ &= f(\neg q_1 \sqcap \neg q_2)\Omega_Y \\ &= f(\neg q_1)\Omega_Y \sqcap f(\neg q_2)\Omega_Y \qquad (f \text{ is a partial function}) \\ &= i_1^\sharp i_1\Omega_X \sqcap i_2^\sharp i_2\Omega_X \\ &= (i_1^\sharp i_1 \sqcap i_2^\sharp i_2)\Omega_X \qquad (2.4(\mathrm{d})). \end{aligned}$$

So we have $i^\sharp i = i_1^\sharp i_1 \sqcap i_2^\sharp i_2$ and consequently $\mathrm{wlp}(f, q_1 \sqcup q_2) = \mathrm{wlp}(f, q_1) \sqcup \mathrm{wlp}(f, q_2)$ by the assumption. ■

Corollary 4.7 *If* $\mathbf{G}(X)$, $\mathbf{G}(Y)$ *are boolean algebras, then the following statements are equivalent:*

(a) $\mathrm{wlp}(\alpha, q_1 \sqcup q_2) = \mathrm{wlp}(\alpha, q_1) \sqcup \mathrm{wlp}(\alpha, q_2)$ *for all* $q_1, q_2 \in \mathbf{G}(X)$,

(b) $\alpha(q_1 \sqcap q_2)\Omega_Y = \alpha q_1\Omega_Y \sqcap \alpha q_2\Omega_Y$ *for all* $q_1, q_2 \in \mathbf{G}(Y)$.

Proof. The proof is analogous to the last proposition. ■

Example 4.8 *Consider the category* **Graph***of directed graphs. It is well-known* [5][9] *that* **Graph** *is a topos. Let* X *be a directed graph with two vertices* x_0, x_1 *and an edge* e, *and* Y *a directed graph with two vertices* y_0, y_1 *and two edges* u, v:

$$X : x_0 \xrightarrow{e} x_1, \qquad Y : y_0 \underset{u}{\overset{v}{\rightrightarrows}} y_1.$$

Define graph morphisms $f, g : X \to Y$ *by* $f(x_0) = g(x_0) = y_0$, $f(x_1) = g(x_1) = y_1$, $f(e) = u$ *and* $g(e) = v$. *The set of all subgraphs (guard functions) of* Y *is*

$$\{0\ (= \emptyset), \{y_0\}, \{y_1\}, \{y_0, y_1\}, \{y_0 \xrightarrow{u} y_1\}, \{y_0 \xrightarrow{v} y_1\}, Y\}.$$

It is easy to see that if $q \in \{\{y_0, y_1\}, \{y_0 \xrightarrow{u} y_1\}, \{y_0 \xrightarrow{v} y_1\}, Y\}$ *then* $\neg q = 0$. *Hence*

we have

$$\mathrm{wlp}(f,q) = \begin{cases} \mathrm{id}_X & \text{if } \neg q = 0 \\ \{x_0\} & \text{if } q = \{y_0\} \\ \{x_1\} & \text{if } q = \{y_1\} \\ 0 & \text{if } q = 0 \end{cases}$$

and so $\mathrm{wlp}(f,q) = \mathrm{wlp}(g,q)$ *for all* $q \in \mathbf{G}(Y)$.

The last example is a weak counterexample to FACT 3.7 of E. Wagner [13] in our sense, because $\{\neg q \mid q \in \mathbf{G}(Y)\} \neq \mathbf{G}(Y)$ in general. The final theorem in this section relates to Wagner's approach.

Theorem 4.9 *Let* $\alpha : X \multimap Y$ *be a relation and* $q : Y \multimap Y$ *a guard function. If* $i : P \to X$ *and* $j : Q \to Y$ *are domain monomorphisms of* $\mathrm{wlp}(\alpha, q)$ *and* $\neg\neg q$*, respectively, then the diagram*

$$\begin{array}{ccc} P & \xrightarrow{\ i\ } & X \\ {\scriptstyle\beta}\downarrow & & \downarrow{\scriptstyle\alpha} \\ Q & \xrightarrow[\ j\ ]{} & Y \end{array}$$

is a pullback in the category of relations, where $\beta = i\alpha j^\sharp$.

Proof. We first show $i\alpha = \beta j$. But $i\alpha = ii^\sharp i\alpha = ii^\sharp i\alpha(\neg\neg q) = ii^\sharp i\alpha j^\sharp j = i\alpha j^\sharp j = \beta j$. Next assume $\tau\alpha = \sigma j$ for two relations $\tau : Z \multimap X$ and $\sigma : Z \multimap Q$. Then we have $\tau\alpha(\neg q) = \sigma j(\neg q) = \sigma j j^\sharp j(\neg q) = \sigma j(\neg\neg q)(\neg q) = 0_{ZQ}$ and $\tau = k(\alpha(\neg q))$ (by 2.8(d)) $= \tau(i^\sharp i)$. Set $\delta = \tau i^\sharp$. Then $\delta i = \tau i^\sharp i = \tau$ and $\delta\beta = \tau i^\sharp\beta = \tau i^\sharp i\alpha j^\sharp = \tau\alpha j^\sharp = \sigma j j^\sharp = \sigma$ (by $jj^\sharp = id_Q$). If there is another relation $\delta' : Z \multimap P$ satisfying $\delta' i = \tau$ and $\delta'\beta = \sigma$, then $\delta' = \delta' i i^\sharp = \tau i^\sharp = \delta$. ∎

Note that the above diagram is also a pullback in the category of partial functions if α is a partial function.

5 Total Correctness

The aim of this section is to prove the composition law for the weakest preconditions (or total correctness). We begin with defining the totality of relations in toposes.

A relation $\alpha : X \multimap Y$ is *total* if $\tau\alpha = 0$ for any relation $\tau : U \multimap X$ implies $\tau = 0$.

The following lemma shows the fundamental properties on the totality:

Lemma 5.1 *Let $\alpha : X \multimap Y$ be a relation.*

(a) *α is total if and only if $d(\alpha) = \mathrm{id}_X$ (or $k(\alpha) = 0_{XX}$).*

(b) *If $\alpha\Omega_Y = \Omega_X$, then α is total.*

(c) *If $\mathbf{G}(X)$ is a boolean algebra, then α is total if and only if $\alpha\Omega_Y = \Omega_X$.*

Proof. (a) If $k(\alpha) = 0$ and $\tau\alpha = 0$ then $\tau = \tau k(\alpha)$ (by 2.8(d)) $= \tau 0 = 0$. Conversely, if α is total then $k(\alpha) = 0$ since $k(\alpha)\alpha = 0$ by 2.8(a).

(b) If $\alpha\Omega_Y = \Omega_X$ and $\tau\alpha = 0$ then $\tau\Omega_X = \tau\alpha\Omega_Y = 0$. Hence $\tau = 0$ by 2.2(c).

(c) It suffices to show that if $\mathbf{G}(X)$ is boolean and α is total then $\alpha\Omega_Y = \Omega_X$. From the totality of α we have $d(\alpha) = \mathrm{id}_X$ by (a). But, because $\mathbf{G}(X)$ is boolean, $i^\sharp i = \neg\neg(i^\sharp i) = d(\alpha) = \mathrm{id}_X$ for a domain monomorphism of α. Therefore $\alpha\Omega_Y = i^\sharp i\Omega_X = \Omega_X$. ■

The following theorem is essential for the proof of 5.3, called the composition rule for total correctness in assertion semantics:

Theorem 5.2 *If a partial function $f : X \multimap Y$ is total, then $k(fk(\beta)) = d(f\beta)$ for each relation $\beta : Y \multimap Z$.*

Proof. Set $p = k(fk(\beta)) \sqcap k(f\beta)$. Then $p \sqsubseteq k(fk(\beta))$ and $p \sqsubseteq k(f\beta)$. From the former we have $pfk(\beta) = 0$ (by 2.7), and from the latter $pf\beta = 0$. Hence $pf = pfk(\beta) = 0$ by 2.8(d). Since f is a total function, $p = 0$ and hence $k(fk(\beta)) \sqsubseteq d(f\beta)$.

Now we show the converse relation $d(f\beta) \sqsubseteq k(fk(\beta))$. Let i be a domain monomorphism of $f\beta$. Then $f\beta\Omega_Z = i^\sharp i\Omega_X$. Because $(i^\sharp i)fk(\beta)\Omega_Y \sqsubseteq i^\sharp i\Omega_X = f\beta\Omega_Z = fd(\beta)\beta\Omega_Z \sqsubseteq fd(\beta)\Omega_Y$ and $((i^\sharp i)fk(\beta))^\sharp fd(\beta) = k(\beta)^\sharp f^\sharp i^\sharp ifd(\beta) \sqsubseteq k(\beta)^\sharp d(\beta) = 0$, we have $i^\sharp ifk(\beta) = 0$ using 2.2(b). Hence $i^\sharp i \sqsubseteq k(fk(\beta))$ and $d(f\beta) \sqsubseteq k(fk(\beta))$, as desired. ■

Corollary 5.3 (Composition Rule for Total Correctness) *If a partial function $f : X \rightharpoonup Y$ is total then* $\mathrm{wp}(f\beta, q) = \mathrm{wp}(f, \mathrm{wp}(\beta, q))$ *for each relation $\beta : Y \rightharpoonup Z$ and each guard function $q : Z \rightharpoonup Z$.*

Proof.

$$\begin{aligned}
\mathrm{wp}(f\beta, q) &= d(f\beta) \sqcap \mathrm{wlp}(f\beta, q) \\
&= \mathrm{wlp}(f, d(\beta)) \sqcap \mathrm{wlp}(f, \mathrm{wlp}(\beta, q)) \qquad (\text{ by 4.3 and 5.2}) \\
&= \mathrm{wlp}(f, d(\beta) \sqcap \mathrm{wlp}(\beta, q)) \\
&= \mathrm{wp}(f, \mathrm{wp}(\beta, q))
\end{aligned}$$

■

Finally we prove that every partial function has a totalizer.

Definition 5.4 *Let $f : X \rightharpoonup Y$ be a partial function. A partial function $t : T \rightharpoonup X$ is a totalizer of f if it satisfies:*

(a) *tf is total,*

(b) *If $u : U \rightharpoonup X$ is such that uf is total, there exists unique partial function $s : U \rightharpoonup T$ with $u = st$.*

Corollary 5.5 *Let $f : X \rightharpoonup Y$ be a partial function. Every domain monomorphism $t : T \rightarrowtail X$ of $d(f) \in \mathbf{G}(X)$ is a totalizer of f. That is, all partial functions have totalizers.*

Proof. At first we show tf is total. Recall t is total by 5.1(b) and $d(f) = t^\sharp t$ since t is a domain monomorphism of $d(f)$. Then $tk(f) = tt^\sharp tk(f) = td(f)k(f) = 0$ and by 5.2 $d(tf) = k(tk(f)) = k(0) = \mathrm{id}_T$. Hence tf is total by 5.1(a). Next we assume uf is total for a partial function $u : U \rightharpoonup X$. If $u = st$ for some partial function $s : U \rightharpoonup T$, then $s = ut^\sharp$ by $tt^\sharp = \mathrm{id}_T$. Because $(ut^\sharp)^\sharp(ut^\sharp) = tu^\sharp ut^\sharp \sqsubseteq tt^\sharp = \mathrm{id}_T$, $ut^\sharp$ is a partial function. Thus we have to only see $ut^\sharp t = u$. On the other hand uf is total and so is u. Hence $k(uk(f)) = d(uf) = \mathrm{id}_U$ by 5.2 and 5.1(a). But we have

$ud(f) = ut^\sharp t = u \Leftrightarrow uk(f) = 0$ (by 2.5) $\Leftrightarrow k(uk(f)) = \mathrm{id}_U$ (by 2.8(c)). Therefore t is a totalizer of f. ■

References

[1] Calenko, M.S.: The structures of correspondence categories, *Soviet Math. Dokl.*, Vol. 18(1977),pp. 1498–1502.

[2] Calenko, M.S., Gisin, V.B. and Raikov, D.A.: Ordered categories with involution, *Dissertations Mathematicae*, Vol. 227(1984), Warszawa.

[3] Dipaola, R.A. and Heller, A.: Dominical categories: recursion theory without element, *J. Symbolic Logic*, Vol. 52, No. 3(1987), pp.594–635.

[4] Dijkstra, E.W.: *A discipline of programming*, Prentice-Hall, 1976.

[5] Goldblatt, R.: *Topoi, The categorial analysis of logic*, North-Holland, 1979.

[6] Hoare, C.A.R.: An axiomatic basis for computer programming, *Comm. ACM*, Vol. 12(1969).

[7] Johnstone, P.T.: *Topos theory*, Academic Press, 1977.

[8] Kawahara, Y.: Relations in categories with pullbacks, *Mem. Fac. Sci. Kyushu Univ. Ser.A*, Vol. 27(1973), pp.149–173.

[9] Kawahara, Y.: Pushout-complements and basic concepts of grammars in toposes, *Theor. Comput. Sci.*, Vol. 77(1990), pp.267–289.

[10] Lambek, J. and Scott, P.J.: *Introduction to higher order categorical logic*, Cambridge University Press, 1986.

[11] Manes, E.G. and Arbib, M.A.: *Algebraic approaches to program semantics*, Springer-Verlag, 1986.

[12] Manes, E.G.: Weakest preconditions: categorial insights, *Lecture Notes in Computer Science*, Vol. 240(1986), pp.182–197.

[13] Wagner, E.G.: A categorical view of weakest liberal preconditions, *Lecture Notes in Computer Science*, Vol. 240(1986), pp.198–205.

Yasuo Kawahara

Research Institute of Fundamental Information Science

Kyushu University 33

Fukuoka 812, Japan.

Email: kawahara@rifis.sci.kyushu-u.ac.jp

Yoshihiro Mizoguchi

Deptartment of Control Engineering and Science

Kyushu Institute of Technology

Iizuka 820, Japan.

Email: ym@ces.kyutech.ac.jp