IJMMS 2004:3, 135–160
PII. S0161171204204415X
http://ijmms.hindawi.com

DEMONIC SEMANTICS: USING MONOTYPES AND RESIDUALS

F. TCHIER

Received 13 April 2002

Relations and relational operators can be used to define the semantics of programming languages. The operations $\vee$ and $\circ$ serve to give *angelic semantics* by defining a program to go right when there is a possibility to go right. On the other hand, the demonic operations $\sqcup$ and $\Box$ do the opposite: if there is a possibility to go wrong, a program whose semantics is given by these operators will go wrong; it is the *demonic semantics*. This type of semantics is known at least since Dijkstra's introduction of the language of guarded commands. Recently, there has been a growing interest in demonic relational semantics of sequential programs. Usually, a construct is given an ad hoc semantic definition based on an intuitive understanding of its behavior. In this note, we show how the notion of *relational flow diagram* (essentially a matrix whose entries are relations on the set of states of the program), introduced by Schmidt, can be used to give a single demonic definition for a wide range of programming constructs. This research had originally been carried out by J. Desharnais and F. Tchier (1996) in the same framework of the binary homogeneous relations. We show that all the results can be generalized by using the monotypes and the residuals introduced by Desharnais et al. (2000).

2000 Mathematics Subject Classification: 18C10, 18C50, 68Q55, 68Q65, 03B70, 06B35.

1. Introduction. The approaches to semantics are categorized as operational, axiomatic, or denotational. We will be concerned with the operational and the denotational semantics of nondeterministic programs. The operational semantics is described by the relation between the initial and final states. In our case, we consider the worst execution of the program; that is, we suppose that the program behaves as badly as possible according to the *demonic relational semantics* . Usually this last one is given an ad hoc semantics definition based on an intuitive understanding of the behavior of the program. Denotational semantics has been introduced by Scott and Strachey. To give the denotational semantics, we associate to a program a mathematical object. In our case, this object is a flow diagram which is a graph whose arrows are weighted by the different steps of the program. The operations are "the demonic choice" and "demonic composition." In this note, we show how the notion of the flow diagram can be exploited to give single demonic operational semantics (with only demonic operators) for a wide range of programming constructs.

2. Relation algebras

2.1. Definition and basic laws. Both homogeneous and heterogeneous relation algebras are employed in computer science. In this note, we use heterogeneous relation algebras whose definition is taken from [9, 26, 27].

Definition 2.1. A *relation algebra* $\mathscr{A}$ is a structure $(A, \vee, \wedge, -, \circ, ^{\smile})$ over a nonempty set A of elements called *relations*. The unary operations $-$ and $^{\smile}$ are total whereas the binary operations $\vee$, $\wedge$, and $\circ$ are partial. By $A_{\vee R}$, the set of elements $Q \in A$ is denoted for which the union $R \vee Q$ is defined and $R \in A_{\vee R}$ for every $R \in A$ is required. If $Q \in A_{\vee R}$, Q *is of the same type* as R. The following conditions are satisfied.

(a) The structure $(A_{\vee R}, \vee, \wedge, -)$ is a Boolean algebra with *zero* element 0_R and *universal* element 1_R. The elements of $A_{\vee R}$ are ordered by an *inclusion*, denoted by $\leq$.

(b) If the products $P \circ R$ and $Q \circ R$ are defined, so is $P \circ Q^{\smile}$. If the products $P \circ Q$ and $P \circ R$ are defined, so is $Q^{\smile} \circ R$. If $Q \circ R$ exists, so does $Q \circ P$ for every $P \in A_{\vee R}$.

(c) The composition is associative: $P \circ (Q \circ R) = (P \circ Q) \circ R$.

(d) There are elements ${}_R\mathrm{id}$ and id_R associated to every relation $R \in A$. The relation ${}_R\mathrm{id}$ behaves as a right identity and the relation id_R as a left identity for $A_{\vee R}$.

(e) The Schröder rule $P \circ Q \leq R \Leftrightarrow P^{\smile} \circ -R \leq -Q \Leftrightarrow -R \circ Q^{\smile} \leq -P$ holds whenever one of the three expressions is defined.

(f) According to Tarski rule, $1 \circ R \circ 1 = 1$ if and only if $R \neq 0$ (provided $1 \circ R \circ 1 = 1$ is defined).

If $R^{\smile} \in A_{\vee R}$, then R is said to be *homogeneous*. If all $R \in \mathscr{A}$ have the same type, the operations are all total and $\mathscr{A}$ itself is said to be *homogeneous*.

For simplicity, the universal, zero, and identity elements are all denoted by 1, 0, and id, respectively. One can use subscripts to make the typing explicit, but this will not be necessary here. The precedence of the relational operators, from highest to lowest, is the following: $-$ and $^{\smile}$ bind equally, followed by $\circ$, followed by $\wedge$, and finally by $\vee$. The scope of $\bigvee_i$ and $\bigwedge_i$ goes to the right as far as possible. The relation $R^{\smile}$ is called the *converse* of R. The partial operations involved in relational expressions are assumed to be defined, even when it is not explicitly mentioned. Another operation that occurs in this note is the *reflexive transitive closure* R^*. It satisfies the well-known laws

$$R^* = \bigvee_{i \geq 0} R^i, \qquad R^* = \mathrm{id} \vee R \circ R^* = \mathrm{id} \vee R^* \circ R, \tag{2.1}$$

where $R^0 = \mathrm{id}$ and $R^{i+1} = R \circ R^i$. From Definition 2.1, the usual rules of the calculus of relations can be derived (see, e.g., [9, 11, 27]). We assume these rules to be known and simply recall a few of them.

Theorem 2.2. *Let $\mathscr{A}$ be a relation algebra and let $P, Q, R \in \mathscr{A}$. Then,*

(a) $P \wedge Q \leq R \Leftrightarrow P \leq -Q \vee R$,
(b) $P \circ (Q \wedge R) \leq P \circ Q \wedge P \circ R$,
(c) $(P \wedge Q) \circ R \leq P \circ R \wedge Q \circ R$,
(d) $P \circ (Q \vee R) = P \circ Q \vee P \circ R$,
(e) $(P \vee Q) \circ R = P \circ R \vee Q \circ R$,
(f) $Q \leq R \Rightarrow P \circ Q \leq P \circ R$,
(g) $P \leq Q \Rightarrow P \circ R \leq Q \circ R$,
(h) $Q \leq R \Leftrightarrow Q^{\smile} \leq R^{\smile}$,
(i) $(Q \vee R)^{\smile} = Q^{\smile} \vee R^{\smile}$,
(j) $(Q \wedge R)^{\smile} = Q^{\smile} \wedge R^{\smile}$,

(k) $(Q \circ R)^{\smile} = R^{\smile} \circ Q^{\smile}$,
(l) $R^{\smile\smile} = R$.

2.2. Galois connections. The notion of Galois connections is very important in what follows. There are many definitions of Galois connections [1]. We choose the following one [5].

Definition 2.3. Let $(S, \leq_S)$ and $(S', \leq_{S'})$ be two preordered sets. A pair (f, g) of functions, where $f : S \to S'$ and $g : S' \to S$, forms a *Galois connection* if and only if the following formula holds for all $x \in S$ and $y \in S'$:

$$f(x) \leq_{S'} y \iff x \leq_S g(y). \tag{2.2}$$

The function f is called the *lower adjoint* and g is the *upper adjoint*.

2.3. Residuals. We now present some operations which are closely related to composition and Galois connections, the *residuals* [5], defined as follows:

$$\begin{aligned} R \leq T/S &\iff R \circ S \leq T, \\ R \leq S\backslash T &\iff S \circ R \leq T, \end{aligned} \tag{2.3}$$

where R, S, and T are relations. The operators / and \ have been given a variety of names in the literature, such as, *left and right factor operators* [12]. We prefer the terminology *left and right residual operators* [18].

3. Monotypes and related operators

3.1. Monotypes. In the calculus of relations, there are two ways for viewing sets as relations; each of them has its own advantages. The first is via vectors: a relation x is a *vector* [27] if and only if $x = x \circ 1$. The second way is via *monotypes* [5]: a relation a is a monotype if and only if $a \leq \mathrm{id}$. The set of monotypes $\{a \mid a \in A_{\vee R}\}$, for a given R, is a complete Boolean lattice. We denote by $a^{\sim}$ the *monotype complement* of a. Monotypes have very simple and convenient properties. Some of them are presented in the following proposition. We draw the attention of the reader to Proposition 3.1(b); we will often use it without mention. It shows that for monotypes, composition and meet have the same effect.

Proposition 3.1. *Let a and b be monotypes. Then,*
(a) $a = a^{\smile} = a^2$,
(b) $a \circ b = a \wedge b = b \circ a$,
(c) $a \vee a^{\sim} = \mathrm{id}$ *and* $a \wedge a^{\sim} = 0$,
(d) $a \leq b \Leftrightarrow b^{\sim} \leq a^{\sim}$,
(e) $a^{\sim} \circ b^{\sim} = (a \vee b)^{\sim}$,
(f) $(a \wedge b)^{\sim} = (a \circ b)^{\sim} = a^{\sim} \vee b^{\sim}$,
(g) $a \circ b^{\sim} \vee b = a \vee b$,
(h) $a \circ b \leq c \Leftrightarrow c^{\sim} \circ b \leq a^{\sim}$,
(i) $a \leq b \Leftrightarrow a \circ 1 \leq b \circ 1$.

All these properties are obvious Boolean laws, except for (g) whose proof is as follows:

$$\begin{aligned} a \circ b^{\smile} \vee b &= a \circ b^{\smile} \vee a \circ b \vee b \quad (\text{since } a \circ b \leq b) \\ &= a \circ (b^{\smile} \vee b) \vee b \quad (\text{by Theorem 2.2(d)}) \\ &= a \vee b \quad (\text{since } \mathrm{id} = b^{\smile} \vee b). \end{aligned} \tag{3.1}$$

3.2. Domain and codomain operators. The domain and codomain of a relation R can be characterized by the vectors $R \circ 1$ and $R^{\smile} \circ 1$, respectively [15, 27]. They can also be characterized by the corresponding monotypes. In this note, we take the latter approach. In what follows, we formally define these operators and give some of their properties.

Definition 3.2. The *domain* and *codomain* operators of a relation R, denoted by $R^<$ and $R^>$, respectively, are the monotypes defined by the equations:

(a) $R^< = \mathrm{id} \wedge R \circ 1$,
(b) $R^> = \mathrm{id} \wedge 1 \circ R$.

These operators can also be characterized by Galois connections (see [5]): for each relation R and each monotype a,

$$R^< \leq a \iff R \leq a \circ 1, \tag{3.2}$$

$$R^> \leq a \iff R \leq 1 \circ a. \tag{3.3}$$

The domain and codomain operators are linked by the equation

$$R^> = R^{\smile <}, \tag{3.4}$$

as can be easily checked.

The proposition below presents some obvious properties of the domain operator. The corresponding properties of the codomain operator can be deduced by duality.

Proposition 3.3. *Let R and S be relations and let a be a monotype. Then,*

(a) $R^< \circ 1 = R \circ 1$,
(b) $(R \circ 1)^< = R^<$,
(c) $R^< \circ R = R$,
(d) $(R \circ S^<)^< = (R \circ S)^<$,
(e) $a^< = a$,
(f) $(a \circ R)^< = a \circ R^<$,
(g) $(R \vee S)^< = R^< \vee S^<$.

3.3. Monotype residuals

Definition 3.4. Let R be a relation and let a be a monotype. The *monotype right residual* and *monotype left residual* of a by R (called *factors* in [6]) are defined, respectively, by

(a) $a \mathbin{/\!\!\cdot} R := ((1 \circ a)/R)^>$,
(b) $R \mathbin{\backslash\!\!\cdot} a := (R \backslash (a \circ 1))^<$.

An alternative characterization of residuals can also be given by means of a Galois connection as follows [4]:

$$b \le a \not/ R \iff (b \circ R)^{>} \le a, \tag{3.5}$$
$$b \le R \not\backslash a \iff (R \circ b)^{<} \le a. \tag{3.6}$$

Since we do not use the operator $\not\backslash$ in the sequel, we only present some properties of $\not/$ in the next theorem.

Theorem 3.5. *Let P, Q, and R be relations and let a and b be monotypes. Then,*
(a) $(a \not/ Q) \not/ R = a \not/ (R \circ Q)$,
(b) $a \not/ (Q \vee R) = a \not/ Q \wedge a \not/ R$,
(c) $(a \wedge b) \not/ R = (a \circ b) \not/ R = a \not/ R \wedge b \not/ R$,
(d) $a \le b \Rightarrow a \not/ R \le b \not/ R$,
(e) $Q \le R \Rightarrow a \not/ R \le a \not/ Q$,
(f) $\mathrm{id} = \mathrm{id} \not/ R$,
(g) $\mathrm{id} = a \not/ 0$,
(h) $a = a \not/ \mathrm{id}$.

We now prove two additional properties of the monotype complement and monotype residual operators. The first of these properties is

$$a^{\sim} = 0 \not/ a. \tag{3.7}$$

Its proof goes as follows:

$$\begin{aligned} a^{\sim} \circ a \le 0 &\iff (a^{\sim} \circ a)^{>} \le 0 \quad \text{(by (3.4) and Proposition 3.3(e))} \\ &\iff a^{\sim} \le 0 \not/ a \quad \text{(by (3.5))} \\ &\iff \mathrm{id} \circ a^{\sim} \le 0 \not/ a \quad (\text{since } \mathrm{id} \circ a^{\sim} = a^{\sim}) \\ &\iff \mathrm{id} \le a \vee 0 \not/ a \quad \text{(by Theorem 2.2(a))}. \end{aligned} \tag{3.8}$$

The second one is a very interesting "implication":

$$a \not/ b = (a^{\sim} \circ b)^{\sim} = a \vee b^{\sim}. \tag{3.9}$$

Its proof is

$$\begin{aligned} a \not/ b &= (0 \not/ a^{\sim}) \not/ b \quad (\text{since } a = 0 \not/ a^{\sim}) \\ &= 0 \not/ (b \circ a^{\sim}) \quad \text{(by Theorem 3.5(a))} \\ &= (a^{\sim} \circ b)^{\sim} \quad \text{(by (3.7))} \\ &= a \vee b^{\sim} \quad \text{(by Proposition 3.1(f))}. \end{aligned} \tag{3.10}$$

Several properties of the complement operator suggest themselves. In the following, we have to use exhaustively the complement of the domain of a relation R, that is, the monotype a such that $a = R^{<\sim}$. To avoid the notation $R^{<\sim}$, we adopt the following notation:

$$R^{\prec} := R^{<\sim}. \tag{3.11}$$

We see the properties of the operator $^{\prec}$.

Proposition 3.6. *Let Q and R be relations and a and b monotypes. Then,*

(a) $0 \not/ Q^{\prec} = Q^{<}$,
(b) $Q^{\prec} \vee Q^{<} = \mathrm{id}$,
(c) $Q^{\prec} \wedge Q^{<} = 0$,
(d) $(R \vee Q)^{\prec} = R^{\prec} \wedge Q^{\prec}$,
(e) $(R^{<} \circ Q^{<})^{\prec} = R^{\prec} \vee Q^{\prec}$,
(f) $0 \not/ Q = Q^{\prec}$,
(g) $(Q \circ a^{\smile})^{\prec} = a \not/ Q$,
(h) $(Q \circ R^{\prec})^{\prec} = R^{<} \not/ Q$.

The first five properties are obviously deduced from (3.11) and Proposition 3.1.

Proof. For property (f), we have to prove that $0 \not/ Q$ satisfies the properties in Proposition 3.1(c):

$$\begin{aligned}
(0 \not/ Q) \circ Q \leq 0 &\Longrightarrow ((0 \not/ Q) \circ Q^{<} \leq 0) \quad \text{(Boolean law)} \\
&\Longleftrightarrow 0 \not/ Q \leq 0 \not/ Q^{<} \quad \text{(by Proposition 3.3(f) and (3.5))} \\
&\Longleftrightarrow \mathrm{id} \circ (0 \not/ Q) \leq 0 \not/ Q^{<} \quad (\text{since } \mathrm{id} \circ (0 \not/ Q) = 0 \not/ Q) \\
&\Longleftrightarrow \mathrm{id} \leq Q \vee 0 \not/ Q^{<} \quad \text{(by Theorem 2.2(a))}.
\end{aligned} \tag{3.12}$$

We now prove property (g):

$$\begin{aligned}
(Q \circ a^{\smile})^{\prec} &= 0 \not/ (Q \circ a^{\smile}) \quad \text{(by property (f))} \\
&= (0 \not/ a^{\smile}) \not/ Q \quad \text{(by Theorem 3.5(a))} \\
&= a \not/ Q \quad \text{(by (3.7))}.
\end{aligned} \tag{3.13}$$

Property (h) is a particular case of property (g) with $a = R^{<}$. □

We now give a definition of various properties of relations [26, 27].

Definition 3.7. A relation R is a *function* if and only if $R^{\smile} \circ R \leq \mathrm{id}$; it is *total* if and only if $R^{<} = \mathrm{id}$.

We will denote the least fixed point of the function f by μf. Similarly, νf denotes the greatest fixed point of f. Because we assume our relation algebra to be complete (see Definition 2.1), least and greatest fixed points of monotonic functions exist. We cite [13] as a general reference on fixed points.

Let f be a monotonic function. The following properties of fixed points are used below:

$$\mu f = \bigwedge \{X \mid f(X) = X\} = \bigwedge \{X \mid f(X) \leq X\}, \tag{3.14a}$$

$$\nu f = \bigvee \{X \mid f(X) = X\} = \bigvee \{X \mid X \leq f(X)\}, \tag{3.14b}$$

$$\mu f \leq \nu f, \tag{3.14c}$$

$$f(Y) \leq Y \Longrightarrow \mu f \leq Y, \tag{3.14d}$$

$$Y \leq f(Y) \Longrightarrow Y \leq \nu f. \tag{3.14e}$$

In the next subsection, we describe notions that are useful for the description of the set of initial states of a program for which termination is guaranteed. These notions are *progressive finiteness* and the *initial part* of a relation.

3.4. Progressive finiteness of a relation. A relation R is progressively finite in terms of points if and only if there are no infinite chains $s_0, \dots, s_i$ such that $s_i R s_{i+1}$ for all $i \geq 0$. In other words, there is no set of points y which are the starting points of some paths of infinite length. For every set of points y,

$$y \leq R \circ y \Longrightarrow y = 0. \tag{3.15}$$

The least set of points which are the starting points of paths of finite length, that is, from which we can proceed only finitely many steps, is called the *initial part* of R denoted by $\mathcal{I}(R)$. This topic is of interest in many areas of computer science and mathematics and is related to recursion and induction principle.

Definition 3.8. (a) The *initial part* of a relation R, denoted by $\mathcal{I}(R)$, is given by

$$\begin{aligned} \mathcal{I}(R) &= \bigwedge \{a \mid a \leq \mathrm{id} : a \swarrow R = a\} \\ &= \bigwedge \{a \mid a \leq \mathrm{id} : a \swarrow R \leq a\}, \\ \mathcal{I}(R)^{\sim} &= \bigvee \{a \mid a \leq \mathrm{id} : (R \circ a)^{<} = a\} \\ &= \mu(a : a \leq \mathrm{id} : a \swarrow R). \end{aligned} \tag{3.16}$$

(b) A relation R is said to be *progressively finite* [27] if and only if $\mathcal{I}(R) = \mathrm{id}$.

The description of $\mathcal{I}(R)$ by the formulation $a \swarrow R = a$ shows that $\mathcal{I}(R)$ exists since $(a \mid a \leq \mathrm{id} : a \swarrow R)$ is monotonic in the first argument (by Theorem 3.5(d)); and because the set of monotypes is a complete lattice, it follows from the fixed-point theorem of Knaster and Tarski that this function has a least fixed point. The progressive finiteness of a relation R is the same as the well-foundedness of $R^{\sim}$.

The initial part $\mathcal{I}(R)$ is a monotype. In a concrete setting, $\mathcal{I}(R)$ is the set of monotypes which are not the origins of infinite paths (by the relation R). Using formulas (3.14) and Boolean laws, one has

$$\begin{aligned} \mathcal{I}(R) &= \bigwedge \{a \mid a \leq \mathrm{id} : a^{\sim} = (R \circ a^{\sim})^{<}\} \\ &= \mu(a : a \leq \mathrm{id} : (R \circ a^{\sim})^{<}), \end{aligned} \tag{3.17}$$

$$\begin{aligned} \mathcal{I}(R)^{\sim} &= \bigvee \{a \mid a \leq \mathrm{id} : (R \circ a)^{<} = a\} \\ &= \bigvee \{a \mid a \leq (R \circ a)^{<}\} \\ &= \nu(a : a \leq \mathrm{id} : (R \circ a)^{<}). \end{aligned} \tag{3.18}$$

Proof. (a) We have

$$\begin{aligned} a = a \swarrow R &\iff a = (R \circ a^{\sim})^{<} \quad \text{(by Proposition 3.6(g))} \\ &\iff a^{\sim\sim} = ((R \circ a^{\sim})^{<})^{\sim} \quad \text{(by (3.11))} \\ &\iff a^{\sim} = (R \circ a^{\sim})^{<} \quad \text{(a complementation)}. \end{aligned} \tag{3.19}$$

(b) We have

$$\begin{aligned}\mathcal{J}(R) &= \bigwedge\{a \mid a^{\sim} = (R\circ a^{\sim})^{<}\} \Longleftrightarrow \mathcal{J}(R)^{\sim} = \bigvee\{a^{\sim} \mid a^{\sim} = (R\circ a^{\sim})^{<}\} \quad \text{(De Morgan law)}\\ &= \mathcal{J}(R)^{\sim} = \bigvee\{b \mid b = (R\circ b)^{<}\} \quad \text{(since } b = a^{\sim}\text{)}.\end{aligned} \tag{3.20}$$

□

Definition 3.9. A relation R is *progressively finite* if and only if, for a monotype a,

$$a \le (R\circ a)^{<} \Longrightarrow a = 0. \tag{3.21}$$

Equivalently, $\nu(a : a \le \mathrm{id} : (R\circ a)^{<}) = 0$ and $\mu(a : a \le \mathrm{id} : a \backslash R) = \mathrm{id}$.

In [4], it is shown that the following definitions are also equivalent:

(a) a relation R is *progressively finite* if and only if, for any vector v,

$$v \le R\circ v \Longrightarrow v = 0; \tag{3.22}$$

(b) a relation R is *progressively finite* if and only if, for any relation Q,

$$Q \le R\circ Q \Longrightarrow Q = 0. \tag{3.23}$$

The next theorem involves the function $w_a(X) := Q \vee P\circ X$, which is closely related to the description of iterations. The theorem highlights the importance of progressive finiteness in the simplification of fixed-point-related properties.

Theorem 3.10. *If the relation P is progressively finite, then the function $(X : Q \vee P\circ X)$ admits a unique fixed point and $\nu(X : Q \vee P\circ X) = \mu(X : Q \vee P\circ X) = P^{*}\circ Q$.*

To close this subsection, we demonstrate some simple useful properties.

Proposition 3.11. *Let R and S be relations, b a monotype, and $I(R) = \bigwedge\{x \mid x = x\circ 1 : -x = R\circ -x\}$. Then,*

(a) *$\mathcal{J}(R)\circ R$ is progressively finite;*
(b) *if R is progressively finite, then $R \wedge S$ is progressively finite;*
(c) *if R is progressively finite, then $b\circ R$ is progressively finite;*
(d) *$\mathcal{J}(R) = \mathcal{J}(R)\backslash R$;*
(e) *$\mathcal{J}(R) = \mathrm{id} \wedge I(R)$.*

Proof. (a) We use Definition 3.9:

$$\begin{aligned}a \le (\mathcal{J}(R)\circ R\circ a)^{<} &\Longleftrightarrow a \le (\mathcal{J}(R))\circ (R\circ a)^{<} \quad \text{(by Proposition 3.3(f))}\\ &\Longleftrightarrow a \le \mathcal{J}(R), \quad a \le (R\circ a)^{<}\\ &\qquad\qquad \text{(since } \circ = \wedge \text{ for monotypes and Boolean law)} \quad (3.24)\\ &\Longrightarrow a \le \mathcal{J}(R), \quad a \le \mathcal{J}(R)^{\sim} \quad \text{(by (3.18) and (3.14e))}\\ &\Longleftrightarrow a \le 0.\end{aligned}$$

(b) Using the same definition,

$$\begin{aligned} a \le \big((R \wedge S) \circ a\big)^{<} &\Longrightarrow a \le (R \circ a)^{<} \quad \text{(Boolean law)} \\ &\Longrightarrow a = 0 \quad \text{(by (3.15))}. \end{aligned} \tag{3.25}$$

(c) It is a particular case of (b), and the proof goes as follows:

$$\begin{aligned} a \le (b \circ R \circ a)^{<} &\Longrightarrow a \le (R \circ a)^{<} \quad (\text{since } b \le \mathrm{id} \text{ and by monotonicity of } {}^{<} \text{ w.r.t. } \le) \\ &\Longrightarrow a = 0 \quad \text{(by (3.15))}. \end{aligned} \tag{3.26}$$

(d) By definition (3.17), $\mathcal{G}(R)$ is the least monotype a verifying $a = a \backslash R$.

(e) We have to show that, for each monotype, there exists a monotype a satisfying the condition $(R \circ a^{\sim})^{<} \le a^{\sim}$, if and only if there exists a vector x such that $a = \mathrm{id} \wedge x$ and x verifies $R \circ -x \le -x$:

$$\begin{aligned} (R \circ a^{\sim})^{<} \le a^{\sim} &\Longleftrightarrow (R \circ a^{\sim})^{<} \circ 1 \le a^{\sim} \circ 1 \quad \text{(Proposition 3.1(a))} \\ &\Longleftrightarrow R \circ a^{\sim} \circ 1 \le a^{\sim} \circ 1 \quad \text{(by Proposition 3.3(a))} \\ &\Longleftrightarrow R \circ -x \le -x \quad (\text{since } x = a^{\sim} \circ 1). \end{aligned} \tag{3.27}$$

□

The precedence from highest to lowest is the following: ${}^{\prec}$, ${}^{<}$, ${}^{>}$, ${}^{\sim}$, $-$, and ${}^{\smile}$ bind equally, followed by $\circ$, $\backslash$, $\wedge$, and finally by $\vee$.

The set of matrices whose entries are relations constitutes a relation algebra [27] with the operators defined as follows.

Definition 3.12. Let R and S be matrices whose entries belong to the same homogeneous algebra. Then,

$$\begin{gathered} (R \vee S)_{i,j} = R_{i,j} \vee S_{i,j}, \qquad (-R)_{i,j} = -R_{i,j}, \qquad (R \circ S)_{i,j} = \bigvee_k R_{i,k} \circ S_{k,j}, \\ (R \wedge S)_{i,j} = R_{i,j} \wedge S_{i,j}, \quad (R_{j,i})^{\smile} = R^{\smile}{}_{i,j}, \quad R \le S \Longleftrightarrow R_{i,j} \le S_{i,j}, \forall i, j, \\ 1_{i,j} = 1, \qquad 0_{i,j} = 0, \qquad \mathrm{id}_{i,j} = \begin{cases} \mathrm{id}, & i = j, \\ 0, & \text{otherwise}, \end{cases} \end{gathered} \tag{3.28}$$

where $R_{i,j}$ denotes the entry i, j of matrix R. Of course, $R \vee S$ and $R \wedge S$ exist only if matrices R and S have the same dimension; the composition $R \circ S$ exists only if the number of columns of R is the same as the number of rows of S. The entries of the identity matrix (which is square) are 0, except those of the diagonal, which are id. The entries of the zero matrix are 0 and those of the universal matrix are 1.

It is recalled by the next examples how some of the angelic operators are applied to Boolean matrices.

Let R be a relation on $S \times S$, then the operations $\circ, ^{\smile}$:

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \circ \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}^{\smile} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}. \tag{3.29}$$

As the demonic calculus will serve as an algebraic apparatus for defining the denotational semantics of the nondeterministic programs, we will define in what follows these operations.

4. A demonic refinement ordering. We now define the refinement ordering (*demonic inclusion*) we will use in the sequel. This ordering induces a complete join semilattice, called a *demonic semilattice*. The associated operations are demonic join ($\sqcup$), demonic meet ($\sqcap$), and demonic composition ($\diamond$). We give the definitions and needed properties of these operations, and illustrate them with simple examples. For more details on relational demonic semantics and demonic operators, see [6, 7, 8, 9, 14].

Definition 4.1. A relation Q *refines* a relation R (see [24]), denoted by

$$Q \sqsubseteq R \iff R^{<} \circ Q \leq R, \quad R^{<} \leq Q^{<}. \tag{4.1}$$

Thus, for instance,

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix} \sqsubseteq \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{4.2}$$

but

$$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \not\sqsubseteq \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix} \not\sqsubseteq \begin{pmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 1 \end{pmatrix}. \tag{4.3}$$

(These Boolean matrices represent relations over sets by the well-known correspondence.)

Proposition 4.2. *Let Q and R be relations. Then the following statements hold.*

(a) *The greatest lower bound (with respect to $\sqsubseteq$) of Q and R is*

$$Q \sqcup R = Q^{<} \circ R^{<} \circ (Q \vee R). \tag{4.4}$$

If $Q^{<} = R^{<}$, then $\sqcup$ and $\vee$ coincide, that is, $Q \sqcup R = Q \vee R$.

(b) *If Q and R satisfy the condition $Q^< \wedge R^< = (Q \wedge R)^<$, their least upper bound is*

$$Q \sqcap R = Q \wedge R \vee Q^< \circ R \vee R^< \circ Q, \tag{4.5}$$

otherwise, the least upper bound does not exist. If $Q^< \wedge R^< = 0$, then $\sqcap$ and $\wedge$ coincide, that is, $Q \sqcap R = Q \wedge R$.

For the proofs, see [10, 14]. Here is an example of these operations:

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \sqcup \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{4.6}$$

This operation corresponds to a demonic nondeterministic choice since the possibility of failure (row 3 of the first matrix or row 1 of the second) is reflected in the result. For the middle row, failure is not possible, and the set of allowed results is the union of the results of the two operands.

Secondly, demonic meet: the existence condition simply means that, on the intersection of their domains, Q and R have to agree for at least one value. For example, consider

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \sqcap \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \tag{4.7}$$

on the intersection of their domains (the second row), the operands agree on the middle value and thus the meet is defined. This is not the case for $\left(\begin{smallmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right)$ and $\left(\begin{smallmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{smallmatrix}\right)$, because they contradict each other on the intersection of their domains.

It is shown in [14] that it is a complete join semilattice. Let f be a monotonic function (with respect to $\sqsubseteq$) having at least one fixed point. Because $(A_{\vee R}, \sqsubseteq)$ is a complete join semilattice, the following properties of fixed points can be transferred from equations (3.14):

(a) $\nu f = \bigsqcup\{X \mid f(X) = X\} = \bigsqcup\{X \mid X \sqsubseteq f(X)\}$,

(b) $Y \sqsubseteq f(Y) \Rightarrow Y \sqsubseteq \nu f$.

In what follows, we will present some properties of functions.

Lemma 4.3. *Let P, Q, and R be relations. Although composition does not distribute over intersection in general (see Theorem 2.2(b)), it does in the following special cases:*

(a) *P function* $\Rightarrow P \circ (Q \wedge R) = P \circ Q \wedge P \circ R$;

(b) *Q function* $\Rightarrow R^< \not\!/ Q = Q^< \vee (Q \circ R)^<$.

Proof. (a) See [27].

(b) As Q is a function, we have

$$\begin{aligned} Q^\smile \circ Q \le \mathrm{id} &\Longrightarrow Q^\smile \circ Q \circ R \le R \quad \text{(by Theorem 2.2(g))} \\ &\Longrightarrow ((Q^\smile \circ Q \circ R))^< \le R^< \quad (^<\text{monotonic with respect to } \le) \\ &\Longleftrightarrow (Q^\smile \circ (Q \circ R)^<)^< \le R^< \quad \text{(by Proposition 3.3(d))} \end{aligned}$$

$$\begin{aligned}
&\iff (((Q\circ R)^<)\circ Q)^> \le R^< \quad \text{(by (3.4) and Proposition 3.1(a))}\\
&\iff (Q\circ R)^< \le R^< \not/ Q \quad \text{(by (3.5))}\\
&\implies Q^< \vee (Q\circ R)^< \le R^< \not/ Q \quad \text{(since } 0 \le R^< \text{ and by Theorem 3.5(d))}.
\end{aligned} \tag{4.8}$$

For the other side, we have

$$\begin{aligned}
&(R^< \not/ Q) \wedge (Q\circ R)^< \le (Q\circ R)^<\\
&\quad\iff (R^< \not/ Q) \wedge ((Q\circ R)^< \vee (Q\circ R^<)^<) \le (Q\circ R)^<\\
&\qquad\qquad \text{(by Theorem 2.2(d) and Proposition 3.6(h))}\\
&\quad\iff (R^< \not/ Q) \wedge Q^< \le (Q\circ R)^< \quad \text{(since } Q^< = (Q\circ R)^< \vee (Q\circ R^<)^<)\\
&\quad\iff R^< \not/ Q \le Q^< \vee (Q\circ R)^< \quad \text{(by Theorem 2.2(a))}.
\end{aligned} \tag{4.9}$$

□

Definition 4.4. The *demonic composition* of relations Q and R [6] is $Q \sqcap R = (R^< \not/ Q)\circ Q\circ R$.

In what follows, we present some properties of $\sqcap$.

Theorem 4.5. *Let P, Q, and R be relations. Then,*
(a) $(P \sqcap Q) \sqcap R = P \sqcap (Q \sqcap R)$,
(b) *R total* $\Rightarrow Q \sqcap R = Q\circ R$,
(c) *Q function* $\Rightarrow Q \sqcap R = Q\circ R$.

Proof. (a) See [6, 7, 8, 14].
(b) We have

$$\begin{aligned}
Q \sqcap R &= (R^< \not/ Q)\circ Q\circ R \quad \text{(by Definition 4.4)}\\
&= Q\circ R \quad \text{(since } R^< = \mathrm{id} \text{ and by Theorem 3.5(f))}.
\end{aligned} \tag{4.10}$$

(c) We have

$$\begin{aligned}
Q \sqcap R &= (R^< \not/ Q)\circ Q\circ R \quad \text{(by Definition 4.4)}\\
&= (Q^< \vee (Q\circ R)^<)\circ Q\circ R \quad \text{(by Lemma 4.3(b))}\\
&= Q^<\circ Q \vee (Q\circ R)^<\circ Q\circ R \quad \text{(by Theorem 2.2(d))}\\
&= Q\circ R \quad \text{(by Proposition 3.3(c) and Proposition 3.6(f), for } Q^<\circ Q = 0).
\end{aligned} \tag{4.11}$$

□

We will present some results that will be used in the sequel.

Proposition 4.6. *Let Q and R be relations and let a be a monotype. Then,*
(a) $R \sqsubseteq a\circ R$,
(b) $R \sqcap a = (a \not/ R)\circ R$,
(c) $(Q \sqcap R)^< = (R^< \not/ Q)\circ Q^<$,
(d) $Q \sqcap (a\circ R) = ((a \not/ Q)\circ Q) \sqcap R$,
(e) $(a\circ Q) \sqcap R = a\circ (Q \sqcap R)$.

PROOF. (a) We have

$$R \sqsubseteq a \circ R \iff (a \circ R)^{<} \circ R \leq a \circ R, \quad (a \circ R)^{<} \leq R^{<} \quad \text{(by Definition 4.1)} \tag{4.12}$$

which is true by Proposition 3.3(c) and (f), for $a \leq \mathrm{id}$.

(b) We have

$$\begin{aligned} R \sqsupset a &= (a \not\leftarrow R) \circ R \circ a \quad \text{(by Definition 4.4)} \\ &= (a \not\leftarrow R) \circ (R \circ a \vee R \circ a^{\sim}) \quad \text{(by Proposition 3.6(g), for } (a \not\leftarrow R) \circ R \circ a^{\sim} = 0) \\ &= (a \not\leftarrow R) \circ R \quad (\text{since } R \circ a \vee R \circ a^{\sim} = R \circ (a \vee a^{\sim}) = R \circ \mathrm{id} = R). \end{aligned} \tag{4.13}$$

(c) We have

$$\begin{aligned} (Q \sqsupset R)^{<} &= \big((R^{<} \not\leftarrow Q) \circ Q \circ R\big)^{<} \quad \text{(by Definition 4.4)} \\ &= (R^{<} \not\leftarrow Q) \circ (Q \circ R)^{<} \quad \text{(by Proposition 3.3(f))} \\ &= (R^{<} \not\leftarrow Q) \circ \big[(Q \circ R)^{<} \vee (Q \circ R^{\prec})^{<}\big] \\ &\qquad \text{(by Proposition 3.6(h), for } (R^{<} \not\leftarrow Q) \circ (Q \circ R^{\prec})^{<} = 0) \\ &= (R^{<} \not\leftarrow Q) \circ Q^{<} \\ &\qquad (\text{since } (Q \circ R)^{<} \vee (Q \circ R^{\prec})^{<} = (Q \circ (R^{<} \vee R^{\prec}))^{<} = (Q \circ \mathrm{id})^{<} = Q). \end{aligned} \tag{4.14}$$

(d) We have

$$\begin{aligned} Q \sqsupset (a \circ R) &= Q \sqsupset (a \sqsupset R) \quad \text{(by Theorem 4.5(c))} \\ &= (Q \sqsupset a) \sqsupset R \quad \text{(by Theorem 4.5(a))} \\ &= \big((a \not\leftarrow Q) \circ Q\big) \sqsupset R \quad \text{(from (b))} \\ &= (a \not\leftarrow Q) \circ Q \sqsupset R \quad \text{(by associativity of } \circ). \end{aligned} \tag{4.15}$$

(e) We have

$$\begin{aligned} (a \circ Q) \sqsupset R &= (a \sqsupset Q) \sqsupset R \quad \text{(by Theorem 4.5(c))} \\ &= a \sqsupset (Q \sqsupset R) \quad \text{(by Theorem 4.5(a))} \\ &= a \circ (Q \sqsupset R) \quad \text{(by Theorem 4.5(c))}. \end{aligned} \tag{4.16}$$

□

After having introduced the monotype operators and some of their properties, in the following, we present these monotype operators applied to matrices. As we already know, a set of such matrices of suitable dimensions constitutes a relation algebra [26, 27]. A monotype matrix is a diagonal matrix such that each entry of the diagonal is included in the identity relation. In what follows, we will present some results related to the operators $^{<}$, $^{>}$, $\not\leftarrow$, $^{\sim}$, and $^{\prec}$ applied to matrices.

THEOREM 4.7. *Let R_i be relations and a_i monotypes, where $1 \le i \le 2$. Then*

(a)

$$\begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{<} = \begin{pmatrix} R_1^{<} \vee R_2^{<} & 0 \\ 0 & R_3^{<} \vee R_4^{<} \end{pmatrix}, \tag{4.17}$$

(b)

$$\begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{>} = \begin{pmatrix} R_1^{>} \vee R_3^{>} & 0 \\ 0 & R_2^{>} \vee R_4^{>} \end{pmatrix}, \tag{4.18}$$

(c)

$$\begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} \not{} \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix} = \begin{pmatrix} a_1 \not{} R_1 \wedge a_2 \not{} R_2 & 0 \\ 0 & a_1 \not{} R_3 \wedge a_2 \not{} R_4 \end{pmatrix}, \tag{4.19}$$

(d)

$$\begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix}^{\sim} = \begin{pmatrix} a_1^{\sim} & 0 \\ 0 & a_2^{\sim} \end{pmatrix}, \tag{4.20}$$

(e)

$$\begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{\prec} = \begin{pmatrix} R_1^{\prec} \wedge R_2^{\prec} & 0 \\ 0 & R_3^{\prec} \wedge R_4^{\prec} \end{pmatrix}. \tag{4.21}$$

For our proofs, we use the rule of indirect equality [6]: for all Q and R,

$$Q = R \iff \{\forall S \mid Q \le S \iff R \le S\}. \tag{4.22}$$

PROOF. (a) Let b_i be monotypes where $1 \le i \le 2$. Then,

$$\begin{aligned}
&\begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{<} \le \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} \\
&\quad \iff \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix} \le \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} \circ \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \quad \text{(by (3.2))} \\
&\quad \iff \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix} \le \begin{pmatrix} b_1 \circ 1 & b_1 \circ 1 \\ b_2 \circ 1 & b_2 \circ 1 \end{pmatrix} \quad \text{(by Definition 3.12)} \\
&\quad \iff R_1 \le b_1 \circ 1, \quad R_2 \le b_1 \circ 1, \quad R_3 \le b_2 \circ 1, \quad R_4 \le b_2 \circ 1 \quad \text{(by Definition 3.12)} \\
&\quad \iff R_1^{<} \le b_1, \quad R_2^{<} \le b_1, \quad R_3^{<} \le b_2, \quad R_4^{<} \le b_2 \quad \text{(by (3.2))} \\
&\quad \iff R_1^{<} \vee R_2^{<} \le b_1, \quad R_3^{<} \vee R_4^{<} \le b_2 \quad \text{(Boolean law)} \\
&\quad \iff \begin{pmatrix} R_1^{<} \vee R_2^{<} & 0 \\ 0 & R_3^{<} \vee R_4^{<} \end{pmatrix} \le \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} \quad \text{(by Definition 3.12).}
\end{aligned} \tag{4.23}$$

(b)

$$\begin{aligned}
\begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{>} &= \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{\smile <} && \text{(by (3.4))} \\
&= \begin{pmatrix} R_1^{\smile} & R_3^{\smile} \\ R_2^{\smile} & R_4^{\smile} \end{pmatrix}^{<} && \text{(by Definition 3.12)} \\
&= \begin{pmatrix} R_1^{\smile <} \vee R_3^{\smile <} & 0 \\ 0 & R_2^{\smile <} \vee R_4^{\smile <} \end{pmatrix} && \text{(from (a))} \\
&= \begin{pmatrix} R_1^{>} \vee R_3^{>} & 0 \\ 0 & R_2^{>} \vee R_4^{>} \end{pmatrix} && \text{(by (3.4)).}
\end{aligned} \tag{4.24}$$

(c)

$$\begin{aligned}
&\begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} \leq \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} \not{\leftarrow} \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix} \\
&\iff \left(\begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} \circ \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix} \right)^{>} \leq \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} && \text{(by (3.5))} \\
&\iff \begin{pmatrix} b_1 \circ R_1 & b_1 \circ R_2 \\ b_2 \circ R_3 & b_2 \circ R_4 \end{pmatrix}^{>} \leq \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} && \text{(by Definition 3.12)} \\
&\iff \begin{pmatrix} (b_1 \circ R_1)^{>} \vee (b_2 \circ R_3)^{>} & 0 \\ 0 & (b_1 \circ R_2)^{>} \vee (b_2 \circ R_4)^{>} \end{pmatrix} \leq \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} && \text{(from (b))} \\
&\iff (b_1 \circ R_1)^{>} \vee (b_2 \circ R_3)^{>} \leq a_1, \quad (b_1 \circ R_2)^{>} \vee (b_2 \circ R_4)^{>} \leq a_2 && \text{(4.25)} \\
&\qquad\qquad \text{(by Definition 3.12)} \\
&\iff (b_1 \circ R_1)^{>} \leq a_1, \quad (b_2 \circ R_3)^{>} \leq a_1, \quad (b_1 \circ R_2)^{>} \leq a_2, \quad (b_2 \circ R_4)^{>} \leq a_2 \\
&\qquad\qquad \text{(Boolean law)} \\
&\iff b_1 \leq a_1 \not{\leftarrow} R_1, \quad b_2 \leq a_1 \not{\leftarrow} R_3, \quad b_1 \leq a_2 \not{\leftarrow} R_2, \quad b_2 \leq a_2 \not{\leftarrow} R_4 \quad \text{(by (3.5))} \\
&\iff b_1 \leq a_1 \not{\leftarrow} R_1 \wedge a_2 \not{\leftarrow} R_2, \quad b_2 \leq a_1 \not{\leftarrow} R_3 \wedge a_2 \not{\leftarrow} R_4 \quad \text{(Boolean law)} \\
&\iff \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} \leq \begin{pmatrix} a_1 \not{\leftarrow} R_1 \wedge a_2 \not{\leftarrow} R_2 & 0 \\ 0 & a_1 \not{\leftarrow} R_3 \wedge a_2 \not{\leftarrow} R_4 \end{pmatrix} \quad \text{(by Definition 3.12).}
\end{aligned}$$

(d)

$$\begin{aligned}
\begin{pmatrix} a_1^{\sim} & 0 \\ 0 & a_2^{\sim} \end{pmatrix} \vee \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} &= \begin{pmatrix} a_1^{\sim} \vee a_1 & 0 \\ 0 & a_2^{\sim} \vee a_2 \end{pmatrix} = \begin{pmatrix} \mathrm{id} & 0 \\ 0 & \mathrm{id} \end{pmatrix} \quad \text{(by (3.7))}, \\
\begin{pmatrix} a_1^{\sim} & 0 \\ 0 & a_2^{\sim} \end{pmatrix} \wedge \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} &= \begin{pmatrix} a_1^{\sim} \wedge a_1 & 0 \\ 0 & a_2^{\sim} \wedge a_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{(by (3.7))}.
\end{aligned} \tag{4.26}$$

From this, we conclude that

$$\begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix}^{\sim} = \begin{pmatrix} a_1^{\sim} & 0 \\ 0 & a_2^{\sim} \end{pmatrix}. \tag{4.27}$$

(e)

$$\begin{aligned} \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{\prec} &= \begin{pmatrix} R_1 & R_2 \\ R_3 & R_4 \end{pmatrix}^{<\sim} \quad \text{(by (3.11))} \\ &= \begin{pmatrix} R_1^{<} \vee R_2^{<} & 0 \\ 0 & R_3^{<} \vee R_4^{<} \end{pmatrix}^{\sim} \quad \text{(from (a))} \\ &= \begin{pmatrix} (R_1^{<} \vee R_2^{<})^{\sim} & 0 \\ 0 & (R_3^{<} \vee R_4^{<})^{\sim} \end{pmatrix} \quad \text{(from (d))} \\ &= \begin{pmatrix} R_1^{\prec} \wedge R_2^{\prec} & 0 \\ 0 & R_3^{\prec} \wedge R_4^{\prec} \end{pmatrix} \quad \text{(from (e) and (3.7))}. \end{aligned} \tag{4.28}$$

□

The previous results will be generalized as follows.

Lemma 4.8. *Let R be a matrix of suitable dimensions and let a be a monotype matrix, that is, with monotypes on the diagonal entries and on the null relation otherwise. Then,*

$$\begin{aligned} (R^{<})_{i,j} &= \begin{cases} \bigvee_k R^{<}_{i,k}, & i = j, \\ 0, & \text{otherwise}, \end{cases} \\ (R^{>})_{i,j} &= \begin{cases} \bigvee_k R^{>}_{k,i}, & i = j, \\ 0, & \text{otherwise}, \end{cases} \\ (a \backslash R)_{i,j} &= \begin{cases} \bigwedge_k a_{k,k} \backslash R_{i,k}, & i = j, \\ 0, & \text{otherwise}, \end{cases} \\ (a^{\sim})_{i,j} &= \begin{cases} (a_{i,i})^{\sim}, & i = j, \\ 0, & \text{otherwise}, \end{cases} \\ (R_{i,j})^{\prec} &= \begin{cases} \bigwedge_k R^{\prec}{}_{i,k}, & i = j, \\ 0, & \text{otherwise}, \end{cases} \end{aligned} \tag{4.29}$$

where $R_{i,j}$ denotes the entry i, j of matrix R. Of course, $a \backslash R$ exists only if the number of rows of the matrix a is the same as the number of columns of the matrix R.

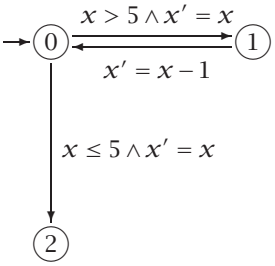
In the next section, we will briefly define *flow diagram programs* [26, 27] to use such notion in our definition of the operational semantics.

5. Demonic relational semantics

5.1. Programs and their semantics. We will use the notion of *flow diagram programs* [26, 27], where the relation algebraic concept of a program is based on flowcharts which describe the control flow of a program in terms of graphs. They distinguish the various steps of the program during its execution without taking into account neither the programming language nor the syntax of the programs. In what follows, we present an example of a program by means of the underlying flowgraphs whose arrows are weighted by relations representing the program steps. Consequently, these relations are the entries of the associated matrix of the graph.

We illustrate the ideas with a general example.

Example 5.1. While $x > 5$, do $x := x - 1$, then we have



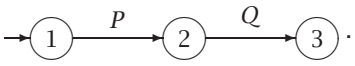
(5.1)

If $x \in \mathbb{N}$ is a relation on $\mathbb{N} \times \{1,2,3\}$, we have

$$\begin{pmatrix} 0 & x > 5 \wedge x' = x & x \leq 5 \wedge x' = x \\ x' = x - 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{5.2}$$

We start our development directly from this matrix representation of (relational) flow diagram programs. How these matrices can be obtained from programs is intuitively obvious, and we refer to [26, 27] for a rigorous treatment.

We illustrate the ideas with a more general example. Let S be the set of states of a program $\mathcal{R}$ which is a sequence of two statements p and q. Assume that the relations on S computed by these two statements are P and Q. Then $\mathcal{R}$ can be represented by the following graph:



(5.3)

The corresponding matrix representation is

$$R = \begin{pmatrix} 0 & P & 0 \\ 0 & 0 & Q \\ 0 & 0 & 0 \end{pmatrix}. \tag{5.4}$$

To represent programs, we use matrices like this one. The entries of these matrices are relations on the set of states of the program under consideration. For simplicity, we consider only the case where an entry represents a single program step rather than a complex program.

Now, to extract the input-output relation of the program from these matrices, we need two other relations (matrices), namely, the input relation ε (*entry*) and the output relation ξ (*exit*) [26, 27]. For the case of the sequence above, these relations are

$$\varepsilon = \begin{pmatrix} \mathrm{id} \\ 0 \\ 0 \end{pmatrix}, \qquad \xi = \begin{pmatrix} 0 \\ 0 \\ \mathrm{id} \end{pmatrix}. \tag{5.5}$$

Notice how these matrices select the entry and the exit points (via their nonzero entries). Because these nonzero entries are id, this selection implies no change in the state of the program. The $\varepsilon^\smile$ matrix is a function (Definition 3.7) because there is a single entry point; for example, for the particular ε given above,

$$\varepsilon \circ \varepsilon^\smile = \begin{pmatrix} \mathrm{id} \\ 0 \\ 0 \end{pmatrix} \circ \begin{pmatrix} \mathrm{id} & 0 & 0 \end{pmatrix} = \begin{pmatrix} \mathrm{id} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \leq \mathrm{id}. \tag{5.6}$$

Let $\mathcal{R}$ be a program with associated matrix R. The *terminating action* of $\mathcal{R}$ [26, 27] is the relation

$$T := R^* \circ R^\prec. \tag{5.7}$$

It links a state s with all the last states ($R^\prec$) reachable from s by R in a computation sequence (R^*).

We are now ready to define $\mathcal{S}(\mathcal{R})$, the *demonic input-output semantics* of $\mathcal{R}$:

$$\mathcal{S}(\mathcal{R}) := \varepsilon^\smile \sqcap \mathcal{I}(R) \sqcap T \sqcap \xi. \tag{5.8}$$

All the operators are demonic and this is an advantage of the use of monotypes in our development.

Expression (5.8) can be transformed as follows. Using Definition 4.4, the fact that $\varepsilon^\smile$ and $\mathcal{I}(R)$ are functions, and Theorem 4.5(c), one gets

$$\mathcal{S}(\mathcal{R}) := \varepsilon^\smile \circ \mathcal{I}(R) \circ (\xi^< \not\leftarrow T) \circ T \circ \xi. \tag{5.9}$$

The relation $\varepsilon^\smile \circ \mathcal{I}(R) \circ (\xi^< \not\leftarrow T) \circ T \circ \xi$ represents executions starting from the states that cannot lead to infinite loops ($\varepsilon^\smile \circ \mathcal{I}(R)$), leading to the exit point (ξ) and linking two states s and s' if s' cannot be acted upon by the program (the term $R^\prec$ in T), and there is a path from s to s' (the term R^* in T).

To illustrate this definition, we present a simple case, that of sequences.

5.2. Sequences. In this subsection, we calculate $\mathcal{S}(\mathcal{R})$ for the case where $\mathcal{R}$ is a sequence. We will calculate the subterms of $\mathcal{S}(\mathcal{R})$ (equation (5.9)), using the matrix R representing a sequence (equation (5.4)) and the input-output relations ε and ξ. We use the results of Lemma 4.8 to calculate the next expressions.

First

$$R^2 = \begin{pmatrix} 0 & P & 0 \\ 0 & 0 & Q \\ 0 & 0 & 0 \end{pmatrix} \circ \begin{pmatrix} 0 & P & 0 \\ 0 & 0 & Q \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & P \circ Q \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{5.10}$$

and it is easy to check that $R^n = 0$ for $n \geq 3$, whence

$$\begin{gathered} R^* = \mathrm{id} \vee R \vee R^2 = \begin{pmatrix} I & P & P \circ Q \\ 0 & \mathrm{id} & Q \\ 0 & 0 & \mathrm{id} \end{pmatrix}, \qquad R^{\prec} = \begin{pmatrix} P^{\prec} & 0 & 0 \\ 0 & Q^{\prec} & 0 \\ 0 & 0 & \mathrm{id} \end{pmatrix}, \\ \varepsilon = \begin{pmatrix} \mathrm{id} \\ 0 \\ 0 \end{pmatrix}, \qquad \xi = \begin{pmatrix} 0 \\ 0 \\ \mathrm{id} \end{pmatrix}. \end{gathered} \tag{5.11}$$

So, the terminating action of $\mathcal{R}$ is

$$T = R^* \circ R^{\prec} = \begin{pmatrix} P^{\prec} & P \circ Q^{\prec} & P \circ Q \\ 0 & Q^{\prec} & Q \\ 0 & 0 & \mathrm{id} \end{pmatrix} \tag{5.12}$$

and the exit relation is

$$\xi^{\prec} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \mathrm{id} \end{pmatrix}. \tag{5.13}$$

Remark 5.2. As we will see, the matrix of the next term takes too much space. In what follows, we will give all the details, but in the following, we will omit the terms $\mathrm{id} \not P$, $P \not 0$, and $P \not \mathrm{id}$, where P is a relation because by Theorem 3.5(f), (g), and (h), each of them is equal to id. So, we will just refer to the remark

$$\xi^{\prec} \not T = \begin{pmatrix} (0 \not P^{\prec}) \circ (0 \not (P \circ Q^{\prec})) & 0 & 0 \\ 0 & 0 \not Q^{\prec} & 0 \\ 0 & 0 & \mathrm{id} \end{pmatrix}. \tag{5.14}$$

By Proposition 3.6(a), (f), and (g), and Remark 5.2, we have

$$\xi^{<} \not{\circ} T = \begin{pmatrix} P^{<} \circ (Q^{<} \not{\circ} P) & 0 & 0 \\ 0 & Q^{<} & 0 \\ 0 & 0 & \mathrm{id} \end{pmatrix}, \qquad T \circ \xi = \begin{pmatrix} P \circ Q \\ Q \\ \mathrm{id} \end{pmatrix},$$

$$\begin{aligned} (\xi^{<} \not{\circ} T) \circ T \circ \xi &= \begin{pmatrix} (Q^{<} \not{\circ} P) \circ P^{<} & 0 & 0 \\ 0 & Q^{<} & 0 \\ 0 & 0 & \mathrm{id} \end{pmatrix} \circ \begin{pmatrix} P \circ Q \\ Q \\ \mathrm{id} \end{pmatrix} \\ &= \begin{pmatrix} (Q^{<} \not{\circ} P) \circ P^{<} \circ P \circ Q \\ Q^{<} \circ Q \\ \mathrm{id} \end{pmatrix} \quad \text{(by Lemma 4.8)} \\ &= \begin{pmatrix} P \square Q \\ Q \\ \mathrm{id} \end{pmatrix} \quad \text{(by Proposition 3.3(c) and Definition 4.4)}. \end{aligned} \tag{5.15}$$

Finally, we obtain $\mathcal{I}(R)$. Obviously, there is no loop in a sequence (remember that the sequenced statements are considered atomic), so that R is progressively finite, that is, $\mathcal{I}(R) = \mathrm{id}$. But, just to illustrate the method, we show it.

Set $\mathcal{I}(R) = \left(\begin{smallmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{smallmatrix}\right)$ and aim at finding the monotypes a, b, and c by using Proposition 3.11(d). Then, we have

$$\begin{aligned} &\begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix} \\ &= \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix} \not{\circ} \begin{pmatrix} 0 & P & 0 \\ 0 & 0 & Q \\ 0 & 0 & 0 \end{pmatrix} \quad \text{(by Proposition 3.11(d))} \\ &= \begin{pmatrix} (a \not{\circ} 0) \circ (b \not{\circ} P) \circ (c \not{\circ} 0) & 0 & 0 \\ 0 & (a \not{\circ} 0) \circ (b \not{\circ} 0) \circ (c \not{\circ} Q) & 0 \\ 0 & 0 & (a \not{\circ} 0) \circ (b \not{\circ} 0) \circ (c \not{\circ} 0) \end{pmatrix} \\ &\qquad \text{(by Lemma 4.8)} \\ &= \begin{pmatrix} b \not{\circ} P & 0 & 0 \\ 0 & c \not{\circ} Q & 0 \\ 0 & 0 & I \end{pmatrix} \quad \text{(by Remark 5.2)}. \end{aligned} \tag{5.16}$$

The only solution is $a = b = c = \mathrm{id}$, then $\mathcal{I}(R) = \left(\begin{smallmatrix} \mathrm{id} & 0 & 0 \\ 0 & \mathrm{id} & 0 \\ 0 & 0 & \mathrm{id} \end{smallmatrix}\right)$, and, consequently, $\varepsilon^{\smile} \circ \mathcal{I}(R) = (\mathrm{id} \quad 0 \quad 0)$.

So, we find that

$$\mathcal{S}(\mathcal{R}) = \varepsilon^{\smile} \circ \mathcal{I}(R) \circ (\xi^{<} \not{\circ} T) \circ T \circ \xi = \begin{pmatrix} \mathrm{id} & 0 & 0 \end{pmatrix} \circ \begin{pmatrix} P \square Q \\ Q \\ \mathrm{id} \end{pmatrix} = P \square Q. \tag{5.17}$$

This is the demonic semantics of a program $\mathscr{R}$ with matrix R given by (5.6).
We now give the case of the guarded command of Dijkstra.

5.3. Guarded command of Dijkstra. In this section, we will trait the Guarded commands

$$R = \begin{pmatrix} 0 & p & q & 0 \\ 0 & 0 & 0 & P \\ 0 & 0 & 0 & Q \\ 0 & 0 & 0 & 0 \end{pmatrix}, \qquad \varepsilon = \begin{pmatrix} \mathrm{id} \\ 0 \\ 0 \\ 0 \end{pmatrix}, \qquad \xi = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \mathrm{id} \end{pmatrix}, \tag{5.18}$$

$$\xi^{<} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathrm{id} \end{pmatrix}, \qquad R^{*} = \begin{pmatrix} \mathrm{id} & p & q & p \circ P \vee q \circ Q \\ 0 & \mathrm{id} & 0 & P \\ 0 & 0 & \mathrm{id} & Q \\ 0 & 0 & 0 & \mathrm{id} \end{pmatrix}, \tag{5.19}$$

$$R^{\prec} = \begin{pmatrix} p^{\sim} \circ q^{\sim} & 0 & 0 & 0 \\ 0 & P^{\prec} & 0 & 0 \\ 0 & 0 & Q^{\prec} & 0 \\ 0 & 0 & 0 & \mathrm{id} \end{pmatrix}, \tag{5.20}$$

$$T = R^{*} \circ R^{\prec} = \begin{pmatrix} p^{\sim} \circ q^{\sim} & p \circ P^{\prec} & q \circ Q^{\prec} & p \circ P \vee q \circ Q \\ 0 & P^{\prec} & 0 & P \\ 0 & 0 & Q^{\prec} & Q \\ 0 & 0 & 0 & \mathrm{id} \end{pmatrix}. \tag{5.21}$$

By using the same rules as the sequence case and Remark 5.2, we have

$$\xi^{<} \backslash T = \begin{pmatrix} (0 \backslash p^{\sim} \circ q^{\sim}) \circ (0 \backslash p \circ P^{\prec}) \circ (0 \backslash q Q^{\prec}) & 0 & 0 & 0 & 0 \\ 0 & 0 \backslash P^{\prec} & 0 & 0 & 0 \\ 0 & 0 & 0 \backslash Q^{\prec} & 0 & \\ 0 & 0 & 0 & \mathrm{id} & \end{pmatrix}$$

(by Lemma 4.8)

$$= \begin{pmatrix} (0 \backslash (p \vee q)^{\sim}) \circ (P^{<} \backslash p) \circ (Q^{<} \backslash q) & 0 & 0 & 0 \\ 0 & P^{<} & 0 & 0 \\ 0 & 0 & Q^{<} & 0 \\ 0 & 0 & 0 & \mathrm{id} \end{pmatrix}$$

(by Proposition 3.6(a)) (5.22)

$$= \begin{pmatrix} (p \vee q) \wedge P^{<} \backslash p \wedge Q^{<} \backslash q & 0 & 0 & 0 \\ 0 & P^{<} & 0 & 0 \\ 0 & 0 & Q^{<} & 0 \\ 0 & 0 & 0 & \mathrm{id} \end{pmatrix}$$

(by the definition of the complement),

$$T \circ \xi = \begin{pmatrix} p \circ P \vee q \circ Q \\ P \\ Q \\ \mathrm{id} \end{pmatrix}.$$

So,

$$(\xi^< \not\backslash T) \circ T \circ \xi = \begin{pmatrix} [(p \vee q) \wedge P^< \not\backslash p \wedge Q^< \not\backslash q] \circ [p \circ P \vee q \circ Q] \\ P^< \circ P \\ Q^< \circ Q \\ \mathrm{id} \end{pmatrix}. \tag{5.23}$$

As the graph is an acyclic finite branching graph, so $\mathcal{I}(R) = \mathrm{id}$. We now simplify the above expression $((\xi^< \not\backslash T) \circ T \circ \xi)$:

$$\varepsilon^\sim \circ \mathcal{I}(R) \circ (\xi^< \not\backslash T) \circ T \circ \xi = [(p \vee q) \wedge P^< \not\backslash p \wedge Q^< \not\backslash q] \circ [p \circ P \vee q \circ Q]$$

$$[(p \vee q) \wedge P^< \not\backslash p \wedge Q^< \not\backslash q] \circ [p \circ P \vee q \circ Q]$$

$$= (p \vee q) \circ (P^< \vee p^\sim) \circ (Q^< \vee q^\sim) \circ (p \circ P \vee q \circ Q) \quad \text{(by (3.9))}$$

$$= (p \circ P^< \vee q \circ P^< \vee q \circ p^\sim) \circ (Q^< \circ p \circ P \vee q \circ Q \vee q^\sim \circ p \circ P)$$

(by Theorem 2.2(d), $p^\sim \circ p = 0$, $q^\sim \circ q = 0$, and Proposition 3.3(c))

$$= p \circ Q^< \circ P \vee p \circ q \circ P^< \circ Q \vee p \circ q^\sim \circ P \vee p \circ q \circ Q^< \circ P \vee q \circ P^< \circ Q \vee p^\sim \circ q \circ Q$$

(by Proposition 3.3(c), Proposition 3.1(b), $q \circ q^\sim = 0$, and $p \circ p^\sim = 0$)

$$= p \circ P \circ (Q^< \vee q^\sim) \vee p \circ q \circ (P^< \circ Q \vee Q^< \circ P) \vee q \circ Q \circ (P^< \vee p^\sim)$$

(by Theorem 2.2(d))

$$= p \circ P \circ (q \circ Q^< \vee q^\sim) \vee p \circ q \circ (P \sqcup Q) \vee q \circ Q \circ (p \circ P^< \vee p^\sim)$$

(by Proposition 4.2(a) and Proposition 3.1(g))

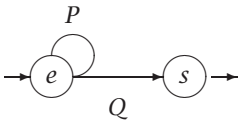
$$= p \circ q \circ (P \sqcup Q) \vee p \circ q^\sim \circ P \vee p^\sim \circ q \circ Q$$

(by Theorem 2.2(d), $p \circ q \circ P \circ Q^< \vee p \circ q \circ Q \circ P^< = p \circ q \circ (P \sqcup Q)$),

$$\mathcal{S}(\mathcal{R}) = p \circ q \circ (P \sqcup Q) \vee q^\sim \circ p \circ P \vee p^\sim \circ q \circ Q. \tag{5.24}$$

This is the demonic semantics of a program $\mathcal{R}$ with the matrix R of equation (5.18).

5.4. While loops. We will study a slightly more general case; its flow graph and associated matrices are



$$R = \begin{pmatrix} P & Q \\ 0 & 0 \end{pmatrix}, \qquad \varepsilon = \begin{pmatrix} \mathrm{id} \\ 0 \end{pmatrix}, \qquad \xi = \begin{pmatrix} 0 \\ \mathrm{id} \end{pmatrix}, \qquad \xi^< = \begin{pmatrix} 0 & 0 \\ 0 & \mathrm{id} \end{pmatrix}, \tag{5.25}$$

with the restriction

$$P^{\prec} \wedge Q^{\prec} = 0. \tag{5.26}$$

In what follows, we give the reflexive transitive closure of R by applying [27, Proposition (3.2.9)(ii)]:

$$\begin{gathered} R^* = \begin{pmatrix} P^* & P^* \circ Q \\ 0 & \mathrm{id} \end{pmatrix}, \qquad R^{\prec} = \begin{pmatrix} (P \vee Q)^{\prec} & 0 \\ 0 & \mathrm{id} \end{pmatrix}, \\ T = R^* \circ R^{\prec} = \begin{pmatrix} P^* \circ (P \vee Q)^{\prec} & P^* \circ Q \\ 0 & \mathrm{id} \end{pmatrix}. \end{gathered} \tag{5.27}$$

By applying Remark 5.2, we have

$$\begin{aligned} \xi^{\prec} \not\backslash T &= \begin{pmatrix} 0 \not\backslash (P^* \circ (P \vee Q)^{\prec}) & 0 \\ 0 & \mathrm{id} \end{pmatrix} \quad \text{(by Lemma 4.8)} \\ &= \begin{pmatrix} (P \vee Q)^{\prec} \not\backslash P^* & 0 \\ 0 & \mathrm{id} \end{pmatrix} \quad \text{(by Proposition 3.6(a) and Theorem 3.5(f) and (g))}, \\ T \circ \xi &= \begin{pmatrix} P^* \circ (P \vee Q)^{\prec} & P^* \circ Q \\ 0 & \mathrm{id} \end{pmatrix} \circ \begin{pmatrix} 0 \\ \mathrm{id} \end{pmatrix} = \begin{pmatrix} P^* \circ Q \\ \mathrm{id} \end{pmatrix}. \end{aligned} \tag{5.28}$$

So,

$$(\xi^{\prec} \not\backslash T) \circ T \circ \xi = \begin{pmatrix} [(P \vee Q)^{\prec} \not\backslash P^*] \circ P^* \circ Q \\ \mathrm{id} \end{pmatrix}. \tag{5.29}$$

Set $\mathcal{J}(R) = \left(\begin{smallmatrix} a & 0 \\ 0 & b \end{smallmatrix}\right)$ and aim at finding the monotypes a and b by using Proposition 3.11(d):

$$\begin{aligned} \mathcal{J}(R) = \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} &= \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} \not\backslash \begin{pmatrix} P & Q \\ 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} a \not\backslash P \wedge b \not\backslash Q & 0 \\ 0 & a \not\backslash 0 \wedge b \not\backslash 0 \end{pmatrix} = \begin{pmatrix} a \not\backslash P \wedge b \not\backslash Q & 0 \\ 0 & \mathrm{id} \end{pmatrix}. \end{aligned} \tag{5.30}$$

Clearly, the solution for b is $b = \mathrm{id}$. This leaves $a = a \not\backslash P$ to be solved. Now, $\mathcal{J}(P)$ is the least monotype c satisfying $c = c \not\backslash P$ (Definition 3.8). Hence, we must choose the least a satisfying $a = a \not\backslash P$; thus, again by Definition 3.8, $a = \mathcal{J}(P)$, then $\mathcal{J}(R) = \left(\begin{smallmatrix} \mathcal{J}(P) & 0 \\ 0 & \mathrm{id} \end{smallmatrix}\right)$:

$$\begin{aligned} \mathcal{S}(R) &= \varepsilon^{\smile} \circ \mathcal{J}(R) (\xi^{\prec} \not\backslash T) \circ T \circ \xi, \\ \varepsilon^{\smile} \circ \mathcal{J}(R) &= \begin{pmatrix} \mathrm{id} & 0 \end{pmatrix} \circ \begin{pmatrix} \mathcal{J}(P) & 0 \\ 0 & \mathrm{id} \end{pmatrix} = \begin{pmatrix} \mathcal{J}(P) & 0 \end{pmatrix}. \end{aligned} \tag{5.31}$$

Then,

$$\begin{aligned}\mathcal{S}(\mathcal{R}) &= \begin{pmatrix}\mathcal{I}(P) & 0\end{pmatrix} \circ \begin{pmatrix}\left[(P \vee Q)^{<} \not\backslash P^*\right] \circ P^* \circ Q \\ \mathrm{id}\end{pmatrix} \\ &= \mathcal{I}(P) \circ \left[(P \vee Q)^{<} \not\backslash P^*\right] \circ P^* \circ Q, \\ \mathcal{S}(\mathcal{R}) &= \mathcal{I}(P) \circ \left[(P \vee Q)^{<} \not\backslash P^*\right] \circ P^* \circ Q.\end{aligned} \tag{5.32}$$

This is the demonic semantics of a program $\mathcal{R}$ with the matrix R of equations (5.25).

6. Concluding remarks. We have presented certain notions concerning relation algebra and the refinement order in Sections 2 and 4.

We have shown how to give a generic demonic semantic definition (equation (5.8)) of programming constructs, based on the concept of relational flow diagram [26, 27]. Then, we have proved that for the sequence, the guarded command of Dijkstra, and the while loop, this definition is equivalent to traditional definitions (e.g., equation (5.17) in the case of the loop), usually given on a construct by construct basis. Along the road, we have derived many interesting intermediate results. Using this approach, it is easy to derive the semantics of other statements, such as the conditional or the guarded choice.

We close this note by a word on the proof style that has been employed. This research had originally been carried out by the author [28] in the same framework of the binary homogeneous relations, where the proofs made intensive use of vectors and complements. Other researchers [6] advocated a different proof style based on *monotypes* and *residuals*. We have shown in the present note that all the results can be generalized by using the monotypes and the residuals introduced by [16].

The approach to demonic input-output relation presented here is not the only possible one. In [20, 21, 22], the infinite looping has been treated by adding to the state space a fictitious state $\perp$ to denote nontermination. In [9, 19, 23, 25], the demonic input-output relation is given as a pair (relation, set). The relation describes the input-output behavior of the program, whereas the set component represents the domain of guaranteed termination.

We note that the preponderant formalism employed until now for the description of demonic input-output relation is the wp-calculus. For more details see [2, 3, 17].

References

[1] C. Arts, *Galois connections presented calculationally*, Research report, Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands, 1992.

[2] R. J. R. Back, *On the correctness of refinement in program development*, Ph.D. thesis, Department of Computer Science, University of Helsinki, Finland, 1978.

[3] R. J. R. Back and J. von Wright, *Combining angels, demons and miracles in program specifications*, Theoret. Comput. Sci. **100** (1992), no. 2, 365–383.

[4] R. C. Backhouse and H. Doombos, *Mathematical induction made calculational*, Computing Science Note 94/16, Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands, 1994.

[5] R. C. Backhouse, P. Hoogendijk, E. Voermans, and J. van der Woude, *A relational theory of datatypes*, Research report, Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands, 1992.

[6] R. C. Backhouse and J. van der Woude, *Demonic operators and monotype factors*, Math. Structures Comput. Sci. **3** (1993), no. 4, 417–433.

[7] R. Berghammer, *Relational specification of data types and programs*, Tech. Report 9109, Fakultät für Informatik, Universität der Bundeswehr München, Germany, 1991.

[8] R. Berghammer and G. Schmidt, *Relational specifications*, Algebraic Methods in Logic and in Computer Science (Warsaw, 1991) (C. Rauszer, ed.), Banach Center Publ., vol. 28, Polish Academy of Sciences, Warsaw, 1993, pp. 167–190.

[9] R. Berghammer and H. Zierer, *Relational algebraic semantics of deterministic and nondeterministic programs*, Theoret. Comput. Sci. **43** (1986), no. 2-3, 123–147.

[10] N. Boudriga, F. Elloumi, and A. Mili, *On the lattice of specifications: applications to a specification methodology*, Formal Aspects of Computing **4** (1992), no. 6, 544–571.

[11] L. H. Chin and A. Tarski, *Distributive and modular laws in the arithmetic of relation algebras*, Univ. California Publ. Math. (N.S.) **1** (1951), 341–384.

[12] J. H. Conway, *Regular Algebra and Finite Machines*, Chapman and Hall, London, 1971.

[13] B. A. Davey and H. A. Priestley, *Introduction to Lattices and Order*, Cambridge Mathematical Textbooks, Cambridge University Press, Cambridge, 1990.

[14] J. Desharnais, N. Belkhiter, S. Ben Mohamed Sghaier, F. Tchier, A. Jaoua, A. Mili, and N. Zaguia, *Embedding a demonic semilattice in a relation algebra*, Theoret. Comput. Sci. **149** (1995), no. 2, 333–360.

[15] J. Desharnais, A. Jaoua, F. Mili, N. Boudriga, and A. Mili, *A relational division operator: the conjugate kernel*, Theoret. Comput. Sci. **114** (1993), no. 2, 247–272.

[16] J. Desharnais, B. Möller, and F. Tchier, *Kleene under a demonic star*, 8th International Conference on Algebraic Methodology and Software Technology (AMAST 2000), Lecture Notes in Computer Science, vol. 1816, Springer-Verlag, Iowa, 2000, pp. 355–370.

[17] E. W. Dijkstra, *A Discipline of Programming*, Prentice-Hall Series in Automatic Computation, Prentice-Hall, New Jersey, 1976.

[18] R. P. Dilworth, *Non-commutative residuated lattices*, Trans. Amer. Math. Soc. **46** (1939), 426–444.

[19] H. Doornbos, *A relational model of programs without the restriction to Egli-Milner-monotone constructs*, Programming Concepts, Methods and Calculi (San Miniato, 1994), IFIP Trans. A Comput. Sci. Tech., vol. A-56, North-Holland Publishing, Amsterdam, 1994, pp. 363–382.

[20] C. A. R. Hoare, I. J. Hayes, J. He, C. C. Morgan, A. W. Roscoe, J. W. Sanders, I. H. Sorensen, J. M. Spivey, and B. A. Sufrin, *Laws of programming*, Comm. ACM **30** (1987), no. 8, 672–686.

[21] C. A. R. Hoare and J. He, *The weakest prespecification. I*, Fund. Inform. **9** (1986), no. 1, 51–84.

[22] ———, *The weakest prespecification. II*, Fund. Inform. **9** (1986), no. 2, 217–251.

[23] R. D. Maddux, *Relation-algebraic semantics*, Theoret. Comput. Sci. **160** (1996), no. 1-2, 1–85.

[24] A. Mili, J. Desharnais, and F. Mili, *Relational heuristics for the design of deterministic programs*, Acta Inform. **24** (1987), no. 3, 239–276.

[25] D. L. Parnas, *A generalized control structure and its formal definition*, Comm. ACM **26** (1983), 572–581.

[26] G. Schmidt, *Programs as partial graphs. I. Flow equivalence and correctness*, Theoret. Comput. Sci. **15** (1981), no. 1, 1–25.

[27] G. Schmidt and T. Ströhlein, *Relations and Graphs*, EATCS Monographs on Theoretical Computer Science, Springer-Verlag, Berlin, 1993.

[28] F. Tchier, *Sémantiques relationnelles démoniaques et vérification de boucles non déterministes*, Thèse de doctorat, Département de Mathématiques et de statistique, Université Laval, Canada, 1996.

F. Tchier: Mathematics Department, King Saud University, P.O. Box 22452, Riyadh 11495, Saudi Arabia

E-mail address: ftchier@hotmail.com