APPLICATIONS OF RELATIONAL CALCULUS TO COMPUTER MATHEMATICS

Kawahara, Yasuo
Department of Mathematics, Kyushu University

APPLICATIONS OF RELATIONAL CALCULUS TO COMPUTER MATHEMATICS

By

Yasuo KAWAHARA*

Abstract

Relational calculus is applicable to some aspects of computer mathematics such as representing non-determinism of automata theory and theory of algorithms. In this note the author gives other proofs of the fundamental properties of languages received by finite automata and a fixed point semantics on simple recursive programs by using relational calculus.

1. Introduction

A relation (correspondence) is a many-valued function generalizing ordinary total functions. Relational calculus based on theory of (binary) relations might be regarded as a field of mathematics investigating applications of theory of relations as well as itself. Relational calculus has been effectively applied to many situations of pure mathematics, in particular to constructing spectral sequences and to proving systematically the properties on various kinds of exact sequences in topology and homological algebra.

In computer mathematics it has been used in representing notions of non-determinism of automata, theory of algorithms, semantics of abstract programs and so on.

The aim of this note is to illustrate the power of relational calculus for formulating familiar ideas in computer mathematics. A several systematic and positive applications of relational calculus to computer mathematics are given, although they are not new results at all.

At first we will give elegant proofs of the elementary properties of regular languages by constructing simple and natural non-deterministic automata within relational calculus.

The final example is concerned on while-do sentences, a kind of recursive programs. An (operational) semantics of while-do sentences is stated by using union of iterations of a relation and its fixed point properties are neatly formulated. But this formulation suggests a new problem on relational calculus. At the end of note we solve it in the case of Heyting valued relations.

---

* Department of Mathematics, Kyushu University 33, Fukuoka 812, Japan.

2. Fundamentals of Relational Calculus

In this section we state the fundamental properites of (binary) relations without proofs which will be necessary later.

A *relation* α of a set A to another set B, denoted by a half arrow $\alpha : A \to B$, is a subset of the cartesian product $A \times B$. We usually identify a mapping (total function) $f : A \to B$ with a relation

$$\{(a, af) ; a \in A\} \qquad (the\ graph\ of\ f).$$

Thus the *identity mapping* id_A of a set A is the diagonal set $\{(a, a) ; a \in A\}$. For two relations $\alpha : A \to B$ and $\beta : B \to C$, the composite $\alpha\beta(=\alpha \cdot \beta) : A \to C$ of α followed by β, and the reverse $\alpha^\# : B \to A$ of α are defined by

$$\alpha\beta = \{(a, c) \in A \times C ; (a, b) \in \alpha \ and \ (b, c) \in \beta \ for \ some \ b \in B\}$$

and

$$\alpha^\# = \{(b, a) \in B \times A ; (a, b) \in \alpha\},$$

respectively. The set of all relations of A to B is denoted by $Rel(A, B)$. We now state the basic properties of relations.

2.1. [*Boolean algebra*] $Rel(A, B)$ *is a Boolean algebra for any sets A and B. That is, $Rel(A, B)$ is a partially ordered set by inclusion $\subseteq$ of subsets and has so-called Boolean operations-union $\alpha \cup \beta$, intersection $\alpha \cap \beta$, and complement $\neg \alpha$ for $\alpha, \beta : A \to B$, and moreover it has the minimum element $\theta_{A,B}$ (the empty relation ϕ), and the maximum element $\Theta_{A,B}$ (the total relation $A \times B$).* $\square$

The following statement asserts that all the set and all the relations between sets form *an I-category* **Rel** in the sense of D. Puppe [9].

2.2. [*I-category*] *For relations $\alpha, \alpha' : A \to B$, $\beta, \beta' : B \to C$ and $\gamma : C \to D$ the following laws are valid:*

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

$$id_A \cdot \alpha = \alpha \cdot id_B = \alpha \qquad (identity),$$

$$\alpha^{\#\#} = \alpha, \quad (\alpha\beta)^\# = \beta^\# \alpha^\# \qquad (involutive),$$

and if $\alpha \subseteq \alpha'$ and $\beta \subseteq \beta'$, then $\alpha\beta \subseteq \alpha'\beta'$, $\alpha^\# \subseteq \alpha'^\#$ (monotone). $\square$

Let $P(B)$ be the *power set* of a set B, that is, the set of all subsets of B. The *membership relation* $\in_B : P(B) \to B$ of B is defined by

$$\in_B = \{(S, b) \in P(B) \times B ; b \in S\},$$

which has the following important property:

2.3. [*Power adjunction*] *For each relation $\alpha : A \to B$, there exists a unique mapping $f : A \to P(B)$ with $\alpha = f \in_B$. In fact $af = \{b \in B ; (a, b) \in \alpha\}$ for $a \in A$.* $\square$

A singleton set I is a terminal object in the category **Set** of all the sets and all the mappings between sets.

2.4. [*Terminability*] *For two sets A and B, $\Theta_{A,1} \cdot \Theta_{1,B} = \Theta_{A,B}$ and $\Theta_{I,I} = id_I$ hold. Moreover, the identity id_I of I is a unique relation from I to I diferent from the empty*

relation $\theta_{I,I}$. $\square$

By the last property the following statements on relations are valid:

(2.4a) For relations $\alpha, \alpha' : I \to I$, $\alpha\alpha'=id_I$ (or $\alpha \cap \alpha'=id_I$) if and only if $\alpha=id_I$ and $\alpha'=id_I$.

(2.4b) For a family $\{\alpha_k : I \to I ; k \in K\}$ of relations, $\cup_k \alpha_k=id_I$ if and only if there is $k \in K$ such that $\alpha_k=id_I$. $\square$

The following distributive law of compositions over union of relations is an indispensable tool in computing relations. It is known that the distributive law is also valid for the case of relations in toposes with suitable coproducts.

2.5. [*Distributivity*] *For a relation* $\alpha : A \to B$ *and a family* $\{\beta_k : B \to C ; k \in K\}$ *of relations, the distributive law* $\alpha(\cup_k \beta_k)=\cup_k \alpha\beta_k$ *is valid.* $\square$

2.6. We denote the disjoint union (coproduct) of two sets Q and Q' by $Q+Q'$, and the inclusion maps by $i : Q \to Q+Q'$ and $j : Q' \to Q+Q'$, respectively. (An arrow $\to$ denotes a total function.) The following identities are elementary and very important in relational calculus:

$$ij^*=\theta_{Q,Q'}, \quad ji^*=\theta_{Q',Q}, \quad ii^*=id_Q, \quad jj^*=id_{Q'}, \quad ii^*\cup jj^*=id_{Q+Q'}.$$

(Note that θ denotes the empty relation.) $\square$

2.7. Let $X \times Y$ be the cartesian product of sets X and Y, and $p : X \times Y \to X$, $q : X \times Y \to Y$ the projections respectively. If $\alpha : A \to X$, $\beta : A \to Y$, $\mu : X \to B$ and $\nu : Y \to B$ are relations, then

(2.7a) $$(\alpha p^* \cap Bq^*)(p\mu \cap q\nu)=\alpha\mu \cap \beta\nu$$

is valid.

Moreover, assume that $p' : X' \times Y' \to X'$, $q' : X' \times Y' \to Y'$, $p'' : X'' \times Y'' \to X''$ and $q'' : X'' \times Y'' \to Y''$ are projections of cartesian products, respectively. For relations $\alpha : X \to X'$ and $\beta : Y \to Y'$, define a relation $\alpha \times \beta : X \times Y \to X' \times Y'$ by

$$\alpha \times \beta=p\alpha p'^* \cap q\beta q'^*.$$

If $\alpha : X \to X'$, $\beta : Y \to Y'$, $\alpha' : X' \to X''$ and $\beta' : Y' \to Y''$ are relations, then

(2.7b) $$(\alpha \times \beta)(\alpha' \times \beta')=\alpha\alpha' \times \beta\beta'$$

is valid.

The last property (2.7b) is easily follows from

$$(\alpha \times \beta)(\alpha' \times \beta')=(p\alpha p'^* \cap q\beta q'^*)(p'\alpha' p''^* \cap q'\beta' q''^*)$$
$$=p\alpha\alpha' p''^* \cap q\beta\beta' q''^*$$
$$=\alpha\alpha' \times \beta\beta',$$

applying (2.7a).

3. Applications to Elementary Theory of Automata

In this section we give some applications of relational calculus to elementary theory of (non-deterministic) automata. At first we define (non-deterministic) automata from our stand point of relational calculus and prove some elementary properties of regular languages by constructing simple and natural (non-deterministic) automata within relational calculus.

Let Σ be an alphabet (a fixed finite set). In a view point of relational calculus, a (non-deterministic) *automaton* M is defined to be a quadruple $M=(Q, \{\delta(\sigma): Q \to Q ; \sigma \in \Sigma\},$ $\tau: I \to Q, \beta: Q \to I)$, where

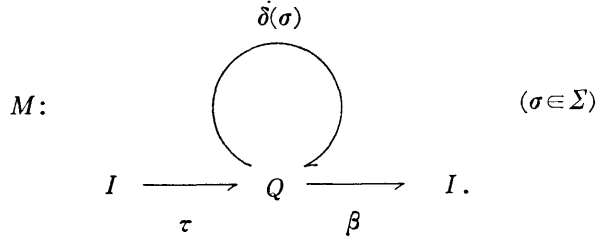> Q is the state set;
>
> $\{\delta(\sigma): Q \to Q ; \sigma \in \Sigma\}$, the set of state-transition relations;
>
> $\tau: I \to Q$, the initial state relation;

and $\beta: Q \to I$, the final state relation.

A automaton M is usefully represented by the following figure:

For an input string $w=\sigma_1\sigma_2 \cdots \sigma_n \in \Sigma^*$ ($\sigma_i \in \Sigma$, $1 \leq i \leq n$, $0 < n$), the iterative state-trasition relation $\delta(w): Q \to Q$ of M is defined by $\delta(w)=\delta(\sigma_1) \cdot \delta(\sigma_2) \cdot \cdots \cdot \delta(\sigma_n)$. For a null string $w=\lambda$, $\delta(\lambda)$ is the identity relation (function) id_Q. Moreover the language $T(M)$ accepted by M (the behavior of M) is defined by

$$T(M)=\{w \in \Sigma^* ; \tau\delta(w)\beta=id_I\},$$

which is the set of input strings over Σ accepted by M.

3.1. For a given automaton $M=(Q, \{\delta(\sigma): Q \to Q ; \sigma \in \Sigma\}, \tau: I \to Q, \beta: Q \to I)$, consider an automaton $M^R=(Q, \{\delta^R(\sigma)=\delta(\sigma)^\sharp: Q \to Q ; \sigma \in \Sigma\}, \tau^R=\beta^\sharp: I \to Q, \beta^R=\tau^\sharp: Q \to I)$:
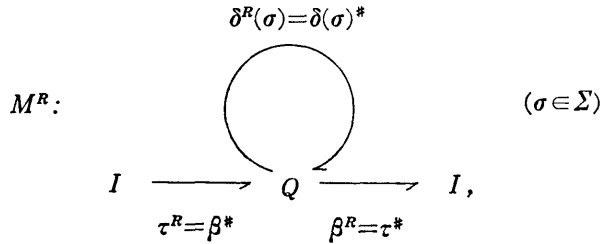
where the symbol $\sharp$ denotes inverse relations (involution). Then it directly follows

from the property 2.2 that for an input string $w=\sigma_1\sigma_2\cdots\sigma_n\in\Sigma^*$ ($\sigma_i\in\Sigma$, $1\leqq i\leqq n$, $0<n$)
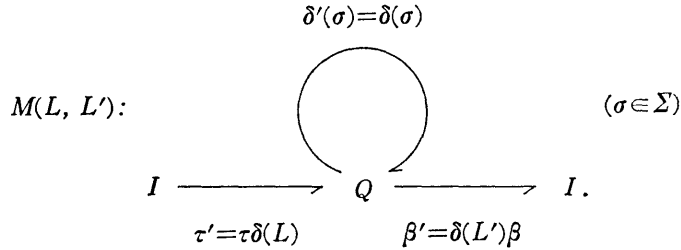
$$\delta^R(w)=\delta^R(\sigma_1)\cdot\delta^R(\sigma_2)\cdot\cdots\cdot\delta^R(\sigma^u)$$

$$=\delta(\sigma_1)^\#\cdot\delta(\sigma_2)^\#\cdot\cdots\cdot\delta(\sigma_n)^\#$$

$$=(\delta(\sigma_n)\cdot\cdots\cdot\delta(\sigma_2)\cdot\delta(\sigma_1))^\#$$

$$=(\delta(w^R))^\#$$

and $\delta^R(\lambda)=id_Q=(\delta(\lambda))^\#$. (Note that we use the superscript symbol R for expressing reversed strings.) Hence $\tau^R\delta^R(w)\beta^R=(\tau\delta(w^R)\beta)^\#$ for all input strings $w\in\Sigma^*$ and so $\tau^R\delta^R(w)\beta^R=id_I$ if and only if $\tau\delta(w^R)\beta=id_I$ for all $w\in\Sigma^*$, which shows

$$T(M^R)=\{w\in\Sigma^*;\ w^R\in T(M)\}$$

$$=\{w^R\in\Sigma^*;\ w\in T(M)\}\qquad(\text{since }w^{RR}=w)$$

$$=\{T(M)\}^R.$$

Since M^R is also finite if M is finite, this indicates a well-known fact that if L is a regular language over Σ, then the reversed language L^R of L is also regular language over Σ. $\quad\square$

3.2. Let $M=(Q, \{\delta(\sigma): Q\to Q;\ \sigma\in\Sigma\}, \tau: I\to Q, \beta: Q\to I)$ be an automaton and L, L' languages over Σ. We define $\delta(L): Q\to Q$ by $\delta(L)=\cup\{\delta(u);\ u\in L\}$ and consider an aumaton $M(L, L')=(Q, \{\delta'(\sigma)=\delta(\sigma): Q\to Q;\ \sigma\in\Sigma\}, \tau'=\tau\delta(L): I\to Q, \beta'=\delta(L')\beta: Q\to I)$:

$$\delta'(\sigma)=\delta(\sigma)$$

$M(L, L'):$

$(\sigma\in\Sigma)$

$$I\longrightarrow Q\longrightarrow I.$$

$$\tau'=\tau\delta(L)\qquad\beta'=\delta(L')\beta$$

Then it is trivial from the distibutive law 2.5 that for any $w\in\Sigma^*$

$$\tau'\delta'(w)\beta'=\tau(\cup\{\delta(u);\ u\in L\})\delta(w)(\cup\{\delta(v);\ v\in L'\})\beta$$

$$=\cup\{\tau\delta(uwv)\beta;\ u\in L, v\in L'\}$$

and hence, by the terminability 2.4 of relations, $\tau'\delta'(w)\beta'=id_I$ if and only if $\tau\delta(uwv)\beta=id_I$ for some $u\in L$, $v\in L'$. Therefore we have

$$T(M(L, L'))=\{w\in\Sigma^*;\ uwv\in T(M)\ for\ some\ u\in L\ and\ v\in L'\}$$

$$=L^{-1}\cdot T(M)\cdot L'^{-1}.$$

This indicates a well-known fact that a derived language of a regular language by any languages are also regular. In particular we have obtained
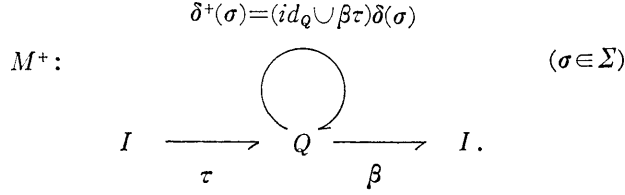
$$T(M(u, v))=u^{-1}\cdot T(M)\cdot v^{-1},$$

$$T(M(\lambda, \Sigma^*))=T(M)\cdot\Sigma^{*-1}=Init\{T(M)\},$$

and
$$T(M(\Sigma^*, \lambda))=\Sigma^{*-1}\cdot T(M)=Fin\{T(M)\},$$

$$T(M(\Sigma^*, \Sigma^*))=\Sigma^{*-1}\cdot T(M)\cdot\Sigma^{*-1}=Sub\{T(M)\},$$

where for a language L over Σ, $Init\{L\}$, $Fin\{L\}$, and $Sub\{L\}$ are the set of all initial segments of strings of L, the set of all final segments of strings of L, and the set of all subsegments of strings of L, respectively.   □

3.3. Let $M=(Q, \{\delta(\sigma):Q\to Q\ ;\ \sigma\in\Sigma\}, \tau:I\to Q, \beta:Q\to I)$ be an automaton and consider the following aumaton $M^+=(Q, \{\delta^+(\sigma)=(id_Q\cup\beta\tau)\delta(\sigma):Q\to Q\ ;\ \sigma\in\Sigma\}, \tau:I\to Q, \beta:Q\to I)$:

$$\delta^+(\sigma)=(id_Q\cup\beta\tau)\delta(\sigma)$$

M^+: $(\sigma\in\Sigma)$

$$I\ \xrightarrow{\ \ \tau\ \ }\ Q\ \xrightarrow{\ \ \beta\ \ }\ I\,.$$

Noticing that $\tau(id_Q\cup\beta\tau)=(id_Q\cup\tau\beta)\tau=\tau$ since $\tau\beta\subseteq id_I$ (by the terminability 2.4 of relations), the distributive law 2.5 implies that for a string $w=\sigma_1\sigma_2\cdots\sigma_n\in\Sigma^*$ ($\sigma_i\in\Sigma$, $1\le i\le n$, $0<n$)

$$\tau\delta^+(w)\beta=\tau\delta(\sigma_1)(id_Q\cup\beta\tau)\delta(\sigma_2)\cdot\cdots\cdot(id_Q\cup\beta\tau)\delta(\sigma_n)\beta$$

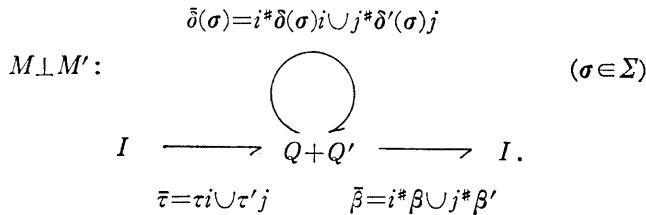$$=\cup\{(\tau\delta(u_1)\beta)\cdot\cdots\cdot(\tau\delta(u_k)\beta)\ ;\ w=u_1\cdots u_k,\ u_i\ne\lambda,\ 0<k\}\,.$$

It is obvious that $\tau\delta^+(\lambda)\beta=\tau\cdot id_Q\cdot\beta=\tau\delta(\lambda)\beta$.   Hence for any string $w\in\Sigma^+$, $\tau\delta^+(w)\beta=id_I$ if and only if there exist $u_1, \cdots, u_k\in\Sigma^+$ ($0<k$) such that $w=u_1\cdots u_k$ and $\tau\delta(u_s)\beta=id_I$ ($s=1, \cdots, k$) by the terminability 2.4a(a), (b).   Also it is trivial that $\lambda\in T(M^+)$ if and only if $\lambda\in T(M)$.   Consequently we have proved

$$T(M^+)=\{T(M)\}^+\,.$$

This indicates a well-known fact that if a language L over Σ is accepted by an automaton, then L^+ is also accepted by an automaton.   □

REMARK.   In the case that $\delta^+(\sigma)=\delta(\sigma)(1_Q\cup\beta\tau)$ or $\delta^+(\sigma)=\delta(\sigma)\cup\beta\tau\delta(\sigma)\cup\delta(\sigma)\beta\tau$ is adopted, the same result is obtained.   But if we put $\delta^+(\sigma)=\delta(\sigma)\cup(\beta\tau\delta(\sigma)\beta\tau)$ then the result is different.

3.4. For given automata $M=(Q, \{\delta(\sigma):Q\to Q\ ;\ \sigma\in\Sigma\}, \tau:I\to Q, \beta:Q\to I)$ and $M'=(Q', \{\delta'(\sigma):Q'\to Q'\ ;\ \sigma\in\Sigma\}, \tau':I\to Q', \beta':Q'\to I)$, we define an automaton $M\perp M'$ by the following figure:

$$\bar\delta(\sigma)=i^*\delta(\sigma)i\cup j^*\delta'(\sigma)j$$

$M\perp M'$: $(\sigma\in\Sigma)$

$$I\ \xrightarrow{\ \ \ \ }\ Q+Q'\ \xrightarrow{\ \ \ \ }\ I\,.$$

$$\bar\tau=\tau i\cup\tau'j\qquad\qquad\bar\beta=i^*\beta\cup j^*\beta'$$
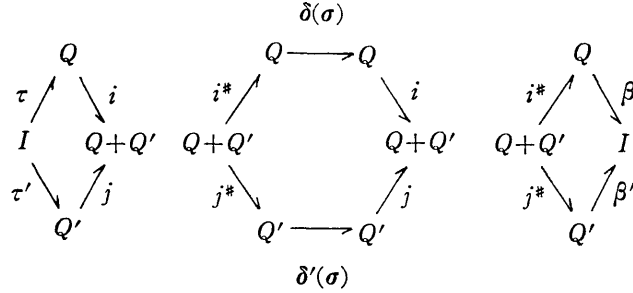
A routine induction on the length of w proves that for any string $w \in \Sigma^*$

$$\bar{\delta}(w) = i^\# \delta(w) i \cup j^\# \delta'(w) j,$$

$$\bar{\tau}\bar{\delta}(w)\bar{\beta} = (\tau i \cup \tau' j)(i^\# \delta(w) i \cup j^\# \delta'(w) j)(i^\# \beta \cup j^\# \beta')$$
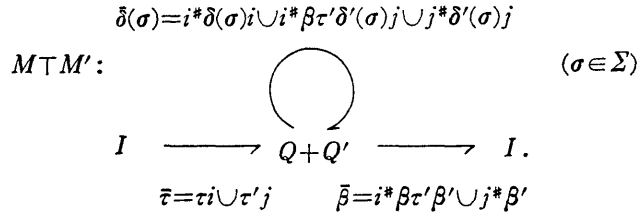
$$= \tau \delta(w)\beta \cup \tau'\delta'(w)\beta'.$$

Thus for any $w \in \Sigma^*$, $\bar{\tau}\bar{\delta}(w)\bar{\beta} = id_I$ if and only if $\tau\delta(w)\beta = id_I$ or $\tau'\delta'(w)\beta' = id_I$ (by the terminability 2.4(b)), which concludes

$$T(M \perp M') = T(M) \cup T(M').$$

This indicates a fact that if there exist automata M, M' accepting languages L, L', respectively, then there exists an automaton accepting union $L \cup L'$ of L and L'. Finally we remark that the state-transition of $M \perp M'$ is well illustrated by the following diagrams:

3.5. For given automata $M = (Q, \{\delta(\sigma): Q \to Q ; \sigma \in \Sigma\}, \tau: I \to Q, \beta: Q \to I)$ and $M' = (Q', \{\delta'(\sigma): Q' \to Q' ; \sigma \in \Sigma\}, \tau': I \to Q', \beta': Q' \to I)$, we define an automaton $M \top M'$ by the following figure:

$$\bar{\delta}(\sigma) = i^\# \delta(\sigma) i \cup i^\# \beta \tau' \delta'(\sigma) j \cup j^\# \delta'(\sigma) j$$

$$\bar{\tau} = \tau i \cup \tau' j \qquad \bar{\beta} = i^\# \beta \tau' \beta' \cup j^\# \beta'$$

A routine induction of the length of w proves that for any string $w \in \Sigma^*$

$$\bar{\delta}(w) = (\cup\{i^\# \delta(u)\beta\tau'\delta'(v)j ; w = uv, v \neq \lambda\}) \cup i^\# \delta(w)i \cup j^\# \delta'(w)^\# j,$$

$$\bar{\tau}\bar{\delta}(w)\bar{\beta} = \cup\{(\tau\delta(u)\beta)(\tau'\delta'(v)\beta') ; w = uv\}.$$

Thus for any $w \in \Sigma^*$, $\bar{\tau}\bar{\delta}(w)\bar{\beta} = id_I$ if and only if $\tau\delta(u)\beta = id_I$ and $\tau'\delta'(v)\beta = id_I$ for some strings u, v with $w = uv$, which proves that
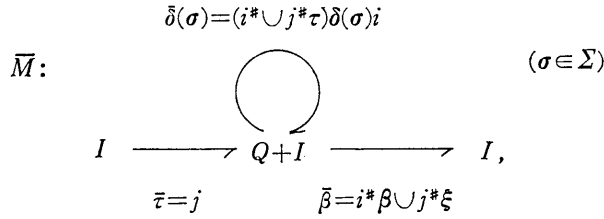
$$T(M \top M') = T(M) \cdot T(M').$$

(Note that we use the symbol $\cdot$ for concatenation of languages.) This indicates a fact that if there exist automata M, M' accepting languages L, L', respectively, then there exists a automaton accepting concatenation $L \cdot L'$ of L and L'. Finally we remark that

the state-transition of $M\top M'$ is well illustrated by the following diagrams:

$$\delta(\sigma)$$

$$\delta'(\sigma)$$
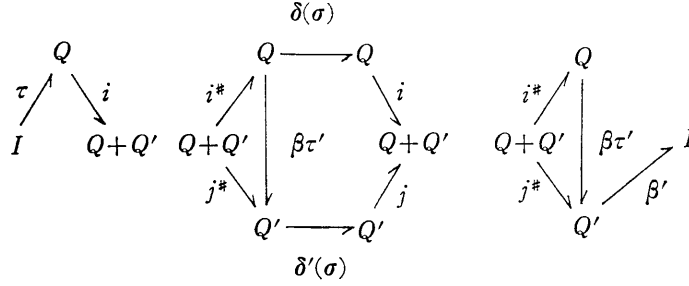
3.6. For a given automaton $M=(Q, \{\delta(\sigma): Q\to Q\,;\ \sigma\in\Sigma\}, \tau: I\to Q, \beta: Q\to I)$ and a given relation $\xi: I\to I$, we define an automaton $\overline{M}$ by the following figure:

$$\bar\delta(\sigma)=(i^{\#}\cup j^{\#}\tau)\delta(\sigma)i$$

$$\overline{M}:\qquad\qquad\bigcirc\qquad\qquad(\sigma\in\Sigma)$$

$$I\ \xrightarrow{\quad\quad}\ Q+I\ \xrightarrow{\quad\quad}\ I\,,$$
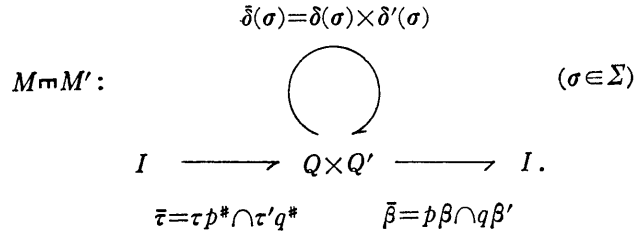
$$\bar\tau=j\qquad\qquad\bar\beta=i^{\#}\beta\cup j^{\#}\xi$$

where $i: Q\to Q+I$ and $j: I\to Q+I$ are the inclusions of a disjoint union $Q+I$. It follows from the elementary identities 2.6 on the inclusions of disjoint unions that for a string $w=\sigma_1\sigma_2\cdots\sigma_n\in\Sigma^*$ $(\sigma_i\in\Sigma,\ 1\leq i\leq n,\ 0<n)$

$$\bar\delta(w)=(i^{\#}\cup j^{\#}\tau)\delta(\sigma_1)i(i^{\#}\cup j^{\#}\tau)\delta(\sigma_2)i\cdot\cdots\cdot(i^{\#}\cup j^{\#}\tau)\delta(\sigma_n)i$$

$$=(i^{\#}\cup j^{\#}\tau)\delta(w)i\,,$$

$$\bar\tau\bar\delta(w)\bar\beta=j(i^{\#}\cup j^{\#}\tau)\delta(w)i(i^{\#}\beta\cup j^{\#}\xi)$$

$$=\tau\delta(w)\beta\,,$$

and $\bar\tau\bar\delta(\lambda)\bar\beta=j\cdot id_Q\cdot(i^{\#}\beta\cup j^{\#}\xi)=\xi$. Hence we have $T(\overline{M})=T(M)\cup\{\lambda\}$ if $\xi=id_I$, and otherwise $T(\overline{M})=T(M)-\{\lambda\}$. $\quad\square$

3.7. For given automata $M=(Q, \{\delta(\sigma): Q\to Q\,;\ \sigma\in\Sigma\}, \tau: I\to Q, \beta: Q\to I)$ and $M'=(Q', \{\delta'(\sigma): Q'\to Q'\,;\ \sigma\in\Sigma\}, \tau': I\to Q', \beta': Q'\to I)$, we define an automaton $M\sqcap M'$ by the following figure:

$$\bar\delta(\sigma)=\delta(\sigma)\times\delta'(\sigma)$$

$$M\sqcap M':\qquad\qquad\bigcirc\qquad\qquad(\sigma\in\Sigma)$$

$$I\ \xrightarrow{\quad\quad}\ Q\times Q'\ \xrightarrow{\quad\quad}\ I\,.$$

$$\bar\tau=\tau p^{\#}\cap\tau'q^{\#}\qquad\bar\beta=p\beta\cap q\beta'$$

Then for any $w\in\Sigma^*$ it follows from $\bar\delta(w)=\delta(w)\times\delta'(w)$ that

$$\bar{\tau}\bar{\delta}(w)\bar{\beta}=(\tau p^{\#}\cap\tau'q^{\#})(\delta(w)\times\delta'(w))(p\beta\cap q\beta')$$

$$=\tau\delta(w)\beta\cap\tau'\delta'(w)\beta' \qquad \text{(by 2.7(a), (b)).}$$

Hence for any $w\in\Sigma^{*}$, $\bar{\tau}\bar{\delta}(w)\bar{\beta}=id_{I}$ if and only if $\tau\delta(w)\beta=id_{I}$ and $\tau'\delta'(w)\beta'=id_{I}$, that is, this proves

$$T(M\sqcap M')=T(M)\cap T(M).$$

This indicates a fact that if there exist automata M, M' accepting languages L, L', respectively, then there exists a automaton accepting intersection $L\cap L'$ of L and L'. □

3.8. For given automata $M=(Q, \{\delta(\sigma):Q\to Q\ ;\ \sigma\in\Sigma\}, \tau:I\to Q, \beta:Q\to I)$ and $M'=(Q', \{\delta'(\sigma):Q'\to Q'\ ;\ \sigma\in\Sigma\}, \tau':I\to Q', \beta':Q'\to I)$, we define an automaton $M\sqcup M'$ by the following figure:

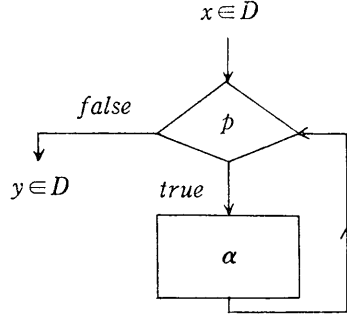$$\bar{\delta}(\sigma)=(\delta(\sigma)\times id_{Q'})\cup(id_{Q}\times\delta'(\sigma))$$

$M\sqcup M':$ $\hspace{3cm}$ $(\sigma\in\Sigma)$

$$I \longrightarrow Q\times Q' \longrightarrow I.$$

$$\bar{\tau}=\tau p^{\#}\cap\tau'q^{\#} \qquad \bar{\beta}=p\beta\cap q\beta'$$

Since $\bar{\delta}(w)=\cup\{\delta(x_1\cdots x_n)\times\delta'(y_1\cdots y_n)\ ;\ w=x_1y_1\cdots x_ny_n\}$, it follows from induction on the length of w and the computation rule 2.7 that $\bar{\tau}\bar{\delta}(w)\bar{\beta}=\cup\{\tau\delta(x_1\cdots x_n)\beta\cap\tau'\delta'(y_1\cdots y_n)\beta'\ ;\ w=x_1y_1\cdots x_ny_n\}$. Therefore using 2.4(a), (b) we can conclude

$$T(M\sqcup M')=\{w\in\Sigma^{*}\ ;\ w=x_1y_1\cdots x_ny_n, x_1\cdots x_n\in T(M), y_1\cdots y_n\in T(M')\}$$

$$=Shuffle[T(M), T(M')]. \qquad □$$

4. Simple Recursive Program Medels

In this section we first consider a fixed point semantics on simple recursive programs by using relational calculus and describe a well-known relational characterization of while-do sentences, which leads an interesting problem on a characterization of iterative relations. The problem is not trivial so that we give a counterexample in additive relations. Finally we will prove the characterization of iterative relations in Heyting valued relations. We assume that the readers are familiar with the notion of Heyting valued stes (Cf. Goldblatt [5]).

Now consider a simple recursive program

while p do α

where a test $p:D\to D$ is a (guard) relation with $p\subseteq id_{D}$ and a statement $\alpha:D\to D$ is a relation.

Obviously a semantics of the above program is represented by an iterative relation $(\cup\{(p\alpha)^n\,;\,n\geq 0\})q$, where $q=id_D-p$. For a relation $\alpha:D\to D$ its iterative relation $\alpha^*:D\to D$ is defined by $\alpha^*=\cup\{\alpha^n\,;\,n\geq 0\}$, where $\alpha^0=id_D$, and $\alpha^{n+1}=\alpha^n\alpha$. Then the following theorem is a relational characterization of the semantics of **while** p **do** α:

THEOREM 4.1. *Let* α, p, $q:D\to D$ *be relations such that* $p\subseteq id_D$, $q\subseteq id_D$ *and* $p\cap q=\theta$. *Then the following statements are mutually equivalent*:

(a) $\gamma=(p\alpha)^*q$;

(b) $\gamma=p\alpha\gamma\cup q$, *and*
    $\delta=p\alpha\delta\cup q$ *implies* $\gamma\subseteq\delta$ *for any* $\delta:D\to D$;

(c) $\gamma=p\alpha\gamma\cup q$, $\gamma\subseteq\gamma q$, *and*
    $u p\alpha\subseteq p\alpha u$ *implies* $u\gamma\subseteq\gamma u$ *for any* $u:D\to D$ *with* $u\subseteq id_D$.   □

The condition (b) asserts that γ is the least fixed point of a relational equation $\delta=p\alpha\delta\cup q$. It is trivial that the conditions (a) and (b) are equivalent and (b) implies (c). The proof that (c) implies (a) follows from the next lemma:

LEMMA 4.2. *Let* α, $\beta:D\to D$ *be relations. Then* $u\alpha\subseteq\alpha u$ *implies* $u\beta\subseteq\beta u$ *for all relations* $u:D\to D$ *with* $u\subseteq id_D$ *if and only if* $\beta\subseteq\alpha^*$.   □

By the way the last lemma does not hold in the case of additive relations. Let A be an abelian group and $\alpha=id_A=\{(a,a)\,;\,a\in A\}$, $\beta=\{(a,0)\,;\,a\in A\}$ subgroups of $A\times A$. That is, α, $\beta:A\to A$ are additive relations. Then $\rho\beta\subseteq\beta\rho$ for any $\rho\subseteq id_A$ but $\beta\not\subseteq\alpha^*$ $(=id_A)$.

Finally we show that the lemma is also valid in the case of Heyting valued relations. Let Ω be a complete Heyting algebra (Cf. Goldblatt [5]).

An *Ω-valued relation* $\alpha:A\to B$ (Ω-relation for short) of an Ω-valued set (Ω-set for short) A to another Ω-set B is a mapping $\alpha:A\times B\to\Omega$ satisfying the following conditions:

(Ω-rel. 1) $[a\approx a']_A\sqcap[a\alpha b]\sqcap[b\approx b']_B\subseteq[a'\alpha b']$ for $a,a'\in A$ and $b,b'\in B$,

(Ω-rel. 2) $[a\alpha b]\subseteq[Ea]_A\sqcap[Eb]_B$ for $a\in A$ and $b\in B$,

where $[a\alpha b](\in\Omega)$ denotes the value of α at $(a,b)\in A\times B$.

Now we define the ordering $\sqsubseteq$, the composition, the involution $\sharp$, the greatest lower bounds $\sqcap$ and the least upper bounds $\sqcup$ of Ω-relations. Let $\alpha,\alpha':A\to B$ and $\beta:B\to C$ be Ω-relations. Then

$$\alpha\sqsubseteq\alpha' \text{ if } [a\alpha b]\subseteq[a\alpha'b] \quad \text{for } (a,b)\in A\times B,$$

$$[a\alpha\beta c] = \sqcup\{[a\alpha b] \sqcap [b\beta c]\; ;\; b \in B\} \qquad \text{for}\ \ (a,\,c) \in A \times C,$$

and

$$[b\alpha^{\#}a] = [a\alpha b] \qquad \text{for}\ \ (b,\,a) \in B \times A.$$

Note that the identity Ω-relation id_A on an Ω-set A is given by $[a\,id_A a'] = [a \approx a']_A$. Next for a family $\{\alpha_k : A \to B\; ;\; k \in K\}$ of Ω-relations, its least upper bound and greatest lower bound are defined as follows:

$$[a(\sqcup\{\alpha_k\; ;\; k \in K\})b] = \sqcup\{[a\alpha_k b]\; ;\; k \in K\}),$$

$$[a(\sqcap\{\alpha_k\; ;\; k \in K\})b] = \sqcap\{[a\alpha_k b]\; ;\; k \in K\}).$$

Similarly to the case of relations between ordinary sets, α^* is defined by $\alpha^* = \sqcup\{\alpha^n\; ;\; n \geq 0\} : A \to A$ for an Ω-relation $\alpha : A \to A$. By routine works we can also show that the laws 2.1, 2.2 and 2.5 are valid for Ω-relations. The following lemma is useful for the proof of the main theorem 4.4:

LEMMA 4.3.   *Let* $\alpha,\ \rho : A \to A$ *be* Ω-*relations with* $\rho \sqsubseteq id_A$. *Then*

(a) $[a\rho b] = [a \approx b]_A \sqcap [a\rho a]$ *for* $(a,\,b) \in A \times A$,

(b) $[a\rho\alpha b] = [a\rho a] \sqcap [a\alpha b]$ *for* $(a,\,b) \in A \times A$,

(c) $[a\alpha\rho b] = [a\alpha b] \sqcap [b\rho b]$ *for* $(a,\,b) \in A \times A$. $\sqsupset$

THEOREM 4.4.   *Let* $\alpha,\ \beta : A \to A$ *be* Ω-*relations on an* Ω-*set* A. *Then* $u\alpha \sqsubseteq \alpha u$ *implies* $u\beta \sqsubseteq \beta u$ *for all* Ω-*relations* $u : A \to A$ *with* $u \sqsubseteq id_D$ *if and only if* $\beta \sqsubseteq \alpha^*$.

Proof.   For each $x \in A$ define an Ω-relation $\rho_x : A \to A$ by

$$[a\rho_x b] = [a \approx b]_A \sqcap (\sqcap\{[a\alpha^* t] \Rightarrow [x\alpha^* t]\; ;\; t \in A\})$$

for $(a,\,b) \in A \times A$. (Where $p \Rightarrow q$ is a pseudo-complement of p relative to q for $p,\,q \in \Omega$.) Then it is trivial that ρ_x is really an Ω-relation on A and $\rho_x \sqsubseteq id_A$. Now we have

and

$$[a\rho_x a] \sqcap [a\alpha b] \sqsubseteq [Eb]_A = [b \approx b]_A$$

$$[a\rho_x a] \sqcap [a\alpha b] \sqsubseteq [b\alpha^* t] \Rightarrow [x\alpha^* t] \qquad \text{for all}\ \ t \in A$$

since

$$[a\rho_x a] \sqcap [a\alpha b] \sqcap [b\alpha^* t]$$

$$\sqsubseteq [a\rho_x a] \sqcap [a\alpha^* t] \qquad\qquad (\text{by}\ \alpha\alpha^* \sqsubseteq \alpha^*)$$

$$\sqsubseteq ([a\alpha^* t] \Rightarrow [x\alpha^* t]) \sqcap [a\alpha^* t] \quad (\text{by the defintion of}\ \rho_x)$$

$$\sqsubseteq [x\alpha^* t].$$

Hence we obtain

$$[a\rho_x a] \sqcap [a\alpha b] \sqsubseteq [b \approx b]_A \sqcap (\sqcap\{[b\alpha^* t] \Rightarrow [x\alpha^* t]\; ;\; t \in A\})$$

$$= [b\rho_x b],$$

which proves $\rho_x \alpha \sqsubseteq \alpha\rho_x$ because $[a\rho_x\alpha b] = [a\rho_x a] \sqcap [a\alpha b]$ and $[a\alpha\rho_x b] = [a\alpha b] \sqcap [b\rho_x b]$ by Lemma 4.3. Therefore $\rho_x\beta \sqsubseteq \beta\rho_x$ holds from the hypothesis of the theorem.   On the other hand we have $[a\rho_x a] \sqsubseteq [x\alpha^* a]$ since

$$[a\rho_x a] = [a \approx a]_A \sqcap [a\rho_x a]$$

$$\sqsubseteq [a\alpha^* a] \sqcap ([a\alpha^* a] \Rightarrow [x\alpha^* a])$$

$$\sqsubseteq [x\alpha^* a]$$

and $[\![x\rho_x x]\!]=[\![x\approx x]\!]_A$ from the defintion of ρ_x. At last we can conclude $\beta\sqsubseteq\alpha^*$ because for $(x,\ y)\in A\times A$

$$[\![x\beta y]\!]=[\![Ex]\!]_A\sqcap[\![x\beta y]\!]=[\![x\approx x]\!]_A\sqcap[\![x\beta y]\!]=[\![x\rho_x x]\!]\sqcap[\![x\beta y]\!]$$

$$=[\![x\rho_x\beta y]\!]\sqsubseteq[\![x\beta\rho_x y]\!]=[\![x\beta y]\!]\sqcap[\![y\rho_x y]\!]$$

$$\sqsubseteq[\![x\alpha^* y]\!].$$

This completes the proof of the theorem.   □

References

[1] M.S. CALENKO: *Classification of correspondence categories and types of regularity for cateogries*, Trans. Mascow Math. Soc., 1982, 239–282.

[2] H. EHRIG and H.-J. KREOWSKI: *Pushout-properties, An analysis of gluing constructions for graphs*, Math. Nachr. **91** (1979), 135–149.

[3] H. EHRIG, H.-J. KREOWSKI, A. MAGGIOLO-SCHETTINI, B.K. ROSEN and J. WINKOWSKI: *Transformations of structures, An algebraic approach*, Math. Systems Theory, **14** (1981), 305–334.

[4] S. EILENBERG and J.B. WRIGHT: *Automata in general algebras*, Inform. Control 11 (1967), 442–470.

[5] R. GOLDBLATT: *Topoi, The categorial analysis of logic*, North-Hollang, Amsterdam, 1979.

[6] S. HUZINO: *Computer Mathematics*, Maki-shoten, Tokyo, 1975. (in Japanese)

[7] Y. KAWAHARA: *Relational tree automata and context-free sets*, Bull. Kyushu Institute of Technology (M. & N.S.) No. **27** (1980), 17–25.

[8] Y. MIZOGUCHI: *Powerset monad, filter monad and primefilter monad in the category of sets with monoid actions*, Bull. Informatics and Cybernetics, Vol. 22 (1985), 83–95.

[9] D. PUPPE: *Korrespodenzen in Abelschen Kategorien*, Math. Ann. **148** (1962), 1–30.

[10] V. TRNKOVA: *Machines and their beoaviour in a category*, Fundamentals of Computation Theory (FCT'79) Ed. by L. Budach, Akademie-Verlag, Berlin, 1979.