Relational Arithmetic

Eric C. R. Hehner, University of Toronto
Theodore S. Norvell, Oxford University

Warning: this paper is unfinished, and in an unsatisfactory state.

Introduction

Relations were introduced by [Tarski] and have become popular recently as a semantics of programs [Backhouse, R.M.Dijkstra, Hehner, Hesselink, Hoare, Hoogerwoord, Mili, deMoor & Bird, Norvell, Z]. A relation is sometimes defined as a set of ordered pairs. More conveniently it can be defined as a function of two variables with a binary (boolean) result. Using programmers' terms, we call the first variable the input, and the second variable the output. A relation serves as a specification by saying which input-output pairs are acceptable. Relations have the advantage over functions from input to output in their ability to express nondeterminism. And refinement of specifications is just implication (or inclusion, in set terminology).

In stepwise refinement, a specification is refined (decomposed, implemented) by finding two or more new specifications whose composition implies the original specification. The new specifications are refined the same way, until we arrive at programs (specifications that are implemented). For example, a refinement step may take us from a specification R to specifications P and Q such that R is refined by the sequential composition of P and Q.

Programming does not always proceed "top-down" by stepwise refinement; sometimes it goes the other way. In a spirit of economy or ecology, we are encouraged to reuse software. For example, we may have a specification R to be implemented, and also a program Q to be used in the implementation of R; we want to find a specification (or program) P such that R is the sequential composition of P and Q. We need to "divide" R by Q to get P. This paper develops the necessary "arithmetic" of relations.

A kind of approximate division called "the weakest prespecification" was introduced by [Hoare & He]. In this paper we introduce exact division of relations, and roots of relations. Our goal is to provide the same sort of means for calculation of specifications (and programs) as we have for calculation of numbers.

Notation

Here are the notations we shall use in this paper, listed according to precedence.

0. $\top\ \bot\ \Top\ \Bot\ \Bit\ \#\ 0\ 1\ (\ )\ \langle\cdot\rangle$
1. adjacency
2. $\sim\ \surd$
3. $+\ -\ \oplus\ \times\ /\ \backslash$ **wpre ck**
4. $=\ \neq\ \leq$
5. $\wedge$
6. $\vee$
7. $\Rightarrow\ \Leftarrow$
8. $\forall\ \exists$
9. $\Equiv\ \Imp\ \Exp$

The large operators $\Equiv\ \Imp\ \Exp$ are identical to $=\ \Rightarrow\ \Leftarrow$ except for precedence.

Ordinary Relations

In our relations, the input and output variables will have the same type. Functions $\langle\cdot\rangle$ and quantifiers $\exists$ $\forall$ will implicitly range over that type. Here are four relations.

$\top\!\!\top$	$= \langle x, y\cdot \top\rangle$	top, true (where $\top$ is the true binary)
$\bot\!\!\bot$	$= \langle x, y\cdot \bot\rangle$	bottom, false (where $\bot$ is the false binary)
$\mathbb{I}$	$= \langle x, y\cdot x{=}y\rangle$	identity
$\neq$	$= \langle x, y\cdot x{\neq}y\rangle$	diversity

Relations can be transposed, composed, and compared as follows.

$\sim P = \langle x, y\cdot P\ y\ x\rangle$	transposition
$P{\times}Q = \langle x, z\cdot \exists y\cdot P\ x\ y \wedge Q\ y\ z\rangle$	composition
$P{=}Q = \forall x, y\cdot P\ x\ y = Q\ x\ y$	
$P{\neq}Q = \exists x, y\cdot P\ x\ y \neq Q\ x\ y$	
$P{\leq}Q = \forall x, y\cdot P\ x\ y \Rightarrow Q\ x\ y$	

The composition is sometimes called "sequential composition", and the symbol is sometimes a semi-colon; we are calling it a "product" (as did Tarski) and using the symbol $\times$ in this paper in order to draw an analogy with number multiplication. The comparison operators are obtained by "lifting" operators from binary up to relations; they suggest we might lift operators systematically, without introducing new symbols, but we do not pursue that suggestion here. The symbol $\leq$ is used to ease the analogy with number comparison; it formalizes refinement of specifications.

Here are some laws of relations. Let P, Q, and R be arbitrary relations. Then

0	$\sim\top\!\!\top = \top\!\!\top$	6	$\top\!\!\top{\times}\top\!\!\top = \top\!\!\top$
1	$\sim\bot\!\!\bot = \bot\!\!\bot$	7	$\mathbb{I}{\times}P = P = P{\times}\mathbb{I}$
2	$\sim\mathbb{I} = \mathbb{I}$	8	$P{\times}\bot\!\!\bot = \bot\!\!\bot = \bot\!\!\bot{\times}P$
3	$\sim\neq = \neq$	9	$P{\times}(Q{\times}R) = (P{\times}Q){\times}R$
4	$\sim\sim P = P$	10	$\bot\!\!\bot \leq P \leq P \leq \top\!\!\top$
5	$\sim(P \times Q) = \sim Q \times \sim P$	11	$P \leq Q \leq R \Rightarrow P \leq R$
		12	$P \leq Q \leq P = P = Q$
		13	$P \leq Q \Rightarrow P{\times}R \leq Q{\times}R \wedge R{\times}P \leq R{\times}Q$

The relations we have just described are, in some sense, like integers: we can always multiply them, but not always divide them. We can define an approximate division operator **wpre**, like the **div** operator in integer arithmetic, as follows:

$$P \textbf{ wpre } Q = \langle x, y\cdot \forall z\cdot P\ x\ z \Leftarrow Q\ y\ z\rangle$$

From this definition we can prove

$$P \leq Q \textbf{ wpre } R = P{\times}R \leq Q$$

This was called "the weakest prespecification" by [Hoare & He]. There are three other approximate division operators: multiplication is not symmetric so we can take it either way round, and we can round up or down. All four of them are awkward to use in calculations, just as integer division is.

A similar approximate division **ck**, called "the conjugate kernel", was defined by [Desharnais et al.] as

$$P \textbf{ ck } Q = \langle x, y\cdot \forall z\cdot P\ x\ z \Leftarrow Q\ y\ z\rangle \wedge (\exists z\cdot Q\ y\ z)$$

from which we can prove

$$P \textbf{ ck } Q \leq P \textbf{ wpre } Q$$

Extraordinary Relations

Our intention in this paper is to define exact division. Just as exact division of integers takes us outside the integers, so exact division of relations will take us outside the relations just described. To distinguish the relations that are analogous to integers from those that are not, we call the former "ordinary" relations, and the latter "extraordinary". Ordinary relations and their transpositions, compositions, and comparisons can be written using $\langle\cdot\rangle$ (more commonly known as λ-expressions); extraordinary relations and their transpositions, compositions, and comparisons cannot.

We previously listed some laws (0 through 13) about ordinary relations; we now take them to be axioms about all relations. We introduce two division operators, / (pronounced "over") and \ (pronounced "under") with the following axioms.

14	$P/\mathbb{I} = P$	20	$\mathbb{I}\backslash P = P$
15	$(P/P)\times P = P$	21	$P\times(P\backslash P) = P$
16	$P\times(Q/R) = (P\times Q)/R$	22	$(Q\backslash R)\times P = Q\backslash(R\times P)$
17	$P/(Q/R) = (P/Q)\times R$	23	$(Q\backslash R)\backslash P = Q\times(R\backslash P)$
18	$(P/Q)/R = P/(R\times Q)$	24	$Q\backslash(R\backslash P) = (R\times Q)\backslash P$
19	$\sim(P/Q) = \sim Q \backslash \sim P$	25	$\sim(Q\backslash P) = \sim P / \sim Q$

A different choice of axioms can define the same theory. The order of presentation can be different. This paper is not intended to make any point about the "right" choice of axioms, or order of presentation.

From these axioms, many laws can be proven. Here are some concerning / .

26	$\mathbb{I}/\mathbb{I} = \mathbb{I}$	31	$Q/Q = \mathbb{I} \Leftarrow \forall R\cdot R\times Q = Q \Rightarrow R=\mathbb{I}$
27	$\mathbb{1}/\mathbb{1} = \mathbb{1}$	32	$Q\times(\mathbb{I}/Q) = \mathbb{I} \Leftarrow \forall R\cdot R\times Q = Q \Rightarrow R=\mathbb{I}$
28	$P/Q = P\times(\mathbb{I}/Q)$	33	$(P\times Q)/Q = P \Leftarrow \forall R\cdot R\times Q = Q \Rightarrow R=\mathbb{I}$
29	$P/(P/P) = P$	34	$(P/Q)\times Q = P \Leftarrow \forall R\cdot R\times Q = Q \Rightarrow R=\mathbb{I}$
30	$(P/P)\times(P/Q) = P/Q$	35	$(P/Q)\times(Q/S) = P/S \Leftarrow \forall R\cdot R\times Q = Q \Rightarrow R=\mathbb{I}$

The laws say that this is exact, not approximate, division. There are similar laws concerning \ , which we do not bother to list here. For the remainder of the paper, we shall ignore ~ and \ .

Laws about number division have an antecedent saying that the divisor is unequal to 0 . To say that a number is unequal to 0 is equivalent to saying that 1 is its only identity. Analogously, Laws 31 through 35 have an antecedent saying that Q 's only left identity is $\mathbb{I}$.

Our algebra so far is "pointless", meaning that the axioms and laws do not refer to the points in an underlying space over which the relations are defined. The advantage is elegance; proofs of some theorems are shorter when we do not descend to the underlying space. One would be justified in wondering why we claim to be talking about relations at all. Indeed, so far the axioms do not often permit us to answer questions about the inverses of ordinary relations defined over a space. Laws 31 through 35 are almost impossible to use because the antecedent contains a universal quantification over all relations. We need an axiom relating division with a space. When restricted to ordinary relations, the antecedent in Laws 31 through 35 is equivalent to saying that Q relates distinct inputs to incomparable sets of outputs (incomparable means that neither is a subset of the other). So we propose the axiom

36 $Q/Q = \mathbb{I} \Leftarrow \forall x, y\cdot (\forall z\cdot Q\ x\ z \Rightarrow Q\ y\ z) \Rightarrow x=y$

(We could have used one of the other consequents. There is also a similar axiom for \ .) This axiom may be inelegant, but it does the job: it is weak enough to be consistent with the other axioms, and strong enough to prove a lot of results.

Domains

In some domains we obtain extraordinary relations by division, and in some we do not. In this section, when we say " Q has an inverse" we mean that we can prove $Q\times(\mathbb{I}/Q) = \mathbb{I}$, or equivalently that $Q/Q = \mathbb{I}$.

There are 2^{n^2} ordinary relations on an n-point domain.

The zero-point domain has one ordinary relation, which we have given the four names $\mathbb{T}$, $\mathbb{L}$, $\mathbb{I}$, and $\mathbb{H}$.

The one-point domain has two ordinary relations, one of which has been given the two names $\mathbb{I}$ and $\mathbb{T}$, and the other of which has been given the names $\mathbb{H}$ and $\mathbb{L}$. Only $\mathbb{I}$ has an inverse, and it is its own inverse, so we do not obtain any extraordinary relations by division in this domain. But we do obtain a model with more than one element, and so prove the consistency of the axioms: if $\mathbb{I}$ and $\mathbb{T}$ are both $\top$ (the true binary), $\mathbb{H}$ and $\mathbb{L}$ are both $\bot$ (the false binary), $\sim$ is identity, and $\times$, $/$, and $\backslash$ are all $\wedge$ (conjunction), then all the axioms are satisfied.

The two-point (binary) domain has 16 ordinary relations. Only $\mathbb{I}$ and $\mathbb{H}$ have inverses, and they are their own inverses, so again we do not obtain any extraordinary relations by division in this domain.

The smallest domain that is increased by division is the three-point domain. It has 512 ordinary relations, and infinitely many relations by division. Here are two ordinary relations that have extraordinary inverses.

$\langle x, y\cdot\ x=y \vee x=y\oplus 1\rangle$
$\langle x, y\cdot\ y=x \vee y=x\oplus 1\rangle$

where $\oplus$ is addition modulo 3.

It is amusing to note that

$\mathbb{H}\times\mathbb{H} = \mathbb{H}$	in the 0-point domain
$\mathbb{H}\times\mathbb{H} = \mathbb{L}$	in the 1-point domain
$\mathbb{H}\times\mathbb{H} = \mathbb{I}$	in the 2-point domain
$\mathbb{H}\times\mathbb{H} = \mathbb{T}$	in a domain of 3 or more points

Moving up to the infinite domain of integers, the relation

$\langle x, y\cdot\ x\le 0 \wedge y=x-1 \ \vee\ x\ge 0 \wedge y=x+1\rangle$

is a nice example of an ordinary relation that is neither injective (it is nondeterministic for input 0) nor surjective (it never allows output 0) and it has a right inverse; its right inverse is its transpose, which is the ordinary relation

$\langle x, y\cdot\ y\le 0 \wedge x=y-1 \ \vee\ y\ge 0 \wedge x=y+1\rangle$

Even more interesting are the ordinary relations

$\langle x, y\cdot\ x=y \vee x=y+n\rangle$

for each integer $n\neq 0$; their inverses are all extraordinary relations.

Rational Form

Rational numbers can be defined as those numbers generated by the integers together with division. Any rational number is the quotient of two integers. Analogously, we may consider those relations generated by the ordinary relations together with relational division. Not all such relations are the quotient of two ordinary relations, but they are all expressible as a continued quotient

$$P \,/\, (Q \,/\, (\ \ldots\ (R \,/\, S)\ \ldots\ ))$$

of ordinary relations. We have not yet developed a normal form or reduction algorithm.

Conclusions

We have just opened the subject of extraordinary relations, but not yet explored it very far. Just as for numbers, there is nothing to stop us from defining relations as the roots of equations, or more generally as the solutions of binary expressions. For example, square root relations are defined by

$$\sqrt{P} \times \sqrt{P} \;=\; P$$

Ordinary relations are useful as specifications of computation. The interpretation of extraordinary relations is not clear. Their value, their importance, will be demonstrated if they are useful in the calculation of programs.

Acknowledgments LATER

References

R.C.Backhouse, E.Voermans, J.C.S.P.van der Woude: "a Relational Theory of Datatypes", *EURICS Workshop on Calculational Theories of Program Structure* (editor Backhouse), 1991.

R.S.Bird, O.deMoor: "Solving Optimization problems with catamorphisms", Mathematics of Program Construction, Oxford, 1992, LNCS 669, Bird, Morgan, Woodcock editors, Springer-Verlag, 1993.

J.Desharnais, A.Jaoua, F.Mili, N.Boudriga, A.Mili: "a Relational Division Operator: the Conjugate Kernel", unknown journal and date

R.M.Dijkstra: *Relational Calculus and Relational Program Semantics*, master's thesis, University of Kiel, 1992.

E.C.R.Hehner: "Predicative Programming", *Communications ACM,* volume 27, number 2, 1984 February, pages 134-151

E.C.R.Hehner: *a Practical Theory of Programming*, Springer-Verlag, 1993.

C.A.R.Hoare: "Programs are Predicates", *Mathematical Logic and Programming Languages* (editors C.A.R.Hoare, J.C. Shepherdson), pages 141-154, Prentice-hall International, 1985.

C.A.R.Hoare, He J.: "the Weakest Prespecification", *Information Processing Letters*", volume 24, pages 127-132, 1987.

R.R.Hoogerwoord: *the Design of Functional Programs: a Calculational Approach*, doctoral thesis, University of Eindhoven, 1989.

M.B.Josephs: *an Introduction to the Theory of Specification and Refinement*, IBM Technical Report RC 12993, 1987.

A.J.Mili, J.Desharnais, F.Mili: "Relational Heuristics for the Design of Deterministic Programs", *Acta Informatica*, volume 24, pages 239-276, 1987.

Oege deMoor, Richard S. Bird: “Solving Optimization Problems with Catamorphisms”, *Proceedings of the Mathematics of Program Construction*, Oxford, 1992, LNCS 669 (editors Bird, Morgan, Woodcock), Springer-Verlag, 1993.

T.S.Norvell, E.C.R.Hehner: “Logical Specifications for Functional Programs”, *Proceedings of the Mathematics of Program Construction*, Oxford, 1992, LNCS 669 (editors Bird, Morgan, Woodcock), pages 269-290, Springer-Verlag, 1993.

A.Tarski: “on the Calculus of Relations”, *Journal of Symbolic Logic* volume 6, pages 73-89, 1941.

J.M.Spivey: *Introducing Z: A Specification Language and its Formal Semantics*, Cambridge University Press, 1988.

USEFUL-LOOKING THEOREM

$$P \times Q = \mathbb{I} \ \wedge \ Q \times P = P \ \Rightarrow \ P = \mathbb{I} = Q$$

PROOF

$$\begin{array}{ll} & P \times Q = \mathbb{I} \ \wedge \ Q \times P = P \\ = & P \times Q = \mathbb{I} \ \wedge \ P \times Q = \mathbb{I} \ \wedge \ Q \times P = P \\ \Longrightarrow & P \times Q = \mathbb{I} \ \wedge \ Q \times P \times Q = \mathbb{I} \\ \Longrightarrow & P \times Q = \mathbb{I} \ \wedge \ Q \times \mathbb{I} = \mathbb{I} \\ = & P \times Q = \mathbb{I} \ \wedge \ Q = \mathbb{I} \\ = & P = \mathbb{I} \wedge Q = \mathbb{I} \end{array}$$

Possible change of terminology, including in title:

> The name change you suggest is:
> ordinary relation --> relation
> relation --> specification

> Some people do not consider
> specifications to be (any generalization of) relations

Doesn't matter, standard relations are only one model of the standard axioms.
As long as these people are content that there are specifications
and that there are specification operators II, TT, FF, *, and ~
that obey the axioms that don't mention / or \, then most of the paper
is relevent to them. (Predicate transformers are a case in point.) In fact
this change should make the paper more interesting to this class
of people.

> And we lose the interest of people who are interested in relations
> because we are not changing or generalizing them.

I'm worried about losing their interest because the things we are calling
"relations" are not the relations they are interested in. We should
make it clear that the thingies we are talking about are relevant to
the relations they are interested in. And we should make the exact
relationship as clear as possible by using standard vocabulary.

What _I_ don't like about the word "specification" is that
what we are doing is relevant to lots of monoids that have nothing to
do with specifications or even computer science. I'd be happy
with another word. This is something new (as far as we know),
so perhaps a new word. "divoids" ?

Comments on draft of may 26 1994.

P 0 L5. "Relations were introduced by Tarski". Are you sure?
I would find "Relational calculus was introduced by Tarski" easier
to believe.

P0 L6. "Backhouse" --> "Backhouse & al."

P2. L2. Does this theorem remain true in the model of extraordinary relations
P2. L3. "There are three other approximate division operators." I'm not
so sure about this. wpre is presented by defining it first in the

model of ordinary relations and then by showing a theorem about it.
You can take the same tack for wpost, but for spre and spost, I
don't think you can define them in the standard model. What you can do is
introduce:

(P _< Q wpre R) = (P*Q _< Q)

...

(Q spost R _< P) = (Q _< R*P)

as four new axioms. However the last two axioms rule out (I believe)
the standard relations as a model.

P4 "Rational form". The big difference between extraordinary relations and
rationals is that rationals can be modeled (fully and faithfully) as
equivalence classes over pairs of integers. We have nothing similar.
Do we?

On 2006-3-22 Vadim Tropashko pointed out that in one way the ordinary relations are more like naturals than integers. $P \vee X = R$ (equation in unknown X) has a solution iff $P \Leftarrow R$. Why not give it a unique solution all the time, and thus invent negative relations (and full subtraction)? And similarly give $P \wedge X = R$ unique solutions.