Courtesy Apt, ETAC, Hoogerwoord, & Voermans

This note deals with a problem posed to me by Apt when I visited him in Amsterdam. I mentioned the problem at the next session of the ETAC, where we did not solve it; Hoogerwoord, however, had suggested a structure of the argument and had provided a major building block. The next day, Voermans provided the missing ingredient and completed the proof.

* * *

We consider an expression built according to the following syntax

<exp> ::= <atom> | ¬<exp>

| ∨<exp><exp> | ∧<exp><exp> .

We are given the following rewrite rules

(0) $\neg\neg x \;\rightarrow\; x$

(1) $\neg\vee x y \;\rightarrow\; \wedge\neg x \neg y$

(2) $\neg\wedge x y \;\rightarrow\; \vee\neg x \neg y$

(3) $\wedge x \vee y z \;\rightarrow\; \vee\wedge x y \wedge x z$

(4) $\wedge\vee x y z \;\rightarrow\; \vee\wedge x z \wedge y z$

in which x, y, z are subexpressions of type <exp> . Our algorithm consists of repeatedly replacing a subexpression matching the left-hand side of a rewrite rule by the corresponding right-hand side of that rule. The challenge is to prove that the algorithm terminates because, sooner or later, none of the rewrite rules is applicable any more.

* * *

We try to define a function f from expressions to natural numbers —or, if the need arises, to a more general well-founded domain— such that the function value decreases when a rewrite rule is applied to the argument. Our first decision —taken for the sake of simplicity— is to define f recursively over the syntax, i.e. besides defining

$$f.\langle atom\rangle \;=\; const$$

we define

$$\begin{aligned} f.(\neg x) &= neg.(f.x) \\ f.(\vee x y) &= dis.(f.x, f.y) \\ f.(\wedge x y) &= con.(f.x, f.y) \quad ; \end{aligned}$$

the challenge is now to define const, neg, dis, and con in such a way that application of rewrite rules leads to a decrease of f. Since rewrite rules can be applied to replace subexpressions, whereas the f-value of the whole expression has to decrease, the functions neg, dis, and con have to be strongly monotonic in all their arguments, i.e.

$$(5)\quad p>p' \wedge q>q' \Rightarrow \begin{array}{l} neg.p > neg.p' \;\wedge \\ dis.(p,q) > dis.(p',q) \;\wedge \\ dis.(p,q) > dis.(p,q') \;\wedge \\ con.(p,q) > con.(p',q) \;\wedge \\ con.(p,q) > con.(p,q') \quad . \end{array}$$

To begin with we focus our attention on (3) and (4), which describe how $\wedge$ distributes over $\vee$, viz. in the same way as $*$ (times) distributes over $+$ (plus). This last observation suggests to choose for dis and con something like $dis.(p,q) = p+q$ and $con.(p,q) = p*q$; I wrote "something like" because the above choice would leave the f-value under rewritings (3) and (4) unchanged. Let us investigate with

$$dis.(p,q) = p+q \quad , \quad con.(p,q) = p*q - c$$

the requirement that (3) leads to a decrease of f:

$$\begin{array}{ll} & f.(\wedge x \vee y z) > f.(\vee \wedge x y \wedge x z) \\ = & \quad \{\text{def. of con ; def. of dis}\} \\ & f.x * f.(\vee y z) - c > f.(\wedge x y) + f.(\wedge x z) \\ = & \quad \{\text{def. of dis ; def. of con}\} \\ & f.x * (f.y + f.z) - c > f.x * f.y - c + f.x * f.z - c \\ = & \quad \{\text{algebra}\} \\ & c > 0 \quad . \end{array}$$

The requirement that application of (4) leads to an f-decrease is equivalent to the same $c>0$. In order to ensure that f-values are natural we choose a natural const satisfying

$$const^2 - c \geq const \;.$$

These conditions can be satisfied, e.g by $c=1$ and $const = 2$. We are left with the obligation of constructing a neg, such that rewritings (0), (1) and (2) decrease f.

From (0) we conclude the requirement

(6) $\quad neg.(neg.p) > p$.

From (1) we conclude the requirement

(7) $\quad neg.(p+q) > neg.p * neg.q - c$.

From (2) we conclude the requirement

(8) $\quad neg.(p*q-c) > neg.p + neg.q$.

These three requirements should be satisfied for $p \geq const$ and $q \geq const$, const being the minimum f-value.

(6) is satisfied if $neg.p > p$; for monotonic neg satisfying $neg.p > p$, (8) is unlikely to present problems for larger arguments; (7) imposes a clear constraint, but since $c > 0$,

$$neg.p = d^p$$

satisfies (7). For $c=1$ and $const = 2$, $d=2$ is too small — since $2^{2\times 2-1} = 2^2 + 2^2$, (8) can be violated — ; $d=3$, however, does the job. In short

$$\begin{aligned}
f.\langle atom\rangle &= 2\\
f.(\neg x) &= 3^{f.x}\\
f.(\vee x y) &= f.x + f.y\\
f.(\wedge x y) &= f.x * f.y - 1
\end{aligned}$$

is a witness demonstrating the existence of a variant function. Another witness is given by
$c, const, d = 1, 3, 2$.

* * *

At the ETAC, I got stuck. I started with (0), (1) & (2), mapping the latter two on each other by ignoring the difference between $\wedge$ and $\vee$; subsequently introducing the ignored difference by taking (3) & (4) into account gave serious problems. The advantage of starting with (3) & (4) is that then $\neg$ is ignored automatically.

The reader is asked to realize how much the derivation has been eased by the introduction of the named functions const, neg, dis, & con .

Nuenen, 29 August 1991

prof. dr. Edsger W. Dijkstra
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712-1188
USA