Sums and differentials

SUMS and DIFFERENTIALS

Roland Backhouse

October 8, 1991

Abstract

An elementary example is given of how the identification of a Galois connection can assist in algorithm derivation.

In the coming months a major activity of the Eindhoven Relational Type Theory Group will be to explore the theory and applications of Galois connections and adjunctions. In order to explain more clearly the sort of objectives I had in mind for such an exploration I sought an appealing example that could be presented within the space of twenty minutes. By good fortune, in connection with a quite different matter, my attention was drawn to the section on "finite calculus" in the book "Constructive Mathematics" by Graham, Knuth and Patashnik. Further reading led me to the construction of the following example which seems to fit my two criteria.

Let f and g denote functions from numbers to reals. Assume that $f.0 = 0$. Define the operators Δ and Σ by

$$\begin{array}{lcl}(\Delta f).x & = & f.(x+1) - f.x \\ (\Sigma g).x & = & \Sigma(y:\ 0 \leq y < x : g.y)\end{array}$$

for all numbers x. Then we have the Galois connection:

(1) $\quad f = \Sigma g \ \equiv\ \Delta f = g$

The proof of this identity involves very elementary quantifier calculus and is therefore omitted.

Applying the extensionality axiom (two functions with the same domain are equal if and only if they are equal at all elements of their common domain) we obtain the equivalent but lengthier:

(2) $\quad \forall(x :: f.x = (\Sigma g).x) \ \equiv\ \forall(y :: (\Delta f).y = g.y)$

$f.x$	$(\Delta f).x$
0	0
cx	c
$f.x + g.x$	$(\Delta f).x + (\Delta g).x$
$f.x \times g.x$	$f.x \times (\Delta g).x + g.(x+1) \times (\Delta f).x$

Table 1: Table of Differentials

Let us suppose our goal is to develop a body of rules that enable one to find efficient ways of evaluating finite sums Σg for given function g. This goal may be approached by tackling the easier problem of developing a body of rules to compute differentials Δf and then using the Galois connection (1) to convert the rules to rules about Σ.

To illustrate this idea let us restrict g to the class of polynomial functions. Our goal is thus to develop a little theory that will enable us to compute finite sums of polynomials such as $\Sigma(y: \ 0 \leq y < x: \ y^2 + 3y + 1)$.

We begin our theory development by exploring the *differentials* of polynomials. Since a polynomial function of x is either a constant function, the identity function, the sum of two polynomial functions or the product of two polynomial functions, table 1 suffices to rewrite $(\Delta f).x$ as a polynomial in x for any given polynomial $f.x$ satisfying the assumption $f.0 = 0$. (In the table c denotes an arbitrary constant. Verification of all four statements is straightforward.) We observe that a table of differentials in the finite calculus looks like a table of differentials in the infinite calculus but for the unfortunate form of the product rule. In particular taking derivatives reduces the degree of a polynomial by exactly one.

Ideally we would now like to construct a similar table for Σ. Four entries would be required, one for constants, one for the identity function, one for a sum and one for a product of two polynomials. The Δ entry for products frustrates this particular goal but nevertheless an algorithm for expressing the sum of a polynomial function as a polynomial function can be derived that exploits the above table of differentials. I shall illustrate the algorithm by considering the Σ entry for the identity function.

Since taking derivatives reduces the degree of a polynomial by one we conjecture that the sum of the identity function is a quadratic polynomial. The coefficients of that poynomial are calculated as follows:

By construction of a and b:

$$
\begin{array}{ll}
 & \forall(x :: \ ax + bx^2 \ = \ \Sigma(y : \ 0 \le y < x : \ y)) \\
\equiv & \qquad \{ \text{ Galois connection: (2) } \} \\
 & \forall(y :: \ \Delta(x \mapsto ax + bx^2).y \ = \ y) \\
\equiv & \qquad \{ \text{ differential calculus: table 1 } \} \\
 & \forall(y :: \ a + by + b(y+1) \ = \ y) \\
\equiv & \qquad \{ \text{ arithmetic } \} \\
 & a + b = 0 \ \wedge \ 2b = 1 \\
\equiv & \qquad \{ \text{ arithmetic } \} \\
 & a = -\frac{1}{2} \ \wedge \ b = \frac{1}{2}
\end{array}
$$

We have thus established the identity

$$\Sigma(y : \ 0 \le y < x : \ y) \ = \ -\tfrac{1}{2}x + \tfrac{1}{2}x^2$$

Extrapolating from this four step calculation one can easily see that it embodies an algorithm to express Σg as a polynomial function for any given polynomial function g. The steps in the algorithm are: postulate that Σg is a polynomial function f with degree one higher than g. Compute (symbolically) the coefficients of Δf using the table of differentials. Equate the expressions obtained for the coefficients of f to the corresponding given coefficients of g. In this way one obtains a system of simultaneous equations which is then solved to obtain the coefficients of f.

The point of this little example is to show how one can predict the behaviour of a relatively complicated operator — in this case Σ — by studying the behaviour of its adjoint — in this case Δ.