MPL's Derivative Operator

Let u be an algebraic expression and let x be a symbol. MPL's operator $\mathit{Derivative}(u, x)$, which evaluates the derivative of u with respect to x, is defined by the following transformation rules:

DERIV-1. If $u = x$, then $\mathit{Derivative}(u, x) \to 1$.

DERIV-2. If $u = v^w$, then

$$\mathit{Derivative}(u, x) \to \\ w * v^{w-1} * \mathit{Derivative}(v, x) + \mathit{Derivative}(w, x) * v^w * \ln(v).$$

This rule applies to expressions that are powers and accounts for expressions where either v or w may depend on x. (The rule is derived using logarithmic differentiation.) Since the *Derivative* operator appears on the right side of the rule, **DERIV-2** is recursive. When w is free of x, the rule reduces (with automatic simplification) to the familiar power rule

$$\frac{d(v^w)}{d\,x} = w \cdot v^{w-1} \frac{d(v)}{d\,x}.$$

DERIV-3. Suppose u is a sum and let $v = \mathit{Operand}(u, 1)$ and $w = u - v$. Then

$$\mathit{Derivative}(u, x) \to \mathit{Derivative}(v, x) + \mathit{Derivative}(w, x).$$

DERIV-4. Suppose u is a product and let $v = \mathit{Operand}(u, 1)$ and $w = u/v$. Then

$$\mathit{Derivative}(u, x) \to \mathit{Derivative}(v, x) * w + v * \mathit{Derivative}(w, x).$$

Rules **DERIV-3** and **DERIV-4** are the sum and product differentiation rules. Again, the rules are recursive because the right side of each rule refers to the *Derivative* operator. Notice that we obtain the derivative of a sum by differentiating both the first operand and the remaining part of the sum, which is obtained by subtracting the first operand from u with automatic simplification. A similar approach is used for a product.

A typical rule for a known function looks like the following:

DERIV-5. If $u = \sin(v)$, then $\mathit{Derivative}(u, x) \rightarrow \cos(v) * \mathit{Derivative}(v, x)$.

Again, the chain rule implies the rule is recursive.

DERIV-6. If *Free_of*$(u, x) =$ **true**, then $\mathit{Derivative}(u, x) \rightarrow 0$.

This rule applies to integers, fractions, symbols, and compound expressions (such as $f(a)$ or $n!$) that are free of the differentiation variable x. Notice that powers, sums, and products are not checked by this rule because they are handled by one of the earlier rules **DERIV-2**, **DERIV-3**, or **DERIV-4**. For example, if b and e are symbols ($\neq x$), then

$$\mathit{Derivative}(b^e, x) \rightarrow 0$$

is obtained by first applying **DERIV-2**, which applies **DERIV-6** (recursively) to both b and e.

We have placed **DERIV-6** at this point in the rule sequence to avoid redundant calls on the *Free_of* operator. The reason for this has to do with the recursive nature of *Free_of*. If **DERIV-6** were at the beginning of the rule sequence, then to compute the derivative (with respect to x) of

$$u = (1 + a)^2 + x^2,$$

the algorithm would first check if u were free of x, which involves the comparison of each complete sub-expression of u to x until the symbol x is found. Since this step would return **false**, we would next apply the sum rule which obtains the derivative in terms of the derivatives of the two operands $(1 + a)^2$ and x^2. To find the derivative of $(1 + a)^2$, we would check (for the second time) if this expression were free of x. By placing the *Free_of* operation later in the rule sequence, we avoid this redundant calculation.

The final transformation rule applies to any expression that is not covered by the earlier rules:

DERIV-7. $\mathit{Derivative}(u, x) \rightarrow$ "$\mathit{Derivative}$"(u, x).

In other words, if none of the earlier rules apply to u, the expression is returned in the unevaluated form $\mathit{Derivative}(u, x)$. The *Derivative* operator on the right is quoted to prevent a recursive evaluation of the operator because, without the quotes, the transformation leads to an infinite sequence of recursions. By including this rule, we obtain a representation for the derivative of expressions that include undefined functions such as

$$\begin{aligned} \mathit{Derivative}(f(x) * g(x), x) \quad \rightarrow \quad & \mathit{Derivative}(f(x), x) * g(x) \\ & + f(x) * \mathit{Derivative}(g(x), x), \end{aligned}$$

where the derivatives of $f(x)$ and $g(x)$ remain in unevaluated form.

Notice that the differentiation quotient rule is not included in our rule sequence because we assume that automatic simplification transforms quotients to products or powers.

The **DERIV** rules are an example of a transformation rule sequence. When describing an algorithm in this way, we assume that a rule is checked only when all earlier rules do not apply. This approach simplifies the presentation because conditions that are handled by earlier rules need not be repeated (in a negative sense) in a later rule.

Most computer algebra systems have an operator that is similar to MPL's *Derivative* operator:

MPL	Maple	Mathematica	MuPAD
Derivative(u, x)	`diff(u, x)`	`D[u, x]`	`diff(u, x)`

(Implementation: Maple (mws), Mathematica (nb), MuPAD (mnb).)

Return to Chapter 1, page 14