ELSEVIER

Journal of Computational and Applied Mathematics 50 (1994) 119–131

JOURNAL OF
COMPUTATIONAL AND
APPLIED MATHEMATICS

The if-problem in automatic differentiation

Thomas Beck *, Herbert Fischer

Institut für Angewandte Mathematik und Statistik, Technische Universität München, Arcisstraße 21, D-80333 München, Germany

Received 18 August 1992

Abstract

We deal with a severe problem that arises in the automatic differentiation of functions. Many computer programs defining a function employ statements of the form *if* $B(x)$ *then* S1 *else* S2, where $B(x)$ is a Boolean expression and S1 and S2 denote subprograms. This often leads to a piecewise definition of the function under consideration. Automatic differentiation of the pieces may be hazardous, for instance in cases where the underlying function is differentiable but one or the other piece is not. In such cases available software often fails to produce correct results. To resolve this perplexity, we distinguish between a function and its representations. In particular, we introduce the notion *derivative-consistent*. Automatic differentiation applied to a derivative-consistent representation of a function yields correct results.

Key words: Automatic differentiation; Piecewise defined function

1. Introduction

We investigate a special problem arising in the automatic differentiation of programs which describe functions. Consider a differentiable real-valued function f of n variables

$$f\colon D \subseteq \mathbb{R}^n \to \mathbb{R}.$$

Assume that we have a program P with input x and output $f(x)$:

$$f(x) \longleftarrow \boxed{P} \longleftarrow x.$$

For $x \in D$ we ask for the function value $f(x)$ *and* the derivative value $f'(x)$. So, we would like to have a program P′ with input x and output $f(x)$, $f'(x)$:

$$f(x),\ f'(x) \longleftarrow \boxed{P'} \longleftarrow x.$$

* Corresponding author.

SSDI 0377-0427(93)E0156-G

Let us depict the transformation of P to P′ by a black box DIFF:

$$\mathrm{P}' \longleftarrow \boxed{\mathrm{DIFF}} \longleftarrow \mathrm{P}.$$

In case P consists of an explicit formula for $f(x)$ involving only rational operations and library functions such as sin, exp, ..., DIFF can be carried out by hand, or by symbolic manipulators, e.g., REDUCE, MACSYMA, MAPLE, or by employing automatic differentiation tools. In case P consists of a straightforward list of explicit formulas, automatic differentiation tools as described in [7,8] and others are a powerful means to perform DIFF. A taxonomy of automatic differentiation tools can be found in [5].

In case P is not straightforward, difficulties arise. Contemporary implementations of DIFF may fail to produce correct results in some instances where P contains statements of the form

$$\text{if } B(x) \text{ then S1 else S2,} \qquad (*)$$

where $B(x)$ is a Boolean expression and S1 and S2 are subprograms. There exist examples where $f(x)$ is defined in branch S2, but $f'(x)$ cannot be computed, not even defined, with the information of branch S2.

The problem of implementing DIFF properly in the presence of statements of the form $(*)$ is what we call *the if-problem*.

In the next section we provide some automatic differentiation details. In Section 3 we state the if-problem and present an illustrative example. Then, in Section 4, we delve into details and we investigate the behavior of piecewise defined functions. For this purpose we distinguish between a function f, a program P describing f, and a representation R of f. The representation R is independent of programming languages. Finally, in Section 5, we give a partial solution for the if-problem.

2. Automatic differentiation

Within the last decade it has become apparent that the derivative of an explicitly given function can be computed efficiently by automatic differentiation techniques. These techniques can be classified into two groups, the *bottom-up* methods and the *top-down* methods. The if-problem appears in both modes, so we choose the bottom-up mode (this is easier to line out). Here we only list some details necessary in the sequel.

Let us take up two familiar ways for building new functions from old ones, first the rational composition and second the library composition.

Consider two differentiable functions

$$u: D \subseteq \mathbb{R}^n \to \mathbb{R} \quad \text{and} \quad v: D \subseteq \mathbb{R}^n \to \mathbb{R}.$$

Let w be one of the functions $u+v$, $u-v$, $u \cdot v$, u/v with the proviso that $v(x) \neq 0$ for all $x \in D$ in the case $w = u/v$. Table 1 shows well-known formulas for the derivative of w.

From these formulas we conclude that the pair $w(x)$, $w'(x)$ can be computed from the pairs $u(x)$, $u'(x)$ and $v(x)$, $v'(x)$. Note that the pair $w(x)$, $w'(x)$ is not a pair of formulas, nor is it a pair of functions, instead it is an element of $\mathbb{R} \times \mathbb{R}^{1,n}$. The mechanism to compute the pair

Table 1
Derivative of rational composition

Type	Function	Derivative
$+$	$w(x)=u(x)+v(x)$	$w'(x)=u'(x)+v'(x)$
$-$	$w(x)=u(x)-v(x)$	$w'(x)=u'(x)-v'(x)$
$\cdot$	$w(x)=u(x)\cdot v(x)$	$w'(x)=v(x)\cdot u'(x)+u(x)\cdot v'(x)$
$/$	$w(x)=u(x)/v(x)$	$w'(x)=(u'(x)-w(x)\cdot v'(x))/v(x)$

$w(x)$, $w'(x)$ from the pairs $u(x)$, $u'(x)$ and $v(x)$, $v'(x)$ does not depend on x, it does not depend on the functions u and v, it merely depends on the type of w. This observation allows to define and implement a function RAT, which accepts the type of w and the pairs $u(x)$, $u'(x)$ and $v(x)$, $v'(x)$, and which produces the pair $w(x)$, $w'(x)$:

$$w(x),\, w'(x) \longleftarrow \boxed{\text{RAT}} \begin{array}{l} \longleftarrow\ * \text{ from } \{+,\ -,\ \cdot\ ,\ /\} \\ \longleftarrow\ u(x),\, u'(x) \\ \longleftarrow\ v(x),\, v'(x) \end{array}$$

Let Λ be a collection of differentiable real-valued functions of one real variable, such as sin, exp, sqrt and the like. For brevity these functions are called *library functions*. Consider some library function

$$\lambda\colon L \subseteq \mathbb{R} \to \mathbb{R}$$

and some differentiable function

$$u\colon D \subseteq \mathbb{R}^n \to \mathbb{R}.$$

Under the proviso that $u(D) \subseteq L$ we define the function

$$w\colon D \subseteq \mathbb{R}^n \to \mathbb{R} \quad \text{with } w(x) := \lambda(u(x)).$$

Then by the chain rule we have

$$w'\colon D \subseteq \mathbb{R}^n \to \mathbb{R}^{1,n} \quad \text{with } w'(x) = \lambda'(u(x)) \cdot u'(x).$$

Here we conclude: the pair $w(x)$, $w'(x)$ can be computed from the pair $u(x)$, $u'(x)$ using λ and λ'. And the mechanism to compute the pair $w(x)$, $w'(x)$ merely is a matter of the library function λ. We assume that we are able to evaluate λ and λ' at any suitable value of the argument. This is no problem as long as λ is one of the commonly used library functions sin, exp, sqrt and the like. Hence, we can define and implement a function LIB, which accepts the name of λ, and the pair $u(x)$, $u'(x)$, and which produces the pair $w(x)$, $w'(x)$:

$$w(x),\, w'(x) \longleftarrow \boxed{\text{LIB}} \begin{array}{l} \longleftarrow\ \lambda \text{ from } \Lambda \\ \longleftarrow\ u(x),\, u'(x) \end{array}$$

Let us illustrate the use of RAT and LIB by a simple example. Consider the function

$$f\colon D \subseteq \mathbb{R}^3 \to \mathbb{R}, \quad \text{with } f(x) := (x_1 - 7) \cdot \sin(x_1 + x_2)/x_3,$$

Table 2
Computation of $Y_9 = (f(x), f'(x))$

$y_1 = x_1$	$f_1(x) = x_1$	$Y_1 = (x_1, [1, 0, 0])$
$y_2 = x_2$	$f_2(x) = x_2$	$Y_2 = (x_2, [0, 1, 0])$
$y_3 = x_3$	$f_3(x) = x_3$	$Y_3 = (x_3, [0, 0, 1])$
$y_4 = 7$	$f_4(x) = 7$	$Y_4 = (7, [0, 0, 0])$
$y_5 = y_1 - y_4$	$f_5(x) = f_1(x) - f_4(x)$	$Y_5 = \text{RAT}(-, Y_1, Y_4)$
$y_6 = y_1 + y_2$	$f_6(x) = f_1(x) + f_2(x)$	$Y_6 = \text{RAT}(+, Y_1, Y_2)$
$y_7 = \sin(y_6)$	$f_7(x) = \sin(f_6(x))$	$Y_7 = \text{LIB}(\sin, Y_6)$
$y_8 = y_5 \cdot y_7$	$f_8(x) = f_5(x) \cdot f_7(x)$	$Y_8 = \text{RAT}(\cdot, Y_5, Y_7)$
$y_9 = y_8 / y_3$	$f_9(x) = f_8(x) / f_3(x)$	$Y_9 = \text{RAT}(/, Y_8, Y_3)$

where $D = \{x \mid x \in \mathbb{R}^3,\ x_3 \neq 0\}$. For given $x \in D$ the function value $f(x)$ can be computed step-by-step as shown in the first column of Table 2.

The setup for the intermediate values $y_1, \ldots, y_9$ gives rise to the definition of intermediate functions $f_1, \ldots, f_9$ as shown in the second column of Table 2. Of course, $f_9 = f$.

Let us introduce pairs $Y_k = (f_k(x), f_k'(x))$ for $k = 1, \ldots, 9$, consisting of a function value and the corresponding derivative value. For $k = 1, 2, 3, 4$ the pair Y_k is obvious. And for $k = 5, 6, 7, 8, 9$ in this order the pair Y_k can be computed from previous pairs using RAT and LIB as shown in the third column of Table 2. The final pair is

$$Y_9 = (f_9(x), f_9'(x)) = (f(x), f'(x)).$$

In Table 2 the first column is a program P for the function f, the second column defines a characterizing sequence $f_1, \ldots, f_9$ of functions with $f_9 = f$, and the third column is the transformed program P′ for computing the pair $f(x)$, $f'(x)$. Furthermore, the resemblance between P and P′ demonstrates that the transformation DIFF from P to P′ is a matter of simple replacements.

Now we turn to the general case. Assume that for a differentiable function $f\colon D \subseteq \mathbb{R}^n \to \mathbb{R}$ we have a straightforward program P as shown in Fig. 1.

The program P defines a *characterizing sequence* $f_1, \ldots, f_s$ of functions $D \to \mathbb{R}$ in an obvious way, and $f_s = f$. To the sequence $f_1, \ldots, f_s$ there corresponds the sequence $f_1', \ldots, f_s'$ of

Step 1: for $k = 1, \ldots, n$
  $y_k = x_k$ (kth component of x)
Step 2: for $k = n+1, \ldots, n+d$
  $y_k = c_k$ (some constant)
Step 3: for $k = n+d+1, \ldots, s$
  $y_k = y_i * y_j$ with $* \in \{+, -, \cdot, /\}$ and $i, j < k$
  or
  $y_k = \lambda(y_i)$ with $\lambda \in \Lambda$ and $i < k$
Step 4: $f(x) = y_s$

Fig. 1. Program P for evaluating f at x.

Step 1: for $k = 1, \ldots, n$
$Y_k = (x_k, [0\ldots 0\ 1\ 0\ldots 0])$ with 1 in position k
Step 2: for $k = n+1, \ldots, n+d$
$Y_k = (c_k, 0\ldots 0])$
Step 3: for $k = n+d+1, \ldots, s$
$Y_k = \mathrm{RAT}(*, Y_i, Y_j)$
or
$Y_k = \mathrm{LIB}(\lambda, Y_i)$
Step 4: $f(x)$ = first member of the pair Y_s
$f'(x)$ = second member of the pair Y_s

Fig. 2. Program P′ for evaluating f and f' at x.

derivatives. For given $x \in D$ the sequence $Y_1, \ldots, Y_s$ of pairs with $Y_k = (f_k(x), f_k'(x))$ for $k = 1, \ldots, s$ can be computed straightforwardly with the program P′ of Fig. 2. Again, the transformation DIFF from P to P′ is a matter of simple replacements.

So far, so good. A broad class of functions fits into the frame of the program P of Fig. 1, all functions which McCormick [6] calls *factorable*. But this is not all. Automatic differentiation software has been written to deal with function-defining programs which involve if-statements (or equivalent control structures) and more. Even if a function could be put into a straightforward program, it may be necessary to employ branching, for instance in Gauss-elimination for solving a parametric system of linear equations.

3. The if-problem

Consider two differentiable functions

$$r_1\colon U_1 \subseteq \mathbb{R}^n \to \mathbb{R} \quad \text{and} \quad r_2\colon U_2 \subseteq \mathbb{R}^n \to \mathbb{R}$$

and a Boolean function

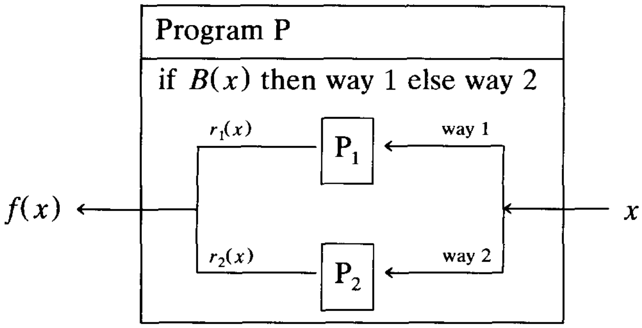
$$B\colon D \subseteq \mathbb{R}^n \to \{\text{true, false}\}.$$

Assume that it is possible to define a function f by

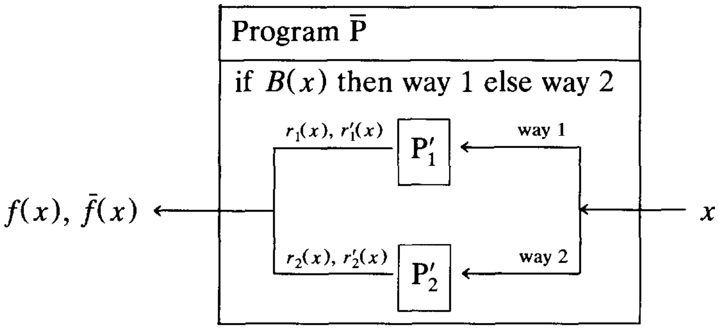
$$f\colon D \subseteq \mathbb{R}^n \to \mathbb{R}, \quad \text{with } f(x) := \begin{cases} r_1(x), & \text{if } B(x) = \text{true}, \\ r_2(x), & \text{if } B(x) = \text{false}. \end{cases}$$

Assume further that the function f is differentiable. (The previous assumption that r_1 and r_2 are differentiable does not imply that f is differentiable.)

For $x \in D$ we ask for $f(x)$ and $f'(x)$. And we would like to obtain these values by automatic differentiation. Therefore we assume that the functions r_1 and r_2 can be described by straightforward programs P_1 and P_2 in accordance with Fig. 1. Then the function f can be described by a program P of the form



How can we accomplish the transformation DIFF from P to a program that accepts $x \in D$ and produces the pair $f(x)$, $f'(x)$? A suggestive approach is: transform P_1 to P_1' and P_2 to P_2' as described in Section 2, and combine P_1' and P_2' to a program $\bar{P}$ as follows:



It seems plausible that $\bar{f}(x) = f'(x)$.

Example 3.1. We choose the following scenario: $D = \{x \mid x \in \mathbb{R}^2,\ 0 < x_1 < 2,\ 0 < x_2 < 2\}$,

$$a\colon D \to \mathbb{R}^{2,2}, \quad \text{with } a(x) = \begin{pmatrix} x_1 - x_2 & 1 \\ 10 & x_1 + x_2 \end{pmatrix},$$

$$b\colon D \to \mathbb{R}^2, \quad \text{with } b(x) = \begin{pmatrix} 100(x_1 + 2x_2) \\ 100(x_1 - 2x_2) \end{pmatrix},$$

$r_1\colon U_1 \to \mathbb{R}$ as defined by the program P_1, as follows:

$$r_1(x) \longleftarrow \boxed{\begin{array}{l} a(x) \longleftarrow \cdots \\ b(x) \longleftarrow \cdots \\ e(x) \longleftarrow a_{21}(x)/a_{11}(x) \\ \bar{a}_{22}(x) \longleftarrow a_{22}(x) - e(x) \cdot a_{12}(x) \\ \bar{b}_2(x) \longleftarrow b_2(x) - e(x) \cdot b_1(x) \\ r_1(x) \longleftarrow \bar{b}_2(x)/\bar{a}_{22}(x) \end{array}} \longleftarrow x,$$

r_2: $U_2 \to \mathbb{R}$ as defined by the program P_2, as follows:

$$r_2(x) \longleftarrow \boxed{\begin{array}{lll} a(x) & \longleftarrow & \cdots \\ b(x) & \longleftarrow & \cdots \\ r_2(x) & \longleftarrow & b_1(x)/a_{12}(x) \end{array}} \longleftarrow x,$$

$$B: D \to \{\text{true}, \text{false}\}, \quad \text{with } B(x) = \begin{cases} \text{true}, & \text{if } a_{11}(x) \neq 0, \\ \text{false}, & \text{if } a_{11}(x) = 0. \end{cases}$$

Then we obtain

$$f: D \to \mathbb{R}, \quad \text{with } f(x) = \begin{cases} r_1(x), & \text{if } a_{11}(x) \neq 0, \\ r_2(x), & \text{if } a_{11}(x) = 0. \end{cases}$$

A closer look at the function f and the programs P_1 and P_2 reveals that

$$f(x) = \text{second component of } z(x) = a(x)^{-1} \cdot b(x)$$

and that $f(x)$ is obtained by Gauss-elimination applied to the parametric linear system

$$a(x) \cdot z(x) = b(x), \quad \text{for every } x \in D.$$

Combining P_1 and P_2 as described above yields the program P. Furthermore, P_1 and P_2 can be considered as straightforward. So the transformation DIFF is applicable to generate the programs P_1' and P_2'. Their combination as described above yields the Program $\bar{P}$. Now choose $x = (1, 1)$. Then $\bar{P}$ produces

$$\bar{f}(1, 1) = [100, 200].$$

Unfortunately, the derivative value for this x is

$$f'(1, 1) = [170, 130].$$

This example shows that our approach (which is exactly what some automatic differentiation software does) failed.

Let us formulate the *if-problem*: how can we accomplish the transformation DIFF in the presence of if-statements?

4. Piecewise defined functions

A program that describes a function f and that involves statements of the form

if $B(x)$ then way 1 else way 2

is nothing but a piecewise definition of f. The example in the previous section shows that for a differentiable function f, defined piecewise using differentiable functions r_1 and r_2, it may happen that $f(x) = r_2(x)$ but $f'(x) \neq r_2'(x)$ for some x. To investigate this subtle situation (which causes the if-problem), we distinguish between a function f, a program P for f, and a representation R of f.

Definition 4.1. Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$. A family of triples $(r_i, D_i, U_i)_{i \in I}$ is called a *representation* of f iff

(1) $(D_i)_{i \in I}$ is a partition of D;
(2) $\forall i \in I$: $D_i \subseteq U_i \subseteq \mathbb{R}^n$;
(3) $\forall i \in I$: r_i is a function $U_i \to \mathbb{R}$;
(4) $\forall i \in I$, $x \in D_i$: $f(x) = r_i(x)$.

The function r_i represents the function f in D_i but for $x \in U_i \backslash D_i$ the value $r_i(x)$ is free. And it is this freedom of the r_i that causes the if-problem.

Example 4.2. Consider f: $\mathbb{R} \to \mathbb{R}$ with $f(x) = x^2$. Choose $D_1 = \mathbb{R} \backslash \{1\}$, $D_2 = \{1\}$, $U_1 = U_2 = \mathbb{R}$, r_1: $U_1 \to \mathbb{R}$ with $r_1(x) = x^2$ and r_2: $U_2 \to \mathbb{R}$ with $r_2(x) = x$. Then $(r_i, D_i, U_i)_{i \in \{1,2\}}$ is a representation of f.

Definition 4.3. Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ and a representation $R = (r_i, D_i, U_i)_{i \in I}$ of f. Let $j \in I$ and $x \in D_j$. We call R *neighborhood-consistent* at x iff there exists a set $N \subseteq \mathbb{R}^n$ such that

(1) N is a neighborhood of x;
(2) $N \subseteq D$;
(3) $N \subseteq U_j$;
(4) $\forall y \in N$: $f(y) = r_j(y)$.

If R is neighborhood-consistent at $x \in D_j$, then the function r_j represents the function f in x *and* in a certain neighborhood of x.

Proposition 4.4. *Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ and a representation $R = (r_i, D_i, U_i)_{i \in I}$ of f. Let $j \in I$ and $x \in D_j$. Then we have*

$$x \in \operatorname{int}(D_j) \quad \Rightarrow \quad R \text{ is neighborhood-consistent at } x.$$

Proof. Assume that x is in the interior of D_j. This means that there exists a neighborhood N of x with $N \subseteq D_j$. Now $N \subseteq D$, $N \subseteq U_j$, and f and r_j agree in N. Hence, R is neighborhood-consistent at x. □

Definition 4.5. Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ and a representation $R = (r_i, D_i, U_i)_{i \in I}$ of f. Let $j \in I$ and $x \in D_j$. Assume that the function f and the function r_j are differentiable at x. We call R *derivative-consistent* at x iff $f'(x) = r_j'(x)$.

If R is derivative-consistent at $x \in D_j$, then the function r_j represents the function f in x, and this point x may be the only point where the two functions agree.

Example 4.6. Consider $f: \mathbb{R} \to \mathbb{R}$ with $f(x) = x^2$. Choose $D_1 = \mathbb{R} \setminus \{1\}$, $D_2 = \{1\}$, $U_1 = U_2 = \mathbb{R}$, r_1: $U_1 \to \mathbb{R}$ with $r_1(x) = x^2$ and r_2: $U_2 \to \mathbb{R}$ with $r_2(x) = 2x - 1$. Then $(r_i, D_i, U_i)_{i \in \{1,2\}}$ is a representation of f, which is derivative-consistent at 1.

Proposition 4.7. *Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ and a representation $R = (r_i, D_i, U_i)_{i \in I}$ of f. Let $j \in I$ and $x \in D_j$. Assume that the function f and the function r_j are differentiable at x. Then we have*

R is neighborhood-consistent at x $\Rightarrow$ R is derivative-consistent at x.

Proof. Assume that R is neighborhood-consistent at x. Then the functions f and r_j agree in a neighborhood N of x, which means $f \mid N = r_j \mid N$. Therefore,

$$f'(x) = (f \mid N)'(x) = \left(r_j \mid N\right)'(x) = r_j'(x).$$

Hence, R is derivative-consistent at x. □

Proposition 4.8. *Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ and a representation $R = (r_i, D_i, U_i)_{i \in I}$ of f. Let $j \in I$ and $x \in D_j$. Assume that*

(1) *R is neighborhood-consistent at x;*
(2) *f is differentiable at x.*

Then we have

(3) *r_j is differentiable at x;*
(4) *$f'(x) = r_j'(x)$.*

Proof. (1) implies that f and r_j agree in a neighborhood N of x. Thus $f \mid N = r_j \mid N$. (2) implies that $f \mid N$ is differentiable at x and this in turn implies (3). Together with Proposition 4.7 we get (4). □

Proposition 4.9. *Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ and a representation $R = (r_i, D_i, U_i)_{i \in I}$ of f. Let $j \in I$ and $x \in D_j$. Assume that*

(1) *R is neighborhood-consistent at x;*
(2) *r_j is differentiable at x.*

Then we have

(3) *f is differentiable at x;*
(4) *$f'(x) = r_j'(x)$.*

Proof. (1) implies that f and r_j agree in a neighborhood N of x. Thus $f \mid N = r_j \mid N$. (2) implies that $r_j \mid N$ is differentiable at x and this in turn implies (3). Together with Proposition 4.7 we get (4). □

Proposition 4.10. *Consider a function f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ and a representation $R = (r_i, D_i, U_i)_{i \in I}$ of f. Let $j \in I$ and $W_j \subseteq D_j$. Assume that*

(1) *$r_j \mid W_j$ is differentiable.*

Then we have

(2) $f \mid W_j$ *is differentiable*;

(3) $\forall x \in W_j$: $f'(x) = r_j'(x)$.

Proof. (1) implies that W_j is open. f and r_j agree in W_j, so (2) is obvious. Consider $x \in W_j$. This x is an interior point of W_j and an interior point of D_j as well. From Proposition 4.4 we know that R is neighborhood-consistent at x. Together with Proposition 4.9 we get (3). □

Proposition 4.11. *Consider a function* f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ *and a representation* $R = (r_i, D_i, U_i)_{i \in I}$ *of* f. *Let* $j \in I$ *and* $x \in D_j$. *Assume that*

(1) f *is differentiable at* x;

(2) r_j *is differentiable at* x;

(3) $\forall y \in D \cap U_j$: $f(y) = r_j(y)$.

Then we have

(4) $f'(x) = r_j'(x)$.

Proof. (1) implies $x \in \mathrm{int}(D)$ and (2) implies $x \in \mathrm{int}(U_j)$. Hence, there exists a neighborhood N of x with $N \subseteq D \cap U_j$. Because of (3), f and r_j agree in N, so R is neighborhood-consistent at x. With Proposition 4.8 or 4.9 (either one) we get (4). □

Proposition 4.12. *Consider a function* f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ *and a representation* $R = (r_i, D_i, U_i)_{i \in I}$ *of* f. *Assume that*

(1) f *is continuously differentiable*;

(2) $\forall i \in I$: r_i *is differentiable*.

Define the function

$$\hat{r}: D \subseteq \mathbb{R}^n \to \mathbb{R}^{1,n}, \quad \textit{with } \hat{r}(x) := r_i'(x), \ \textit{for } x \in D_i.$$

Then we have

$$R \textit{ is derivative-consistent in } D \ \Rightarrow \ \hat{r} \textit{ is continuous}.$$

Proof. Assume that R is derivative-consistent in D. Let $x \in D$. Assume that $x \in D_j$ for some $j \in I$. Let $x_1, x_2, \dots$ be an arbitrary sequence in D with $x = \lim(x_m)$. To fix notation we set $x_m \in D_{j(m)}$. Now we get

$$\hat{r}(x) = r_j'(x) = f'(x) = f'(\lim(x_m)) = \lim(f'(x_m)) = \lim\left(r_{j(m)}'(x_m)\right) = \lim(\hat{r}(x_m)).$$

This shows that $\hat{r}$ is continuous at x. □

Proposition 4.13. *Consider a function* f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ *and a representation* $R = (r_i, D_i, U_i)_{i \in I}$ *of* f. *Assume that*

(1) f *is continuously differentiable*;

(2) $\forall i \in I$: r_i *is differentiable*.

Define the function

$$\hat{r}: D \subseteq \mathbb{R}^n \to \mathbb{R}^{1,n}, \quad \textit{with } \hat{r}(x) := r_i'(x), \ \textit{for } x \in D_i.$$

Assume further that

(3) $\forall x \in D, \exists k \in I$: $x \in \overline{D}_k$ *and* D_k *open*.

Then we have

$$\hat{r} \text{ is continuous} \Rightarrow R \text{ is derivative-consistent in } D.$$

Proof. Assume that $\hat{r}$ is continuous. Let $x \in D$. Assume that $x \in D_j$ for some $j \in I$. (3) implies that there exists $k \in I$ and a sequence $x_1, x_2, \ldots$ in D_k with $x = \lim(x_m)$. Note that $x_m \in \operatorname{int}(D_k)$ and that $f'(x_m) = r_k'(x_m)$ for $m = 1, 2, \ldots$ because of Propositions 4.4 and 4.7. Now we get

$$f'(x) = f'(\lim(x_m)) = \lim(f'(x_m)) = \lim(r_k'(x_m)) = \lim(\hat{r}(x_m)) = \hat{r}(\lim(x_m))$$
$$= \hat{r}(x) = r_j'(x).$$

This shows that R is derivative-consistent at x. □

Having a representation $(r_i, D_i, U_i)_{i \in I}$ of a function f it is desirable to obtain

$$x \in D_j \Rightarrow f'(x) = r_j'(x).$$

If this is not possible for some reason, we may try

$$x \in D_j \Rightarrow \exists k \in I: f'(x) = r_k'(x).$$

Proposition 4.14. *Consider a function* f: $D \subseteq \mathbb{R}^n \to \mathbb{R}$ *and a representation* $R = (r_i, D_i, U_i)_{i \in I}$ *of* f. *Assume that*

(1) f *is continuously differentiable*;

(2) $x \in D_j$ *for some* $j \in I$.

Assume further that there exists $k \in I$ *such that*

(3) $x \in U_k$;

(4) $x = \lim(x_m)$ *with* $x_m \in D_k$ *for* $m = 1, 2, \ldots$;

(5) r_k *is continuously differentiable*;

(6) R *is derivative-consistent in* D_k.

Then we have

(7) $f'(x) = r_k'(x)$.

Proof. The index j in (2) is not relevant, it only serves to show that x is in the subset D_j and that $f'(x)$ is expressed using r_k where j and k need not coincide. We get

$$f'(x) = f'(\lim(x_m)) = \lim(f'(x_m)) = \lim(r_k'(x_m)) = r_k'(\lim(x_m)) = r_k'(x). \quad \square$$

5. Partial solution for the if-problem

Normally one does not care much about the domain of a function. But in Example 3.1 and in the propositions of Section 4 we have seen that being aware of "which function has which domain" may be crucial.

Let us consider the if-problem in a restricted setting and let us carefully record the domains of functions used.

Consider the functions

$$\alpha\colon A \subseteq \mathbb{R}^n \to \mathbb{R}^m, \qquad c\colon C \subseteq \mathbb{R}^n \to \mathbb{R}, \qquad \omega_1\colon W_1 \subseteq \mathbb{R}^m \to \mathbb{R}, \qquad \omega_2\colon W_2 \subseteq \mathbb{R}^m \to \mathbb{R}.$$

Assume that these functions (or rather programs for these functions) only use the rational operations $+$, $-$, $\cdot$, $/$ and differentiable library functions (see Section 2). Assume further that the domains A, C, W_1, W_2 are maximal, that is, whenever a function can be evaluated for a particular argument, then this argument belongs to the function's domain. This implies that the sets A, C, W_1, W_2 are open and the functions α, c, ω_1, ω_2 are differentiable.

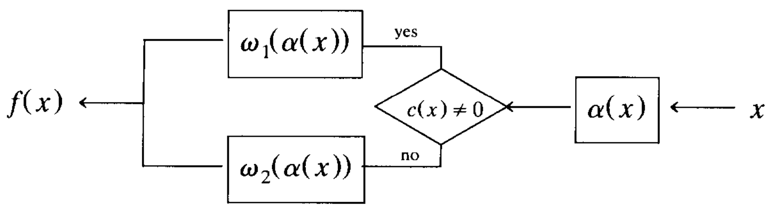
Define the sets

$$D_1 := \{x \mid x \in A,\ x \in C,\ c(x) \neq 0,\ \alpha(x) \in W_1\},$$
$$D_2 := \{x \mid x \in A,\ x \in C,\ c(x) = 0,\ \alpha(x) \in W_2\},$$
$$D := D_1 \cup D_2.$$

Assume that $D \neq 0$ and define the function

$$f\colon D \subseteq \mathbb{R}^n \to \mathbb{R}, \quad \text{with } f(x) := \begin{cases} \omega_1(\alpha(x)), & \text{for } c(x) \neq 0, \\ \omega_2(\alpha(x)), & \text{for } c(x) = 0. \end{cases}$$

This function f can be depicted in the diagram



and described in the (rudimentary) program

compute $\alpha(x)$
compute $c(x)$
if $c(x) \neq 0$ then way 1 else way 2
way 1: $f(x) = \omega_1(\alpha(x))$
way 2: $f(x) = \omega_2(\alpha(x))$

Define the sets

$$U_1 := \{x \mid x \in A,\ \alpha(x) \in W_1\}, \qquad U_2 := \{x \mid x \in A,\ \alpha(x) \in W_2\}.$$

Assume that $U_1 \neq \emptyset$ and $U_2 \neq \emptyset$ and define the functions

$$r_1\colon U_1 \subseteq \mathbb{R}^n \to \mathbb{R}, \quad \text{with } r_1(x) := \omega_1(\alpha(x)), \qquad r_2\colon U_2 \subseteq \mathbb{R}^n \to \mathbb{R}, \quad \text{with } r_2(x) := \omega_2(\alpha(x)).$$

Then $(r_i, D_i, U_i)_{i \in \{1,2\}}$ is a representation of f.

Now choose $x_* \in \mathbb{R}^n$. Assume that $f(x_*)$ can be evaluated. Then $x_* \in D$. This means in particular that $c(x_*)$ can be evaluated. Let us consider the case $c(x_*) \neq 0$. Here $x_* \in D_1$, and furthermore, $x_* \in \text{int}(A)$, $x_* \in \text{int}(C)$, and $x_* \in \text{int}(D_1)$. With Propositions 4.4 and 4.7 we obtain

$$f'(x_*) = r_1'(x_*).$$

Since r_1 is straightforward, $r_1'(x_*)$ can be obtained as described in Section 2 by automatic differentiation.

References

[1] Th. Beck, Automatisches Differenzieren von Algorithmen, Dissertation, Techn. Univ. München, 1991.

[2] H. Fischer, Automatic differentiation of characterizing sequences, *J. Comput. Appl. Math.* **28** (1989) 181–185.

[3] H. Fischer, Automatic differentiation of the vector that solves a parametric linear system, *J. Comput. Appl. Math.* **35** (1991) 169–184.

[4] H. Fischer, Special problems in automatic differentiation, in: A. Griewank and G.F. Corliss, Eds., *Automatic Differentiation of Algorithms: Theory, Implementation, and Application* (SIAM, Philadelphia, PA, 1991) 43–50.

[5] D.W. Juedes, A taxonomy of automatic differentiation tools, in: A. Griewank and G.F. Corliss, Eds., *Automatic Differentiation of Algorithms: Theory, Implementation, and Application* (SIAM, Philadelphia, PA, 1991) 315–329.

[6] G.P. McCormick, *Nonlinear Programming* (Wiley, New York, 1983).

[7] L.B. Rall, *Automatic Differentiation: Techniques and Applications*, Lecture Notes in Comput. Sci. **120** (Springer, Berlin, 1981).

[8] L.B. Rall, Differentiation in PASCAL-SC: type GRADIENT, *ACM Trans. Math. Software* **10** (2) (1984) 161–184.