

# 1. 数式解析を行うプログラム

## — FAMILIA —

魚木五夫 鈴木喜代司 渡辺 剛 (東洋工業)

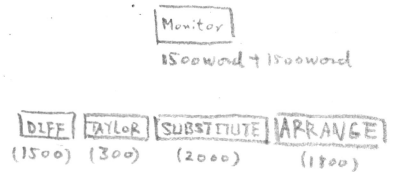
### 1. はじめに

計算機の応用の一つに記号計算 (Symbol Manipulation)<sup>[1]</sup> と言われるものがある。例えば、数式について (1) 整理する, (2) 展開する, (3) 因数分解する, (4) 微分する<sup>[1]</sup>, (5) 不定積分する<sup>[2]</sup>, (6) 代入する, 等々がある。上に掲げたものは各々単独に使われることもあるが、相互の組合せで用いる方がより実用的である。例えば、ある数式の微係数を求めるには (4) と (6) が一緒に用いられることになる。

ここでは、数式についての

- (1) 微分プログラム DIFF(differentiation)
- (2) 代入プログラム SUBSTIT(substitution)
- (3) 整理プログラム ARRANGE(arrangement)

について 中間結果を報告する。

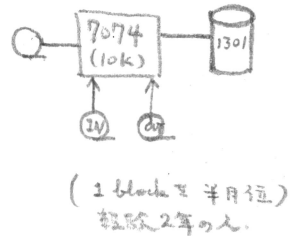


### 2. 微分プログラム DIFF

#### 2.1 Source Statement

一般の形式は

```
DIFF (i1, i2 ..... in), variable
expression
```



ここで、variable に何の変数について微分するかの変数名を、expression には、微分した数式(modeの制限がないFORTRAN型のExpression)を書く、 $i_1, i_2, \dots, i_n$  には正の整数を書いて、 $i_1$ 回、 $i_2$ 回、.....、 $i_n$ 回の微分が欲しいことを表示する。その他、DIFF および区切符号は正しく書く。

例えば `DIFF (1, 3), X`  
`1 / ((X-A)**2 + B**2)**N`

1回だけの微分のとき、指定(1)を省略できる。

今もう一つの形式は

```
DIFF (i = n1, n2, n3), variable
expression
```

これは、微分の回数指定がDO型にできる。その他は一般の形式と同じである

## 2.2 DIFF Processor

与えられた数式、Table 1 に掲げた微分公式に変形して微分を行つている。Processor を大別して、Phase 1, 2, 3 となつている。

(Phase 1) 与えられた数式を  $n$  行3列のM-matrixに変換して微分公式が直接適用できるようにする。M-matrix の各行の構成はOperand, Operator, Operand, である。変換するとき、数式を右側からscanしてoperatorの優先度の高い順に、M-matrix に乗せる。例えば

$$A * X ** 2 + B * X + \text{SINF} (C * X)$$

に対するM-matrixは

row	operand	operator	operand
1	C	*	X
2	SINF	•	$r_1$
3	B	*	X
4	X	**	2
5	A	*	$r_4$
6	$r_5$	+	$r_3$
7	$r_6$	+	$r_2$

(Phase 2) M-matrix の各行について微分公式を適用して、実際に微分する。その結果を  $n$  行3列のD-matrix に乗せる。このときM-matrix の各行がD-matrixを作る。

(Phase 3) 微分した結果がD-matrixに乗っているのを、微分した数式として組み上げる。

## 2.3 実験結果

## 3. 代入プログラム SUBSTIT

### 3.1 Source Statement

一般の形式は

SUBSTIT, $V_1 = e_1, V_2 = e_2, \dots, V_n = e_n$ expression
---

ここで  $V_1, V_2$  等は、代入したい変数名、 $e_1, e_2$  等は  $V_1, V_2$  等に代入する数式 (mode の制限がないFORTRAN 型の Expression)、expression には代入したい数式を書く。

### 3.2 SUBSTIT Processor

このProcessor は DIFF Processor の考えと同じである。

(Phase 1) 与えられた数式に対して M-matrix を同じ手法で作る。また代入する数式  $e_1, e_2, \dots$  に対しては M-matrix と同じ型の S-table を作る。

(Phase 2) M-matrix の各行について微分公式を適用する代りに S-table を参照して代入を実際に行う。その結果は M-matrix 自身に乘せる。

(Phase 3) 代入した結果が M-matrix に乗っているのを、代入した数式として組み上げる。

### 3.3 実験の結果

## 4. 整理プログラム ARRANGE

### 4.1 Source Statement

一般の形式は

```
ARRANGE, Variable
expression
```

ここで、Variable には、どの変数について整理するかの変数名をかく、特に指定したい変数がないとき省略する。指定されない変数は、辞書式且つ昇巾の順序に並べる。expression には整理したい数式を書く。FORTRAN型の Expression と違っているところは

- (1) mode の制限がない。
- (2) exponent は integer に制限
- (3) function は使用できない。

これらの諸制限は暫定的なもので、最終的には除去したい。

### 4.2 ARRANGE Processor

与えられた数式を整理するのであるが、ここで「整理」の意味を規定する。

- (1) 括弧 ( ) を除去する。
- (2) 変数は辞書式且つ昇巾の順序に並べる。また、項の相互についてもこれに準ずる。
- (3) 同<sup>類</sup>項をまとめる。
- (4) 分子と分母に数式を二分割する。

Processor を大別して Phase 1~9 となつている。

(Phase 1) 数式の各々の項を refined term に変換する。refined term とは

係数, 変数, 指数, .....

例えば  $4 * A ** 2 + 12 * A * B + 9 * B * 5 * C$  は変換されて,  $4, A, 2 + 12, A, 1, B, 1 + 9, B, 1 * 5, C, 1$  となる.

(Phase 2) operator  $-$  を係数の符号に含ませて, operator  $+$  に置きかえることにより, operator  $-$  を表面上見えなくする.

(Phase 3) operator  $/$  を指数の符号に含ませて, operator  $*$  に置きかえることにより, operator  $/$  を表面上見えなくする.

(Phase 4) operator  $**$  を表面上なくする. このとき分母になる候補を括弧  $[ ]$  でとし. 例えば,  $(Coe, Var, Exp, \dots) ** N$  について  $(Coe^N, Var, Exp * N, \dots)$ , また数式  $A$  が operator  $+$  を含んでいるとして  $(A) ** N$  について,  $N \geq 0$  なら  $(A) * (A) * \dots$ ,  $N < 0$  なら  $[(A) * (A) * \dots]$  と変換する.

(Phase 5) 数式を分子と分母に二分劃する.

(Phase 6) 分子と分母について括弧  $(, )$  を除去する.

(Phase 7) 項の中で変数相互の順序を入れかえる. 変数の中に同じものがあれば, 指数の加算を行って一個にまとめる. 指数が零の変数は除去する.

(Phase 8) 項相互の順序を入れかえる. 同類項は係数の加算で一個にまとめる. 係数が零のものは除去する.

(Phase 9) 係数及び指数が1であるとき, これを除去する. 係数及び指数が minus のとき operator  $-$  を表面に出す. refined term には operator  $*$  と  $**$  を挿入する. 括弧  $[ ]$  は operator  $/$  にかえる.

#### 4.3 実験の結果

### 5. 結 び

\* DIFF, SUBSTIT, ARRANGE を macro instruction とみて使用できるようにする.

\* TAYLOR, FACTOR 等の macro も考えている.

特分

行記りになりや311 (1) 文獻 [2]

## 参考文献

- [1] HANSON, J. W., CAVINESS, J. S., and JOSEPH, C. "Analytic Differentiation by Computer". Comm. ACM 7, 5, 349-355
- [2] SLAGLE, J. R. "A Heuristic Program that Solves Symbolic Integration Problems in Freshman Calculus" J. ACM 10, 4, 507-520
- [3] CLAPP, L. G., and KAIN, R. Y. "A Computer Aid for Symbolic Mathematics". Proc. Fall Joint Comp. Conf. 1963, 509-517
- [4] WENGERT, P. E. "A Simple Automatic Derivative Evaluation Program" Comm. ACM 7, 8, 463-464
- [5] WILKINS, R. D. "Investigation of a New Analytical Method for Numerical Derivative Evaluation" Comm. ACM 7, 8, 465-471

TABLE 1 微分公式集

1	$(\Gamma \pm \Delta)' = \Gamma' \pm \Delta'$
2	$(\Gamma * \Delta)' = \Gamma' * \Delta + \Gamma * \Delta'$
3	$(\Gamma / \Delta)' = (\Gamma' * \Delta - \Gamma * \Delta') / (\Delta ** 2)$
4	$(\Gamma ** \Delta) = \Gamma ** \Delta * (\Delta * \Gamma' / \Gamma + \Delta' * \text{LOGF}(\Gamma))$
5	$(\text{LOGF}(\Gamma))' = \Gamma' / \Gamma$
6	$(\text{EXPF}(\Gamma))' = \text{EXPF}(\Gamma) * \Gamma'$
7	$(\text{SINF}(\Gamma))' = \text{COSF}(\Gamma) * \Gamma'$
8	$(\text{COSF}(\Gamma))' = -\text{SINF}(\Gamma) * \Gamma'$
9	$(\text{TANF}(\Gamma))' = \text{SECF}(\Gamma) ** 2 * \Gamma'$
10	$(\text{COTF}(\Gamma))' = -\text{CSCF}(\Gamma) ** 2 * \Gamma'$
11	$(\text{SECF}(\Gamma))' = \text{TANF}(\Gamma) * \text{SECF}(\Gamma) * \Gamma'$
12	$(\text{CSCF}(\Gamma))' = -\text{CSCF}(\Gamma) * \text{COTF}(\Gamma) * \Gamma'$
13	$(\text{ASINF}(\Gamma))' = \Gamma' / \text{SQRTF}(1 - \Gamma ** 2)$
14	$(\text{ACOSF}(\Gamma))' = -\Gamma' / \text{SQRTF}(1 - \Gamma ** 2)$
15	$(\text{ATANF}(\Gamma))' = \Gamma' / (1 + \Gamma ** 2)$
16	$(\text{ACOTF}(\Gamma))' = -\Gamma' / (1 + \Gamma ** 2)$
17	$(\text{ASECF}(\Gamma))' = \Gamma' / (\Gamma * \text{SQRTF}(1 - \Gamma ** 2))$
18	$(\text{ACSCF}(\Gamma))' = -\Gamma' / (\Gamma * \text{SQRTF}(1 - \Gamma ** 2))$
19	$(\text{SINHF}(\Gamma))' = \text{COSHF}(\Gamma) * \Gamma'$
20	$(\text{COSHF}(\Gamma))' = \text{SINHF}(\Gamma) * \Gamma'$
21	$(\text{TANHF}(\Gamma))' = \text{SECHF}(\Gamma) ** 2 * \Gamma'$
22	$(\text{COTHF}(\Gamma))' = -\text{CSCHF}(\Gamma) ** 2 * \Gamma'$
23	$(\text{SECHF}(\Gamma))' = -\text{SECHF}(\Gamma) * \text{TANHF}(\Gamma) * \Gamma'$
24	$(\text{CSCHF}(\Gamma))' = -\text{CSCHF}(\Gamma) * \text{COTHF}(\Gamma) * \Gamma'$
25	$(\text{ASNHF}(\Gamma))' = \Gamma' / \text{SQRTF}(1 + \Gamma ** 2)$
26	$(\text{ACSHF}(\Gamma))' = -\Gamma' / \text{SQRTF}(1 - \Gamma ** 2)$
27	$(\text{ATNHF}(\Gamma))' = \Gamma' / (1 - \Gamma ** 2)$
28	$(\text{ACTHF}(\Gamma))' = \Gamma' / (1 - \Gamma ** 2)$
29	$(\text{ASCHF}(\Gamma))' = -\Gamma' / (\Gamma * \text{SQRTF}(1 - \Gamma ** 2))$
30	$(\text{ACCHF}(\Gamma))' = -\Gamma' / (\Gamma * \text{SQRTF}(1 + \Gamma ** 2))$
31	$(\text{SQRTF}(\Gamma))' = -\Gamma' / (2 * \text{SQRTF}(\Gamma))$

TABLE 1

Function Name	Mathematical Representation
LOGF (X)	$\log_e e^x$
EXPF (X)	$e^x$
SINF (X)	$\sin x$
COSF (X)	$\cos x$
TANF (X)	$\tan x$
COTF (X)	$\cot x$
SECF (X)	$\sec x$
CSCF (X)	$\operatorname{cosec} x$
ASINF (X)	$\operatorname{Sin}^{-1} x$
ACOSF (X)	$\operatorname{Cos}^{-1} x$
ATANF (X)	$\operatorname{Tan}^{-1} x$
ACOTF (X)	$\operatorname{Cot}^{-1} x$
ASECF (X)	$\operatorname{Sec}^{-1} x$
ACSCF (X)	$\operatorname{Cosec}^{-1} x$
SINHF (X)	$\sin h x$
COSHF (X)	$\cos h x$
TANHF (X)	$\tan h x$
COTHF (X)	$\cot h x$
SECHF (X)	$\sec h x$
CSCHF (X)	$\operatorname{cosec} h x$
ASNHF (X)	$\operatorname{Sin} h^{-1} x$
ACSHF (X)	$\operatorname{Cos} h^{-1} x$
ATNHF (X)	$\operatorname{Tan} h^{-1} x$
ACTHF (X)	$\operatorname{Cot} h^{-1} x$
ASCHF (X)	$\operatorname{Sec} h^{-1} x$
ACCHF (X)	$\operatorname{Cosec} h^{-1} x$
SQRTF (X)	$\sqrt{x}$

本 PDF ファイルは 1965 年発行の「第 6 回プログラミング—シンポジウム報告集」をスキャンし、項目ごとに整理して、情報処理学会電子図書館「情報学広場」に掲載するものです。

この出版物は情報処理学会への著作権譲渡がなされていませんが、情報処理学会公式 Web サイトの [https://www.ipsj.or.jp/topics/Past\\_reports.html](https://www.ipsj.or.jp/topics/Past_reports.html) に下記「過去のプログラミング・シンポジウム報告集の利用許諾について」を掲載して、権利者の検索をおこないました。そのうえで同意をいただいたもの、お申し出のなかったものを掲載しています。

#### 過去のプログラミング・シンポジウム報告集の利用許諾について

情報処理学会発行の出版物著作権は平成 12 年から情報処理学会著作権規程に従い、学会に帰属することになっています。

プログラミング・シンポジウムの報告集は、情報処理学会と設立の事情が異なるため、この改訂がシンポジウム内部で徹底しておらず、情報処理学会の他の出版物が情報学広場 (=情報処理学会電子図書館) で公開されているにも拘らず、古い報告集には公開されていないものが少からずありました。

プログラミング・シンポジウムは昭和 59 年に情報処理学会の一部門になりましたが、それ以前の報告集も含め、この度学会の他の出版物と同様の扱いにしたいと考えます。過去のすべての報告集の論文について、著作権者（論文を執筆された故人の相続人）を探し出して利用許諾に関する同意を頂くことは困難ですので、一定期間の権利者搜索の努力をしたうえで、著作権者が見つからない場合も論文を情報学広場に掲載させていただきたいと思えます。その後、著作権者が発見され、情報学広場への掲載の継続に同意が得られなかった場合には、当該論文については、掲載を停止致します。

この措置にご意見のある方は、プログラミング・シンポジウムの辻尚史運営委員長 ([tsuji@math.s.chiba-u.ac.jp](mailto:tsuji@math.s.chiba-u.ac.jp)) までお申し出ください。

加えて、著作権者について情報をお持ちの方は事務局まで情報をお寄せくださいますようお願い申し上げます。

期間：2020 年 12 月 18 日～2021 年 3 月 19 日

掲載日：2020 年 12 月 18 日

プログラミング・シンポジウム委員会

情報処理学会著作権規程

<https://www.ipsj.or.jp/copyright/ronbun/copyright.html>