ALTRAN Programs for SIGSAM Problem #6

by

A. D. Hall and S. C. Johnson
Bell Laboratories
Murray Hill, New Jersey 07974

I. Introduction

In [1], it is suggested that an algebraic manipulation system be used to gain some insight into the following problem:

Given

$$F_i(x) = \prod_{j=1}^{i} \frac{1}{(\lambda^2-(x+a_j)^2)}$$

find

$$S_i = \sum_{j=1}^{i} (A_j^{(i)} + B_j^{(i)}) \text{ for } i = 1,2,\ldots$$

where

$$A_j^{(i)} = F_i(x)\cdot(\lambda-x-a_j)\big|_{x=\lambda-a_j}$$

$$B_j^{(i)} = F_i(x)\cdot(\lambda+x+a_j)\big|_{x=-\lambda-a_j}$$

In this paper we briefly present two ALTRAN programs for computing S_i, $i=1,2,\ldots$.

Because the amount of CPU time required to compute S_i grows so rapidly we obtained only S_1, S_2, S_3 and S_4, although we believe these programs could compute S_5. While the results do not provide much insight into the proposed problem, they clearly demonstrate the advantage of using a factored representation for rational functions in a symbolic algebra system.

II. Programs

The first program computes the S_i directly using the algorithm suggested by the formulation of the problem:

```
PROCEDURE MAIN

INTEGER I, J, N = 4
LONG ALGEBRAIC ( L:N**2, A(1:N):8*N, X:1 ) F, S

DO I = 1,N

  F = 1
  DO J = 1, I
    F = F / ( (L-X-A(J)) * (L+X+A(J)) )
    DOEND

  S = 0
  DO J = 1, I
    S = S + (F*(L-X-A(J)))(X=L-A(J)) + (F*(L+X+A(J)))(X=-L-A(J))
    DOEND

  WRITE S
  DOEND

END
```

A faster program for computing the S_i can be obtained by observing that

$$A_j^{(i)} = \frac{A_j^{(i-1)}}{(a_j-a_i)(2\lambda-a_j+a_i)}, \quad j < i$$

$$A_i^{(i)} = \frac{F_{i-1}(\lambda-a_i)}{2\lambda}$$

and by proving that

$$\sum_{j=1}^{i} A_j^{(i)} = \sum_{j=1}^{i} B_j^{(i)}$$

all of which are left as exercises for the reader.

In the program below, AT(J), $1 \leq J \leq I$, holds the values of $A_j^{(i)}$ needed for the computation of S_i.

III. Results

Both programs were run using a factored and unfactored internal representation and about 10,000 words of workspace. Using an unfactored representation the value of S_4 can not be computed with this small a workspace. (More than 30,000 words would be needed). The incremental CPU times in seconds for the Honeywell 6070 are shown in the table below.

Program 1

	Factored	Unfactored
S_1	0.7	1.1
S_2	3.3	7.3
S_3	19.5	54.0
S_4	512.0	—

Program 2

	Factored	Unfactored
S_1	0.4	0.4
S_2	1.1	2.3
S_3	5.1	13.2
S_4	56.2	—

The value of S_4 is shown below as printed by ALTRAN, using the factored representation.

```
PROCEDURE MAIN

INTEGER I, J, N = 4
LONG ALGEBRAIC ( L:N**2, A(1:N):3*N, X:1 ) F, S, AT
ARRAY (N) AT

F = 1

DO I = 1,N

  DO J = 1, I-1
    AT(J) = AT(J)/( (A(J) - A(I)) * (2*L - A(J) + A(I)) )
    DOEND

  AT(I) = F(X=L-A(I))/(2*L)

  S = 0
  DO J = 1, I
    S = S + AT(J)
    DOEND

  S = 2*S
  F = F / ( (L-X-A(I)) * (L+X+A(I)) )

  WRITE S
  DOEND

END
```

```
* S

2 * ( 640*L**6 - 144*L**4*A(1)**2 + 96*L**4*A(1)*A(2) + 96*L**4*A(1)*A(3) +
96*L**4*A(1)*A(4) - 144*L**4*A(2)**2 + 96*L**4*A(2)*A(3) + 96*L**4*A(2)*A(4) -
144*L**4*A(3)**2 + 96*L**4*A(3)*A(4) - 144*L**4*A(4)**2 + 12*L**2*A(1)**4 -
16*L**2*A(1)**3*A(2) - 16*L**2*A(1)**3*A(3) - 16*L**2*A(1)**3*A(4) +
16*L**2*A(1)**2*A(2)**2 + 8*L**2*A(1)**2*A(2)*A(3) + 8*L**2*A(1)**2*A(2)*A(4) +
16*L**2*A(1)**2*A(3)**2 + 8*L**2*A(1)**2*A(3)*A(4) + 16*L**2*A(1)**2*A(4)**2 -
16*L**2*A(1)*A(2)**3 + 8*L**2*A(1)*A(2)**2*A(3) + 8*L**2*A(1)*A(2)**2*A(4) +
8*L**2*A(1)*A(2)*A(3)**2 - 48*L**2*A(1)*A(2)*A(3)*A(4) + 8*L**2*A(1)*A(2)*
A(4)**2 - 16*L**2*A(1)*A(3)**3 + 8*L**2*A(1)*A(3)**2*A(4) + 8*L**2*A(1)*A(3)*
A(4)**2 - 16*L**2*A(1)*A(4)**3 + 12*L**2*A(2)**4 - 16*L**2*A(2)**3*A(3) -
16*L**2*A(2)**3*A(4) + 16*L**2*A(2)**2*A(3)**2 + 8*L**2*A(2)**2*A(3)*A(4) +
16*L**2*A(2)**2*A(4)**2 - 16*L**2*A(2)*A(3)**3 + 8*L**2*A(2)*A(3)**2*A(4) +
8*L**2*A(2)*A(3)*A(4)**2 - 16*L**2*A(2)*A(4)**3 + 12*L**2*A(3)**4 -
16*L**2*A(3)**3*A(4) + 16*L**2*A(3)**2*A(4)**2 - 16*L**2*A(3)*A(4)**3 +
12*L**2*A(4)**4 - A(1)**4*A(2)**2 + A(1)**4*A(2)*A(3) + A(1)**4*A(2)*A(4) -
A(1)**4*A(3)**2 + A(1)**4*A(3)*A(4) - A(1)**4*A(4)**2 + 2*A(1)**3*A(2)**3 -
A(1)**3*A(2)**2*A(3) - A(1)**3*A(2)**2*A(4) - A(1)**3*A(2)*A(3)**2 - A(1)**3*A(2)*
A(4)**2 + 2*A(1)**3*A(3)**3 - A(1)**3*A(3)**2*A(4) - A(1)**3*A(3)*A(4)**2 +
2*A(1)**3*A(4)**3 - A(1)**2*A(2)**4 - A(1)**2*A(2)**3*A(3) - A(1)**2*A(2)**3*
A(4) + 3*A(1)**2*A(2)**2*A(3)**2 + 3*A(1)**2*A(2)**2*A(4)**2 - A(1)**2*A(2)*
A(3)**3 - A(1)**2*A(2)*A(4)**3 - A(1)**2*A(3)**4 - A(1)**2*A(3)**3*A(4) +
3*A(1)**2*A(3)**2*A(4)**2 - A(1)**2*A(3)*A(4)**3 - A(1)**2*A(4)**4 +
A(1)*A(2)**4*A(3) + A(1)*A(2)**4*A(4) - A(1)*A(2)**3*A(3)**2 - A(1)*A(2)**3*
A(4)**2 - A(1)*A(2)**2*A(3)**3 - A(1)*A(2)**2*A(4)**3 + A(1)*A(2)*A(3)**4 +
A(1)*A(2)*A(4)**4 + A(1)*A(3)**4*A(4) - A(1)*A(3)**3*A(4)**2 - A(1)*A(3)**2*
A(4)**3 + A(1)*A(3)*A(4)**4 - A(2)**4*A(3)**2 + A(2)**4*A(3)*A(4) -
A(2)**4*A(4)**2 + 2*A(2)**3*A(3)**3 - A(2)**3*A(3)**2*A(4) - A(2)**3*A(3)*
A(4)**2 + 2*A(2)**3*A(4)**3 - A(2)**2*A(3)**4 - A(2)**2*A(3)**3*A(4) +
3*A(2)**2*A(3)**2*A(4)**2 - A(2)**2*A(3)*A(4)**3 - A(2)**2*A(4)**4 +
A(2)*A(3)**4*A(4) - A(2)*A(3)**3*A(4)**2 - A(2)*A(3)**2*A(4)**3 +
A(2)*A(3)*A(4)**4 - A(3)**4*A(4)**2 + 2*A(3)**3*A(4)**3 - A(3)**2*A(4)**4 ) /
( ( L ) * ( 2*L - A(3) + A(4) ) * ( 2*L + A(3) - A(4) ) * ( 2*L - A(2) + A(4) ) *
( 2*L + A(2) - A(4) ) * ( 2*L - A(2) + A(3) ) * ( 2*L + A(2) - A(3) ) *
( 2*L - A(1) + A(4) ) * ( 2*L + A(1) - A(4) ) * ( 2*L - A(1) + A(3) ) *
( 2*L + A(1) - A(3) ) * ( 2*L - A(1) + A(2) ) * ( 2*L + A(1) - A(2) ) )
```

Continued on page 36

CONTENTS:

1. The natural input to the system
1.1. Introduction
1.2. Basic FORMAC capability's of NETFORM
1.3. The ideas of formula manipulation
..... the sum of squares
1.4. Formulation of linear equations
..... the sum of squares
1.5. Discussion

2. The linear equations solver
2.1. Introduction
2.2. Canonical expressions
2.3. Sparce matrix expansion
2.4. Proparty's and applications of sparce matrix expansion
2.5. NETFORM - instructions for the solution of a set of linear equations :
Two examples

3. Analysis of networks
3.1. Introduction
3.2. The description of an electrical network via a system of linear equations
3.3. Definition of network elements
3.4. The description of an electrical network via ist structure
3.5. Discussion
3.5.1. The uniqueness of a solution
3.5.2. Examples of singularity
3.5.3. The degree of the characteristic equation
3.5.4. The state variables
Litterature.

LITTERATURE:

1 Tobey, R.G. - Bobrow, R.J. - Zilles, S.
Automatic Simplification in FORMAC,
Proc. of fall joint computer conference,
Spartan books vol. 27 nov. 1965

PL/1 - FORMAC Symbolic Mathematics Interpreter 360D - 03.3. 004, 'IBM manual'.

Bobrow, D.G.
Symbol Manipulation languages and techniques,
Proc. of ifip working conference on symbol manipulation languages.

2 J. Mucha.
Zur nummerischen und nichtnummerischen Empfindlichkeits analyse linearer elektrische Netzwerke, Ph. D. thesis Aachen.

3 S. Seshu and M.B. Reed.
Linear graphs and elektrical networks,
Addison-Wesley 61-5309.

4 A.K. Kevorkian and J. Snoek.
Theory and applications of structural analysis in disjointing and constructing hierarchical systems, Shell report.

5 D.K. Faddejew / W.H. Faddejewa.
Numerische methoden der linearen algebra,
VEB deutscher verlag der wissenschaften, Berlin 1964.

6 J. Lipson.
Symbolic methods for the computer solution of linear equations with applications to flowgraphs,
Proceedings of the 1968 summer institute on symbolic mathematical computation, Robert G. Tobey, editor.

ALTRAN Programs for SIGSAM Problem #6

(Continued from page 13)

IV. Remarks

In an attempt to understand the astonishing growth in the CPU time required for higher orders, we gathered some timing statistics from the routines internal to ALTRAN. Surprisingly, 75% of the CPU time is spent in the polynomial "divide-if-divisible" operation, while only 13% is spent in polynomial gcd computations. This suggests that substantial improvements in performance can be gained by using a faster algorithm for this operation (see [2]).

References

1. D. P. McCarthy, Problem #6 - The Kiang Problem, SIGSAM Bulletin No. 28, (December 1973), pg. 12.
2. S. C. Johnson, Sparse Polynomial Arithmetic, in preparation.