P R L

TECHNICAL NOTE

TN-84-37

"AUTOMATIC INTERPOLATOR"
A FORTRAN PROGRAM
FOR INTERPOLATION

By

K.J.Shah & V.R.Choksi

April 1984

PHYSICAL RESEARCH LABORATORY
AHMEDABAD 380009

C O N T E N T S

.....

" AUTOMATIC INTERPOLATOR "
A FORTRAN PROGRAM FOR INTERPOLATION

By

K.J. Shah & V.R. Choksi
Physical Research Laboratory
Ahmedabad-380009

ABSTRACT

This note represents Lagrangian Form of Interpolating Polynomial to the single and multi-dimensional cases to interpolate single and multivariate tables. In case of periodic functional array, a Trigonometrical Interpolation is performed by using Hermit's Formula. A general software package named "Automatic Interpolator" based on the above method has been developed in FORTRAN X and tested on DEC-1090 system. The Program can be utilised to interpolate single and multivariate arrays. A complete documentation of the Program and instruction set for execution are provided. The Program is available in compiled form on System Disk, at the Computer Center of PRL, Ahmedabad.

1. INTRODUCTION

In this article, we describe basis of Interpolation and an algorithm to interpolate single and multivariate arrays tabulated at equidistant or unequidistant arguments for finite interval.

A brief description of the basis of interpolation and its importance are presented in Section 1.1. Section 1.2 describes Numerical Algorithm and Section 1.3 presents characteristics of the Numerical Process.

1.1 Interpolation:

Interpolation can be defined as an art of reading between the lines of a table. In low level mathematics, the term usually denotes the process of computing intermediate value of a functional array. In higher mathematics, the term denotes the replacement of Function of complicated nature by simple one.

In practical problems of Physics, Interpolation is frequently used to deal with function whose analytical form is either totally unknown or else the function is of such a complicated nature that it cannot be easily subjected to arithmatical operations. Also, in order to reduce computational efforts over a certain interval, in some cases, it may be required to resort to interpolate, even though analytical representation of function is known.

There are several formulae for Numerical Interpolation, namely Forward, Backward and Central difference Formulae. Among these, most of the formulae are for equidistant argument. But Lagrange's Interpolation Formula can be used for equidistant as well as unequidistant arguments and Inverse Interpolation. Also, it is more convenient for computer programming relative to time consumption and memory requirement.

1.2 Numerical Algorithm:

In the theory of Interpolation, a complicated function is replaced by a polynomial or trigonometric series. This functional substitution is justified by the following stated theorems proved by Weierstrass in 1885.

Theorm 1:

Every continuous function in interval (a, b), can be represented by a Polynomial to assign degree of closeness i.e. for arbitrarily chosen $\epsilon > 0$, there exist a Polynomial $P(x)$ such that for every $x \in (a, b)$,

$$|F(x) - P(x)| < \epsilon$$

Theorem 2:

Every continuous function $F(x)$ of period 2 can be represented by a finite trigonometric series in the form

$$g(x) = a_0 + \sum_{i=1}^{n} \left[a_i . \sin ix + b_i . \cos ix \right],$$

such that $|F(x) - g(x)| < \epsilon$

Let one variable functional array be tabulated at (n + 1) pairs of points (x_i , F_i), i = 1, 2 ..... n+1. The approximate Lagrangian 's Interpolating Polynomial passing through above points is given by

$$P_n(x) = \sum_{i=1}^{n} \prod_{\substack{j=i \\ i \neq j}}^{n} \frac{x - x_j}{x_i - x_j} F(x_i) + R_n$$

$$\text{Where } R_n = \prod_{i=1}^{n} \frac{(x - x_i)}{n!} F^n(\xi)$$

$$(x_o < \xi < x_n)$$

is the remainder term...............(1)

If the given functional array is periodic the approximate Hermite's Trigonometric Interpolating Polynomial passing through above points is given by

$$P_n(x) = \sum_{i=1}^{n} \prod_{\substack{j=1 \\ i \neq j}}^{n} \frac{\mathrm{Sin}(Y - Y_j)}{\mathrm{Sin}\,(Y_i - Y_j)} F(x_i) + R_n$$

where R_n is as above ....(1a)

In case of multi-dimensional arrays, one can adopt above Lagragian's Form of Interpolating Polynomial. In this process, we hold one variable constant, write a series of Lagrangian Polynomials for interpolation at the given value of the other variable and then combine these values in a final Lagrangian form. Resulting expressions for arrays of two and three variables are respectively as follows:

$$P_n(x,Y) = \sum_{i=1}^{np} \prod_{\substack{j=1 \\ i \neq j}}^{np} \frac{(Y - Y_j)}{(Y_i - Y_j)} \cdot \sum_{k=1}^{np} \prod_{\substack{l=1 \\ k \neq l}}^{np} \frac{(x - x_l)}{(x_k - x_l)} \; F(x_k, Y_i)$$

$+ R_n$, where R_n is remainder ....(2)

$$P_n(x,Y,Z) = \sum_{i=1}^{np} \prod_{\substack{j=1 \\ i \neq j}}^{np} \frac{(Z - Z_j)}{(Z_i - Z_j)} \cdot \sum_{k=1}^{np} \prod_{\substack{l=1 \\ k \neq l}}^{np} \frac{(Y - Y_l)}{(Y_k - Y_l)} \cdot$$

$$\sum_{m=1}^{np} \prod_{\substack{n=1 \\ m \neq n}}^{np} \frac{(x - x_n)}{(x_m - x_n)} \cdot F(X_m, Y_k, Z_i) + R_n$$

Where R_n is remainder ....(3)

In case of Periodic arrays for two and three variables, Hermite's Trigonometrical Interpolating formulae are respectively as follows:

$$P_n(X,Y) = \sum_{i=1}^{np} \prod_{\substack{j=1 \\ i \neq j}}^{np} \frac{\sin(Y - Y_i)}{\sin(Y_i - Y_j)} \cdot \sum_{k=1}^{np} \prod_{\substack{l=1 \\ k \neq l}}^{np} \frac{\sin(X - X_l)}{\sin(X_m - X_l)} \cdot$$

$$F(X_k, Y_i) + R_n \qquad \ldots\ldots\ldots\ldots (2a)$$

$$P_n(X,Y,Z) = \sum_{i=1}^{np} \prod_{\substack{j=1 \\ i \neq 1}}^{np} \frac{\sin(Z - Z_j)}{\sin(Z_i - Z_j)} \cdot \sum_{k=1}^{np} \prod_{\substack{l=1 \\ k \neq l}}^{np} \frac{\sin(Y - Y_l)}{\sin(Y_k - Y_l)}$$

$$\sum_{m=1}^{np} \prod_{\substack{n=1 \\ m \neq n}}^{np} \frac{\sin(X - X_n)}{\sin(X_m - X_n)} \cdot F(X_m, Y_k, Z_i) + R_n$$

$$\ldots\ldots\ldots\ldots (3a)$$

The program evaluates the interpolated value by using Eqn. (1) or Eqn. (2) or Eqn. (3) according as the input array is one, two or three dimensional. In case of periodic array, it evaluates the interpolated values by virtue of Eqn. (1a), Eqn. (2a) or Eqn. (3a) according as the input array is one, two or three dimensional. The program selects Deg. + 1 consecutive array elements with center value for which interpolation is needed.

1.3 Characteristics of Numerical Process:

The program arranges the given functional arrays in ascending order. Then, according to the linear or periodic mode, supplied by the user, the program performs polynomial or trigonometrical interpolation respectively, with the given degree of freedom.

In case, the nature of functional array is many to one in the neighbourhood of interpolating point, it assigns the unique value as the interpolated value.

2. DOCUMENTATION OF THE PROGRAM

This section describes the contents of the program.

2.1 Definition of the Program:

SUBROUTINE MTITP (V, IC, JOSC, MT, N, X, Y, XIP, YIP, Z, Z3, ZIP, W, IDG)

2.2 Parameters of the Program:

V : (Real * 4, SCALAR, OUT-COMING) This variable represents the interpolated value for the given argument.

IC : (INTEGER * 4, SCALAR, IN-COMING, INTACT) It is decision making parameter

IC = 0 : Indicates that given functional array is in ascending or decending order.

IC = 1 : Indicates that given functional array is not in order.

JOSC : (INTEGER * 4, SCALAR, IN-GOING, INTACT)
It is decision making parameter.
IOSC = 0 indicates that Functional array is not periodic.
IOSC = 1 indicates that Functional array is periodic.

MT : (INTEGER * 4, SCALAR, IN-GOING, INTACT)
It is decision making parameter.
MT = 1 indicates that functional array is of one variable.
MT = 2 indicates that functional array is of two variable.
MT = 3 indicates that functional array is of three variable.

IDG : (INTEGER * 4, SCALAR, IN-GOING, INTACT)
This variable indicates the degree of freedom, which may attain value 2, 3, 4, 5, 6, 7 according to user's need.

N : (INTEGER * 4, SCALAR, IN-GOING, INTACT)
This variable represents the number of points of the functional array.

X : (REAL * 4, ARRAY, IN-GOING, DESTROYED)
This array represents the abscissa of the Functional array.

Y : (REAL * 4, ARRAY, IN-GOING, DESTROYED)
This array represents the ordinates of the functional array.

XIP : (REAL * 4, SCALAR, IN-GOING, INTACT)
This variable represents the value for which interpolation is to be performed.

YIP : (REAL * 4, SCALAR, IN-GOING, INTACT)
This variable represents the value for which interpolation is to be performed (to be used for two variable functional array)

Z : (REAL * 4, ARRAY, IN-GOING, DESTROYED)
This two dimensional array represents the functional value of two variable array.

Z3 : (REAL * 4, ARRAY, IN-GOING, DESTROYED)
This array represents the normal co-ordinates of the functional array.

ZIP : (REAL * 4, SCALAR, IN-GOING, INTACT)
This variable represents the value for which interpolation is to be performed. (to be used for three variable functional array).

W : (REAL * 4, ARRAY, IN-GOING, DESTROYED)
This three dimensional array represents the functional value of three variable array.

2.3 General Characteristics of the Program:

(a) Language : FORTRAN X

(b) Precision : Single, Double

(c) Name of the subprogram called : SUBROUTINE 'ASCND' which arranges given functional array in ascending order.

2.4. Special Characteristics of the Program:

(a) The program uses object-time dimension for arrays.

(b) User should supply the arguments in the CALL statement according to the explanation given in section 3.

(c) User should supply number of points of the functional array not less than six.

(d) In case of inverse interpolation abscissa and ordinate should be interchanged.

(e) In case of double precision requirement, user should call the program as 'DMTITP', and all real parameters should be defined in double precision.

2.5 Description of the output:

This program gives interpolated value as output for given input.

3. HOW TO EXECUTE THIS PROGRAM:

User can access this program by copying REELFILE from system disk and linking it with main program while execution.

e.g. Copy DSKC: (P, P_n) * . * =
DSKC: (342, 104) MTITP, REL.

```
Main Program
        ,
        ,
        ,

CALL MTITP (V, IC, JOSC, MT, N, X, Y, XIP, T1, T2,
            T3, T4, T5, IDG)  (For single variable)
   OR
CALL MTITP (V, IC, JOSC, MT, N, X, Y, XIP, YIP, Z,
            T1, T2, T3, IDG) (For double variables)
   OR
CALL MTITP (V, IC, JOSC, MT, N, X, Y, XIP, YIP, T1,
            Z3, ZIP, W, IDG) (For triple variables)
            ,
            ,
            ,
            ,
            END
```

NOTE : T1, T2, T3, T4, T5 should be supplied as dummy arguments.

4 Table

*Array is not shown in the table

Sr. No.	Function/ Array*	Analytical Value	Numerical values (Periodic)			Numerical values (Non-periodic)		
			Deg. 4	Deg. 5	Deg. 6	Deg. 4	Deg. 5	Deg. 6
1.	$f(x) = \cos x$							
	$x = 98^{\circ}$	-0.13917E00	-0.13777E00	-0.13917E00	-0.14169E00	-0.13915E00	-0.13917E00	-0.13917E00
	$x = 320^{\circ}$	0.76604E00	0.76604E00	0.76604E00	0.76604E00	0.76604E00	0.76604E00	0.76604E00
2.	$f(x) = e^{x}(x+5)$							
	$x = 1.30$	0.23117E02	0.23134E02	0.23110E02	0.23134E02	0.23134E02	0.23110E02	0.23134E02
	$x = 8.20$	0.48061E05	0.48093E05	0.48047E05	0.48059E05	0.48093E05	0.48047E05	0.48059E05
3.	$f(x,y) = e^{x}\sin y + y - 0.1$							
	$x = 1.60$, $y = 0.33$	0.25853E00	0.25854E00	0.25852E00	0.25853E00	0.25854E00	0.25852E00	0.25853E00
4.	*Array							
	$x = 1.60$, $y = 0.33$	-	0.18405E01	0.18353E01	0.18353E01	0.18353E01	0.18352E01	0.18352E01
5.	$f(x,y,z) = x \cdot e^{y+z}$							
	$x = 1.30$							
	$y = 0.35$, $z = 0.46$	0.29223E01	0.29223E01	0.29223E01	0.29223E01	0.29732E01	0.29732E01	0.29732E01