# NASA Technical Memorandum 80070

(NASA-TM-80070) INTERP3: A COMPUTER ROUTINE FOR LINEAR INTERPOLATION OF TRIVARIATE FUNCTIONS DEFINED BY NONDISTINCT UNEQUALLY SPACED VARIABLES (NASA) 15 p HC A02/MF A01 CSCL 09B G3/61

N79-21800

Unclas 23952

# INTERP3 - A COMPUTER ROUTINE FOR LINEAR INTERPOLATION OF TRIVARIATE FUNCTIONS DEFINED BY NONDISTINCT UNEQUALLY SPACED VARIABLES

DENISE C. HILL AND SHELBY J. MORRIS, JR.

MARCH 1979

NASA
National Aeronautics and Space Administration

**Langley Research Center**
Hampton, Virginia 23665

# INTERP3 - A COMPUTER ROUTINE FOR LINEAR INTERPOLATION OF TRIVARIATE FUNCTIONS DEFINED BY NONDISTINCT UNEQUALLY SPACED VARIABLES

Denise C. Hill and Shelby J. Morris, Jr.

Langley Research Center
Hampton, Virginia 23665

## SUMMARY

INTERP3 is a computer routine designed to linearly interpolate a variable which is a function of three independent parameters. It is unique in that the variables within the parameter arrays do not have to be distinct, or equally spaced, and the array variables can be in increasing or decreasing order.

## INTRODUCTION

The type of trivariate functions which can be evaluated by existing interpolation routines is severely limited by restrictions imposed by the routines. In the Langley computer complex, these routines include ITRI, which requires the parameter variables to be distinct and in algebraically increasing order, and FUNGEN which assumes that the function is known for equal intervals of the parameter variables.

The desire to use the computer for linear interpolation in a data table of propeller slipstream characteristics generated the need for a computer routine to interpolate a trivariate function defined by nondistinct, unequally spaced parameter variables. INTERP3 is designed to fulfill that need. Furthermore, this routine allows the parameter variables to be in increasing or decreasing order.

### Description of Program

This program performs linear interpolation for a set of functions defined over the same set of independent parameter points. By the use of multiple entry statements, INTERP3 evaluates the function at the point (XIND, YIND, ZIND), producing a value WD, where:

XIND = The input x parameter where interpolation is desired

YIND = The input y parameter where interpolation is desired

ZIND = The input z parameter where interpolation is desired

WD = The output interpolated value of the function at the point (XIND, YIND, ZIND).

This program does not allow extrapolation. If any input parameter exceeds the data table range, a warning is printed and WD will not be calculated, but given the value of zero.

The names and descriptions of the subroutines are as follows:

INTERP3 - Dummy main program to initiate calculations. Control returns to this routine after evaluation of each input point.

XSEAR - Main routine. Loads data table and performs interpolation in X parameter array.

YZSEAR - Performs interpolation in Y and Z parameter arrays.

WSEAR - Performs interpolation in W array.

A FORTRAN listing of this program is presented in Appendix A, and a flow chart is presented in figure 1.

## Description of Data File

The data file contains all tabular values in the X, Y, Z, and W arrays. The purpose of FORMAT 601 is to read these parameter values in the sequence x, y, z, w for each record. To ensure that all parameter values are read into the program, a record must be added to the end of the data table file with dummy values that do not equal the respective x, y, z, w values of the last true record. This edited file must be placed at TAPE5, prior to execution of the program.

A sample case including data file and input/output listings is presented in Appendix B.

## CONCLUSION

INTERP3 is a program designed to linearly interpolate a variable which is a function of three independent parameters. It is unique in that values within the parameter arrays do not have to be distinct, or equally spaced and array values can be either algebraically increasing or decreasing. These features

of INTERP3 give it a capacity of interpolating many different kinds of trivariate functions.

## APPENDIX A
## LISTING OF PROGRAM INTERP3

```
      PROGRAM INTERP3(INPUT,OUTPUT,TAPE5,TAPE6=OUTPUT)
C  DRIVER PROGRAM
C
200    FORMAT(1H1)
       WRITE(6,200)
C     THE FOLLOWING RECORD MUST BE READ INTO THE PROGRAM  BEFORE THE FIRST INPUT VALUES
       IREAD=1
C
C     THE INPUT VALUES XIND,YIND, AND ZIND ARE ENTERED HERE
       XIND=.55
       YIND=3.0
       ZIND=1.47
       CALL XSEAR(XIND,YIND,ZIND,IREAD)
       XIND=.50
       YIND=.2
       ZIND=.3
       CALL XSEAR(XIND,YIND,ZIND,IREAD)
       XIND=.55
       YIND=3.3
       ZIND=2.00
       CALL XSEAR(XIND,YIND,ZIND,IREAD)
       XIND=.7
       YIND=2.8
       ZIND=.25
       CALL XSEAR(XIND,YIND,ZIND,IREAD)
       XIND=.6
       YIND=3.2
       ZIND=1.5
       CALL XSEAR(XIND,YIND,ZIND,IREAD)
       XIND=.65
       YIND=3.8
       ZIND=1.0
       CALL XSEAR(XIND,YIND,ZIND,IREAD)
       XIND=.65
       YIND=3.6
       ZIND=2.0888
       CALL XSEAR(XIND,YIND,ZIND,IREAD)
       END
```

```
      SUBROUTINE XSEAR(XIND,YIND,ZIND,IREAD)
C  THIS SUBROUTINE LOADS YOUR DATA TABLE ON THE FIRST BRANCH
C  AND SEARCHES THROUGH THE X ARRAY
      COMMON/DATIN/ TZ(20,20,20),TW(20,20,20),TY(20,20,20),TX(20),
     1MIZ(20,20),MIY(20),MIX
      DIMENSION  YDL(20),YDH(20)
      IF(IREAD.EQ.0)GO TO 204
      IREAD=0
C  THE PROGRAM USES LINEAR INTERPOLATION
C  THE PROGRAM PRODUCES A VALUE OF WO=F(ZIND,YIND,XIND) WHERE
C        XIND= INPUT X PARAMETER WHERE INTERPOLATION IS DESIRED
C        YIND= INPUT Y PARAMETER WHERE INTERPOLATION IS DESIRED
C        ZIND= INPUT Z PARAMETER WHERE INTERPOLATION IS DESIRED
C THE INPUT TABULAR VALUES FOR THE INTERPOLATION ARE AS PER THE 601 FORMAT
C  THE FOLLOWING 601 FORMAT MUST BE ALTERED TO READ THE TABULAR VALUES..
C  X, Y, Z, W, --AS LOCATED ON YOUR DATA FILE
601   FORMAT(10X,4F10.4)
      ISKIP=0
      READ(5,601)X,Y,Z,W
700   FORMAT(10X,4F10.4)
C  THE NEXT THREE LOOPS LOAD YOUR TABULAR VALUES
C  LOOP FOR X PARAMETER ARRAY
      DO 100 ID=1,20
      XO=X
C
C  LOOP FOR Y PARAMETER ARRAY
      DO 101 ID1=1,20
      YO=Y
C
C  LOOP FOR Z PARAMETER ARRAY AND W VALUES
      DO 102 ID2=1,20
      IF(ID2.EQ.1)GO TO 5
      IF(ISKIP.EQ.1)GO TO 6
      READ(5,601)X,Y,Z,W
      WRITE(6,700)X,Y,Z,W
      IF(EOF(5))203,6
6     ISKIP=0
      IF(Y.EQ.YO)GO TO 5
      MIZ(ID1,ID)=ID2-1
      ISKIP=1
      GO TO 201
5     MIY(ID)=ID1-1
```

```
      IF(X.EQ.XO)GO TO 4
      ISKIP=1
      GO TO 202
4     CONTINUE
      MIX=ID-1
      TX(ID)=X
      TY(ID1,ID) =Y
      TZ(ID2,ID1,ID)=Z
      TW(ID2,ID1,ID)=W
102   CONTINUE
201   CONTINUE
101   CONTINUE
202   CONTINUE
100   CONTINUE
203   CONTINUE
      WRITE(6,300)
204    CONTINUE
800   FORMAT(9X,*END OF FILE*,/,1H1)
C     CHECK OF MONOTONIC DIRECTION
      IF(TX(1).GT.TX(2))GO TO 18
C   SEARCH THRU X TABULAR VALUES
      IF(XIND.GT.TX(MIX))GO TO 15
      IF(XIND.LT.TX(1))GO TO 15
      GO TO 16
15    WRITE(6,300)
300   FORMAT(2X,*  INPUT VALUE OF XIND OUT OF X TABLE RANGE*)
      WD=0.0
      WRITE(6,200)XIND,YIND,ZIND,WD
      RETURN
16    DO 30 IDUM=1,MIX
      IF(XIND.LE.TX(IDUM))GO TO 17
30    CONTINUE
17    IXH=IDUM
      IXL=IXH-1
      IF(XIND.EQ.TX(IDUM))IXL=IDUM
      GO TO 60
18    IF(XIND.LT.TX(MIX))GO TO 15
      IF(XIND.GT.TX(1))GO TO 15
      DO 40 IDUM=1,MIX
      IF(XIND.GE.TX(IDUM))GO TO 17
40    CONTINUE
60    CONTINUE
```

```
      NYDL=MIY(IXL)
       DO 70 ID=1,NYDL
      YDL(ID)=TY(ID,IXL)
70    CONTINUE
      CALL YZSEAR(NYDL,IXL,YDL,YIND,ZIND,WL)
      IF(IXH.EQ.IXL)WD=WL
      IF(IXH.EQ.IXL)WRITE(6,200)XIND,YIND,ZIND,WD
      IF(IXH.EQ.IXL)RETURN
      NYDH=MIY(IXH)
       DO 80 ID1=1,NYDH
      YDH(ID1)=TY(ID1,IXH)
80    CONTINUE
      CALL YZSEAR(NYDH,IXH,YDH,YIND,ZIND,WH)
      TXH=TX(IXH)
      TXL=TX(IXL)
      WD=WL+(WH-WL)*(XIND-TXL)/(TXH-TXL)
200   FORMAT(2X,*XIND=*,F6.4,3X,*YIND=*,F8.4,3X,*ZIND=*,F8.4,3X,*WD=*
     **,F6.4,//)
      WRITE(6,200)XIND,YIND,ZIND,WD
       RETURN
      END
```

```
      SUBROUTINE YZSEAR(NY,NX,TYI,YIND,ZIND,WD)
C  THIS SUBROUTINE SEARCHES THE Y AND Z ARRAYS
      DIMENSION TYI(NY),ZII(20),WII(20),ZIII(20),WIII(20)
      COMMON/DATIN/ ZI(20,20,20),WI(20,20,20),YY(20,20,20),XX(20),
     1MIZ(20,20),MIY(20),MIX
1     FORMAT(2X,*  INPUT VALUE OF YIND OUT OF Y TABLE RANGE*)
2     FORMAT(2X,*  INPUT VALUE OF ZIND OUT OF Z TABLE RANGE*)
C     CHECK OF MONOTONIC DIRECTION OF Y ARRAY
      IF(TYI(1).GT.TYI(2))GO TO 7
      IF(YIND.LT.TYI(1))GO TO 3
      IF(YIND.GT.TYI(NY))GO TO 3
      GO TO 10
3     WD=0.0
      WRITE(6,1)
      RETURN
10    DO 20 I=1,NY
      IF(YIND-TYI(I))4,5,20
20    CONTINUE
4     NYH=I
      NYL=I-1
      GO TO 30
5     NYH=I
      NYL=NYH
30    CONTINUE
      NZD=MIZ(NYL,NX)
      DO 40 ID=1,NZD
      ZII(ID) = ZI(ID,NYL,NX)
      WII(ID)=WI(ID,NYL,NX)
40    CONTINUE
      GO TO 50
6     WD=0.0
      WRITE(6,2)
      RETURN
7     IF(YIND.GT.TYI(1))GO TO 3
       IF(YIND.LT.TYI(NY))GO TO 3
      DO 80 I=1,NY
      IF(YIND-TYI(I))80,5,4
80    CONTINUE
50    CONTINUE
C     CHECK OF MONOTONIC DIRECTION OF Z ARRAY
      IF(ZII(1).GT.ZII(2))GO TO 90
       IF(ZIND.GT.ZII(NZD))GO TO 6
```

```
      IF(ZIND.LT.ZII(1))GO TO 6
      GO TO 100
90     IF(ZIND.LT.ZII(NZD))GO TO 6
      IF(ZIND.GT.ZII(1))GO TO 6
100   CONTINUE
      CALL WSEAR(NZD,ZII,WII,WL,ZIND)
      IF(NYH.EQ.NYL)WD=WL
      IF(NYH.EQ.NYL)GO TO 70
      NZDD=MIZ(NYH,NX)
      DO 60 ID1=1,NZDD
      ZIII(ID1)=ZI(ID1,NYH,NX)
      WIII(ID1)=WI(ID1,NYH,NX)
60    CONTINUE
C     SECOND CHECK OF Z ARRAY DIRECTION
      IF(ZIII(1).GT.ZIII(2))GO TO 110

      IF(ZIND.LT.ZIII(1))GO TO 6
      IF(ZIND.GT.ZIII(NZDD))GO TO 6
      GO TO 120
110   IF(ZIND.GT.ZIII(1))GO TO 6
      IF(ZIND.LT.ZIII(NZDD))GO TO 6
120   CONTINUE
      CALL WSEAR(NZD,ZII,WII,WH,ZIND)
       TYH=TYI(NYH)
       TYL=TYI(NYL)
      WD=WL+(WH-WL)*(YIND-TYL)/(TYH-TYL)
70    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE WSEAR(NZ,ZI,WI,WD,ZIND)
C  THIS SUBROUTINE PERFORMS INTERPOLATION IN THE W ARRAY
      DIMENSION  ZI(20), WI(20)
      IF(ZI(1).GT.ZI(2))GO TO 30
      DO 10 ID=1,NZ
      IF(ZIND-ZI(ID))3,2,10
2     WD=WI(ID)
      GO TO 20
3     WL=WI(ID-1)
      WH=WI(ID)
      ZL=ZI(ID-1)
      ZH=ZI(ID)
      WD=WL+(WH-WL)*(ZIND-ZL)/(ZH-ZL)
      GO TO 20
10    CONTINUE
20    CONTINUE
      RETURN
30    DO 40 ID=1,NZ
      IF(ZIND-ZI(ID))40,2,3
40    CONTINUE
      END
```

## APPENDIX B - SAMPLE CASE

(a) Data File Listing

The following data file is listed as it would appear at TAPE5. FORMAT 601 reads the values x,y,z,w, respectively. Note the arrangement of the values in each paramter array, and the last file record, which contains dummy values.

| x | y | z | w |
|---|---|---|---|
| .5000 | 0.0000 | .2070 | .3280 |
| .5000 | 0.0000 | .3420 | .4361 |
| .5000 | 0.0000 | .5037 | .5142 |
| .5000 | 0.0000 | .6848 | .5524 |
| .5000 | .2000 | .2000 | .2905 |
| .5000 | .2000 | .4000 | .4580 |
| .5000 | .2000 | .6000 | .5660 |
| .5000 | .2000 | .8000 | .6140 |
| .5000 | .2000 | 1.0000 | .6180 |
| .5500 | 3.0000 | .4883 | .1345 |
| .5500 | 3.0000 | .6777 | .1940 |
| .5500 | 3.0000 | 1.0675 | .3056 |
| .5500 | 3.0000 | 1.4706 | .4082 |
| .5500 | 3.0000 | 1.8861 | .5030 |
| .5500 | 3.0000 | 2.3118 | .5899 |
| .5500 | 3.2000 | .5994 | .1564 |
| .5500 | 3.2000 | 1.0255 | .2756 |
| .5500 | 3.2000 | 1.4662 | .3848 |
| .5500 | 3.2000 | 1.9204 | .4862 |
| .5500 | 3.2000 | 2.3848 | .5766 |
| .5500 | 3.4000 | .3008 | .0555 |
| .5500 | 3.4000 | .7581 | .1877 |
| .5500 | 3.4000 | 1.2318 | .3087 |
| .5500 | 3.4000 | 1.7204 | .4201 |
| .5500 | 3.4000 | 2.2217 | .5217 |
| .6500 | 3.2000 | .6146 | .1607 |
| .6500 | 3.2000 | .8318 | .2227 |
| .6500 | 3.2000 | 1.0530 | .2825 |
| .6500 | 3.2000 | 1.2782 | .3395 |
| .6500 | 3.2000 | 1.5070 | .3941 |
| .6500 | 3.2000 | 1.7397 | .4469 |
| .6500 | 3.2000 | 1.9751 | .4970 |
| .6500 | 3.2000 | 2.2132 | .5441 |
| .6500 | 3.8000 | .4215 | .0719 |
| .6500 | 3.8000 | .9895 | .2160 |
| .6500 | 3.8000 | 1.5758 | .3471 |
| .6500 | 3.8000 | 2.1778 | .4655 |
| .6500 | 3.8000 | 2.7915 | .5724 |
| 0.0000 | 0.0000 | 0.0000 | 0.0000 |

(b) Input/Output Listing

Compare the following input/output values to those presented in the sample data file. Error messages will be printed whenever extrapolation is attempted.

```
XIND= .5500   YIND= 3.0000   ZIND= 1.4700   WD= .4080

XIND= .5000   YIND=  .2000   ZIND=  .3000   WD= .3743

XIND= .5500   YIND= 3.3000   ZIND= 2.0000   WD= .5017

  INPUT VALUE OF XIND OUT OF X TABLE RANGE
XIND= .7000   YIND= 2.8000   ZIND=  .2500   WD=0.0000

XIND= .6000   YIND= 3.2000   ZIND= 1.5000   WD= .3924

XIND= .6500   YIND= 3.8000   ZIND= 1.0000   WD= .2183

XIND= .6500   YIND= 3.6000   ZIND= 2.0888   WD= .5195
```

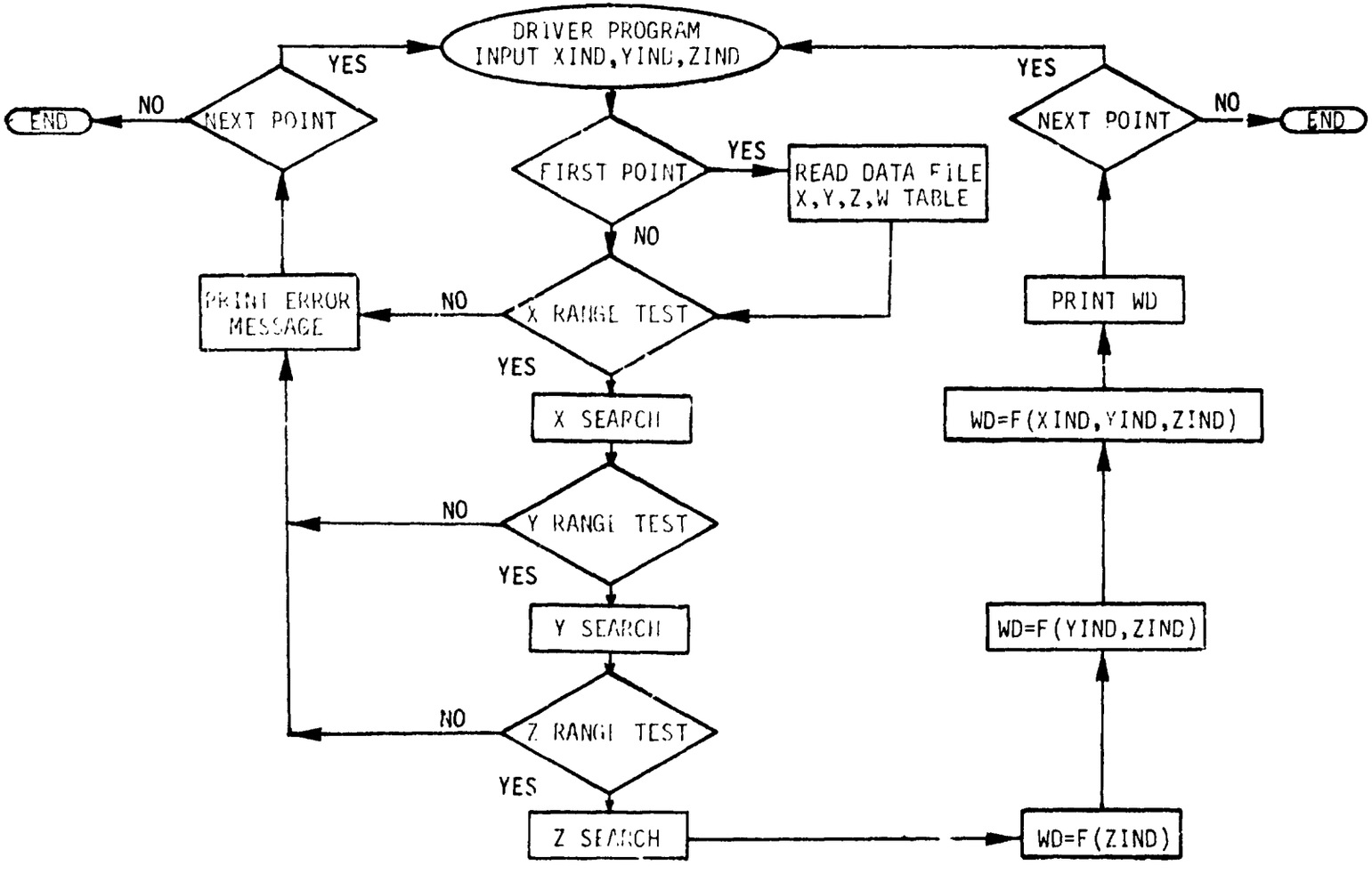


Figure 1.- Flow Chart of INTERP3