Developments In Business Simulation & Experiential Exercises, Volume 20,1993

Reducing the Complexity of Interactive Variable Modeling in Business Simulations Through Interpolation

Kenneth Goosen, University of Arkansas at Little Rock

ABSTRACT

This paper, which is prepared in response to a question raised by Steven Gold concerning use of the interpolation approach in modeling, describes how the interpolation approach achieves interaction among variables in a simulation. The theory underlying use of the approach is presented and a five-step algorithm is suggested for implementation. Application of the approach to non-multiplicative models is also addressed. The paper concludes that based on the principle of creating interpolation arrays in terms of percentages, and given a starting base quantity, the interpolation methodology can effectively emulate the interaction of any type of equation that contains multiple interacting variables.

INTRODUCTION

How a linear interpolation algorithm can be used to achieve results normally associated with complex curvilinear equations was presented by Goosen (1986). In this paper, only a brief mention was made of how the interpolation model could be used to achieve variable interaction. A valid question can be asked concerning whether signification interaction among variables can be achieved by using the interpolation method introduced in Goosen's paper titled, An Interpolation Approach to Developing Mathematical Functions for Business Simulations." Because the primary concern in that paper was to introduce the methodology of interpolation, the discussion and illustration of variable interaction was limited to a dependent variable and a change in a single independent variable.

The purpose of this paper is to describe how the interpolation methodology can be extended to accomplish interactive effects among any number of variables. Interaction of variables in models using multiplicative and non-multiplicative equations are separately described.

TWO CHARACTERISTICS OF MULTIPLICATIVE EQUATIONS

A multiplicative model is an equation where each independent variable is treated as a multiplier of the other variables. The equation, V = A x B x C, is an example of the basic form of a multiplicative equation. The inclusion of exponents which also contain the independent variables (e.g., $Y = A^{K1A} \times B^{k2B}$) does not change the fundamental characteristics of the multiplicative model.

A detailed analysis of a simple multiplicative equation, such as Y = A x B, is sufficient to reveal several unique characteristics of multiplicative equations that allow our interpolation methodology to emulate interactive variable behavior. Two basic characteristics of multiplicative equations underlie the theoretical foundation of this approach.

The first characteristic concerns the effect of a change in an independent variable on the percentage change of the dependent variable. Successive changes in values for B at a given value of A will result in percentage changes in Y which will be equal to percentage changes in Y for the same changes in B at other assigned values of A. That is, the measured changes in Y resulting from a change in B are a constant percentage for all values of A. Percentage changes in V due to changes in B are basically independent of values for A.

An interactive graphical model is presented in Figure 1. Note that for each change in the value of A, a shift in the curve results.

Graph A shows values of V for different values of B when A = a_1. Graph B shows the shift in the function, A x B, when A = a_2. Graph C show an additional shift in the function when A = a_3.

In Graph A. when A = a_1 and the initial quantity is y_1 at b_1:
A change in B to b_2 results in a percentage of y_2/y_1 for the change in V to y_2. A change in B to b_3 results in a percentage of y_3/y_1 for the change in V to y_3.
Note: $V_2/Y_1 \neq V_3/V_1$

In Graph B, when A = a_2 and the initial quantity is y_4 at b,:
A change in B to b_2 results in a percentage of y_5/y_4 for the change in V to y_5. A change in B to b_3 results in a percentage of y_6/Y_4 for the change in V to y_6.
Note: $y_5/y_4 \neq y_6/y_4$

In Graph C, when A = a_3 and the initial quantity is y_7 at b_1:
A change in B to b_2 results in a percentage of y_8/y_7 for the change in Y to y_8. A change in B to b_3 results in a percentage of y_9/y_7 for the change in Y to y_9.
Note: $y_8/y_7 \neq y_9/y_7$

At each value for A the increases in Y can be summarized as follows:

When A = a_1 and for changes in B (b_1, b_2, b_3) the percentage changes for Y are:

$$[\ y_1/y_1,\ y_2/y_1,\ y_3/y_1\ ]$$

When A = a_2 and for changes in B (b_1, b_2, b_3) the percentage changes for Y are:

$$[\ y_4/y_4,\ y_5/y_4,\ y_6/y_4\ ]$$

When A = a_3 and for changes in B (b_1, b_2, b_3) the percentage changes for Y are:

$$[\ y_7/y_7,\ y_8/y_7,\ y_9/y_7\ ]$$

Therefore, at a value of b_1 for B when A = a_1, a_2, a_3:

$$y_1/y_1 = y_4/y_4 = y_7/y_7$$

and when B changes from b_1 to b_2 and for values of a_1, a_2, a_3 for A:

$$y_2/y1 = y_5/y_4 = y_8/y_7$$

and when B changes from b_2 to b_3 and for values of a_1, a_2, a_3 for A:

$$y_3/y_1 = y_6/y_4 = y_9/y_7$$

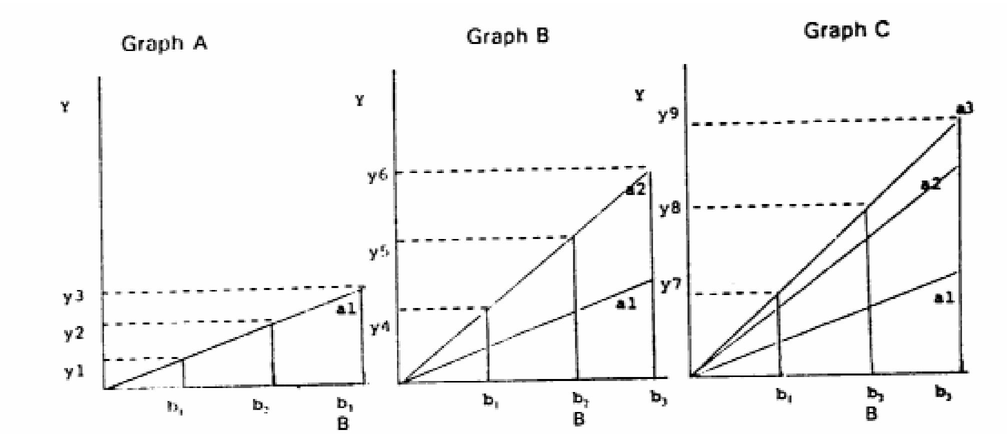


Figure 1 Graphical Illustration of Percentage Changes in the Dependent Variable

The significance of these equalities is that a change in B. regardless of the value assigned to A, results in the same percentage change in V. n other words, for each shift in the curve due to a change in A, changes in B will have no effect on the percentage change in Y.

The relationship of changes in A relative to B and changes in B relative to A means that interpolation schedules in terms of percentages rather than absolute values can be prepared. The following example shows how a schedule of changes in V values have been converted to a schedule of percentage changes. The quantity schedule is based on the equation Y = A x B

Quantity Schedule

(Y = A x B)

A	b_1 4	b_2 6	b_3 8	b_4 10
10	40	60	80	100
20	80	120	160	200
30	120	180	240	300
40	160	240	320	400

Schedule of percentage Changes in Y Resulting from Changes in B

A		b_1 4	b_2 6	b_3 8	b_4 10
a_1	10	1.	1.5	2.0	2.5
a_2	20	1.	1.5	2.0	2.5
a_3	30	1.	1.5	2.0	2.5
a_4	40	1.	1.5	2.0	2.5

The percentages are computed by using the quantities in the bi column as the initial quantities. For example when A = 1 0 the percentage changes in Y resulting from changes in B are 40/40 (1), 60/40 (1.5), 80/40 (2), and 100/40 (2.5). Note that the percentages associated with the B values are the same for each value of A. Consequently, as a practical matter, interpolation in multiplicative equations can be accomplished by using only a single row of percentages.

A second unique characteristic concerns multiplicative equations that have maximum or minimum values at certain values for B. The value of B that determines the minimum or maximum is the same regardless of the value for A. Figure 2 shows a graph created by using the Gold/Pray demand model based on the original parameters presented in their 1 984 paper.

Note that in Figure 2 whether the price, is $10, $20, or $30, the amount of advertising that maximizes quantity is $200,000. Regardless of the price, the optimal value for advertising is the same. The reason again has to do with percentage relationships. In Figure 2, each change in price produces a constant percentage shift in the demand schedule. A proportional shift in the advertising/quantity schedule occurs for each change in the assigned value for price.

The significance of this characteristic is that if the intent in using interpolation is to emulate a multiplicative type model where the model has maximum or minimum values, then care must be taken to see that each sketched array of A values reaches its maximum at the same value of B.

ILLUSTRATION OF INTERACTIVE MODELING THROUGH INTERPOLATION

From the above graphical and numerical example it is apparent that for multiplicative models the interaction of variables can be expressed in terms of percentages. This fact allows a simple

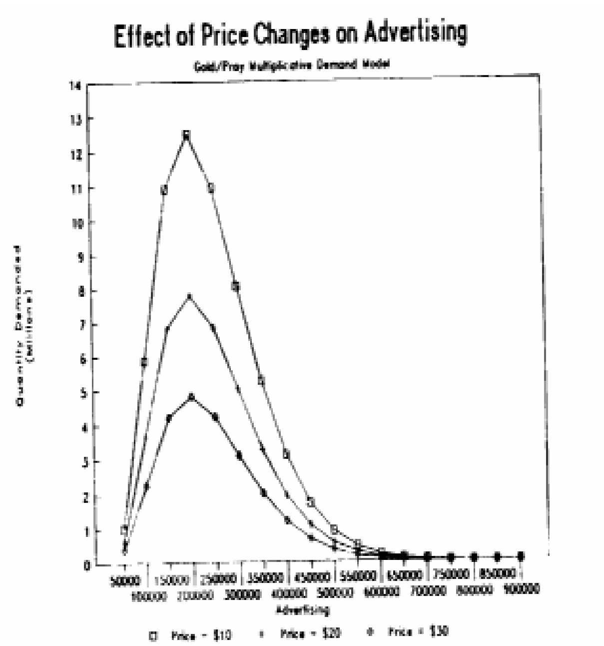


Figure 2 Effect of Changes in An Interacting Variable on a Multiplicative Model

equation to be developed to achieve the same interactive variable effects inherent in multiplicative models:

$$Y = BQ \times P_a \times P_b \times P_c \ldots$$

BQ - Base quantity
P_a - A percentage change in quantity that results from a change in A.
P_b - A percentage change in quantity that results from a change in B.
P_c - A percentage change in quantity that results from a change in C.

Specific values for P_a P_b P_c, are determined by interpolation from percentage change arrays.

In order to achieve interaction among variables, the interpolation algorithm model we presented in our original paper must be modified to allow the inclusion of percentage arrays. The modified algorithm may be stated as follows:

Step 1 - For each independent variable sketch on graph paper the desired function in terms of percentages of change. The sketched curve may be linear or curvilinear. Note: In order to emulate a ` multiplicative function, only one percentage line needs to be drawn; however; the emulation of interactive nonmultiplicative functions through interpolation requires that more than 1 percentage function line be sketched.

Step 2 -For each graph, identify points on the function at selected interval increments of the specified independent variable on the X-axis. Determine from the graph the corresponding percentage change.

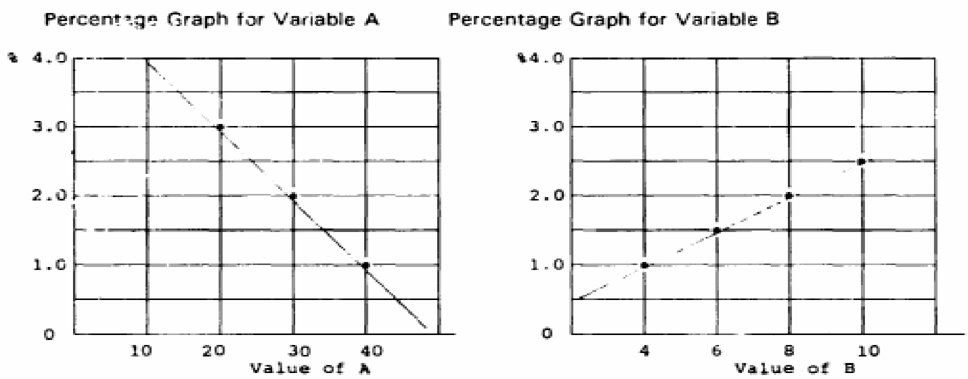
Step 3 - Prepare schedules listing the values assigned to each independent variable and the corresponding percentage values.

Step 4 - Develop an interpolation equation that will provide percentage values for all selected values of the independent variables.

Step 5 - Compute the value of the dependent variable using the equation $V = BQ \times P_a x P_b \times P_c$..

Figure 3 Interpolation Algorithm to Achieve Variable Interaction

Step 1 Draw percentage functions



Step 2 Select points on the sketched percentage functions as illustrated.

Step 3 Prepare percentage schedules for the independent variables, A and B.

A	%
10	4.0
20	3.0
30	2.0
40	1.0

B	%
4	1.0
6	1.5
8	2.0
10	2.5

Step 4 Use the equation presented in the 1986 paper

$$IV = Y_i + \left[\frac{DV - X_i}{X_{i+1} - X_i} * (Y_{i+1} - Y_i)\right]$$

To achieve the effect of interaction, this equation must be used twice in this example. The schedules of percentage changes for both A and B require interpolation. Assume that the values assigned to A and B were respectively 25 and 7.

Step 5 By use of interpolation the appropriate percentages for A and B respectively would be 250% and 175%. Assuming a base quantity of 40, the value of the dependent variable would be:

$$Y = 40 \times 2.5 \times 1.75 = 175$$

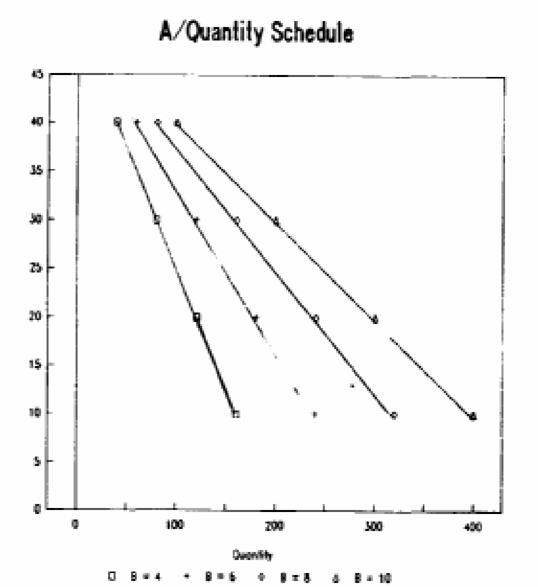
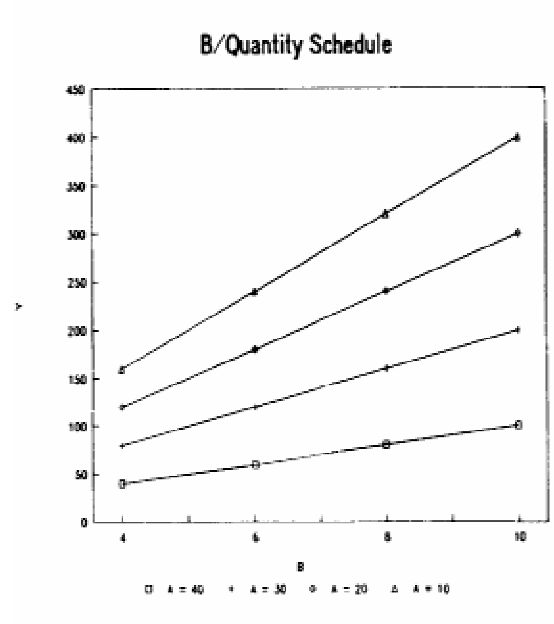


Figure 4 Graphical Illustration of Variable Interaction from Data Created by Interpolation



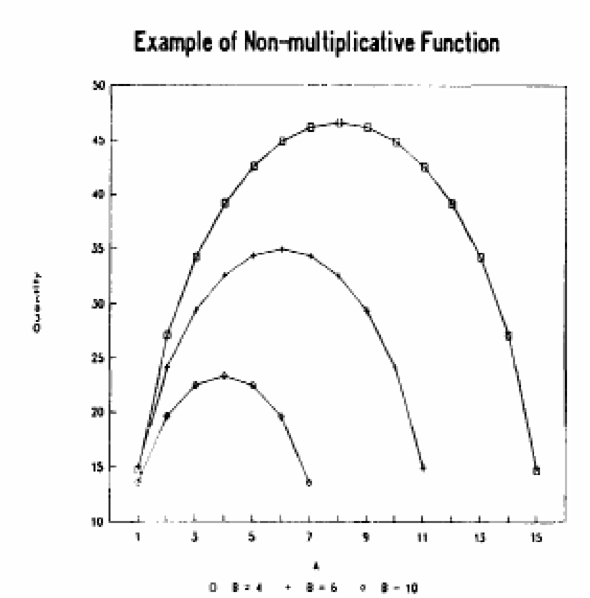


Figure 5 Example of Interactive Behavior in a Non-Multiplicative Function

This five-step algorithm is illustrated in Figure 3. The appendix to this paper presents an example of an interpolation computer program makes the required interpolation calculations.

The graphs in Figure 4 are prepared from values generated by our interactive interpolation algorithm in Figure 3.

INTERACTIVE MODELING OF NON-MULTIPLICATIVE EQUATIONS

Regarding all other type of models which here are collectively described as non-multiplicative, the problem of achieving interaction among variables is somewhat more difficult to understand. However, implementation of the interpolation procedure is only slightly more difficult.

An example of a non-multiplicative model is the following where A and B are considered to be variables and C, D. and E are constants:

$$Q = \sqrt{2CAB - D(A^2) - E(B^2)}$$

Figure 5 shows the behavior of this function for three different values of B.

Note that in Figure 5, the maximum quantity of each curve is at a different value for B. Also, a change in B (e.g., from b1 to b_2 and from b_1 to b_3) at different values of A will not result in proportionate changes in quantity. For each value of B there is a different schedule of percentage changes. To achieve through interpolation the same type of non-multiplicative equation interaction among variables. a schedule of percentage changes such as the following must be prepared, assuming A is the primary variable:

BQ = 12.08

Schedule of Percentage Changes for B

	A	% change (B = 3)
a_1	1	1.00
a_2	2	1.35
a_3	3	1.44
a_4	4	1.35
a_5	5	1.00

	A	b_1 3	b_2 4	b_3 5	b_4 6
a_1	1	1.00	1.12	1.24	1.24
a_2	2	1.00	1.20	1.35	1.48
a_3	3	1:00	1.28	1.50	1.68
a_4	4	1.00	1.37	1.75	2.00
a_5	5	1.00	1.86	2.41	2.85

These schedules are derived from the equation, $Q = \sqrt{2CAB - D(A^2) - E(B^2)}$, where D =40, E = 6 and C = 40. Notice that for each value of A, the array of percentages is different. Also, note that for the primary variable, A, the percentage schedule of changes requires only one column.

In order to determine the effect of changes in B, the interpolation algorithm must identify the value of A first and then interpolate the appropriate array of percentages for the given value of A. The value of A in non-multiplicative equations is important and must be explicitly recognized in the process of interpolation. For example, a value of 1.5 for A requires that an array of percentages at that value be determined by interpolation.

The author has developed an effective computer program (see appendix) for this type of interpolation. This program, which is relatively small, allows emulation of non-multiplicative equations to be easily accomplished. Given this computerized interpolation algorithm, the only requirement is that a family of curves be sketched and converted either to percentages or quantity schedules. The 5-step method may be used to create the appropriate change schedules required for interpolation. The complexity issue raised by Goad is greatly diminished once this computerized interpolation algorithm is employed.

SUMMARY

Based on the principle of creating interpolation arrays in terms of percentages and given a starting base quantity, the interpolation methodology developed can effectively emulate the interaction of any type of equation that contains multiple interacting variables. What is required is the use of the 5-step interpolation procedure to create a percentage change schedule for each variable. The advantage of using interpolation to achieve variable interaction is that the simulation designer can create any type of function that will give the desired results at all levels of activity or decision levels.

References

Gold, S & Pray T.F., (1984) Simulating Market- and Firm-level Demand Functions in Computerized Business Simulations, Simulations and Games. 1 5, 346-363.

Goosen. Kenneth R., "An Interpolation Approach to Developing Mathematical Functions for Business Simulations", Developments in Business Simulation and Experiential Exercises, 13, 248-258

Appendix: Example of an Interpolation Computer Program

The implementation of the proposed interpolation algorithm in a simulation requires writing a program that will automatically make all of the required interpolation calculations. This appendix presents such a program in Basic. The DATA statements in lines 10000-10230 contain the same information presented in step three of Figure 3. In this program variable A is presented as price and variable B is presented as advertising. This program will interpolate both multiplicative and nonmultiplicative functions. Note that the statements in lines 10120-10230 have the same data values. This means that the type of function being emulated through interpolation is multiplicative in nature.

```
50 CLS
60 DEFDBL A-H, O-Z
70 DIM P(20), Q(20), X(20), Y(20), ADV1(20), ADV2(20),
ADV3(20),  ADV4(20),  ADV5(20),  ADV6(20),  ADV7(20),
ADV8(20), ADV9(20), ADV10(20), ADV11(20)
80 DIM TVA1(20), TVA2(20), TVAIV(20), ADX(20), QU(20),
QP(20), ADV(20), QUP(20), QUA(20), PXP(20), PXA(20),
ADXP(20), ADXA(20), PX(20), AP(20)
82 NU = 11: BQ = 40

100 INPUT "Price decision           "; PD
110 INPUT "Advertising decision     "; AD

130 FOR I = 1 TO NU: READ P(I): NEXT I
140 FOR I = 1 TO NU: READ QP(I): NEXT I
145 FOR I = 1 TO NU: READ ADV(I): NEXT I
150 FOR I = 1 TO NU: READ ADV1(I): NEXT I
160 FOR I = 1 TO NU: READ ADV2(I): NEXT I
170 FOR I = 1 TO NU: READ ADV3(I): NEXT I
180 FOR I = 1 TO NU: READ ADV4(I): NEXT I
190 FOR I = 1 TO NU: READ ADV5(I): NEXT I
200 FOR I = 1 TO NU: READ ADV6(I): NEXT I
210 FOR I = 1 TO NU: READ ADV7(I): NEXT I
220 FOR I = 1 TO NU: READ ADV8(I): NEXT I
230 FOR I = 1 TO NU: READ ADV9(I): NEXT I
240 FOR I = 1 TO NU: READ ADV10(I): NEXT I
250 FOR I = 1 TO NU: READ ADV11(I): NEXT I
390 CLS : PRINT STRINGS(80, "_")

530 DV = PD

1000 FOR I = 1 TO NU
1110 X(I) = P(I)
1115 Q(I) = QP(I)
1120 Y(I) = Q(I)
1130 NEXT I

1140 GOSUB 2000

1200 PQIV = IV
1205 IF TM = 2 GOTO 1300

1300  Identify  appropriate  advertising  arrays  for
interpolation
1310 IF PD >= P(1) AND PD <= P(2) THEN GOTO 1600
1320 IF PD >= P(2) AND PD <= P(3) THEN GOTO 1610
1330 IF PD >= P(3) AND PD <= P(4) THEN GOTO 1620
1340 IF PD >= P(4) AND PD <= P(5) THEN GOTO 1630
1350 IF PD >= P(5) AND PD <= P(6) THEN GOTO 1640
1360 IF PD >= P(6) AND PD <= P(7) THEN GOTO 1650
1370 IF PD >= P(7) AND PD <= P(8) THEN GOTO 1660
1380 IF PD >= P(8) AND PD <= P(9) THEN GOTO 1670
1390 IF PD >= P(9) AND PD <= P(10) THEN GOTO 1680
1400 IF PD >= P(10) AND PD <= P(11) THEN GOTO 1690
1410 IF PD >= P(11) AND PD <= P(12) THEN GOTO 1700
```

```
1600 FOR I = 1 TO NU: TVA1(I) = ADV1(I): TVA2(I) =
ADV2(I): NEXT I
1602 PD1 = P(1): PD2 = P(2)
1605 GOTO 1720
1610 FOR I = 1 TO NU: TVA1(I) = ADV2(I): TVA2(I) =
ADV3(I): NEXT I
1612 PD1 = P(2): PD2 = P(3)
1615 GOTO 1720
1620 FOR I = 1 TO NU: TVA1(I) = ADV3(I): TVA2(I) =
ADV4(I): NEXT I
1622 PD1 = P(3): PD2 = P(4)
1625 GOTO 1720
1630 FOR I = 1 TO NU: TVA1(I) = ADV4(I): TVA2(I) =
ADV5(I): NEXT I
1632 PD1 = P(4): PD2 = P(5)
1635 GOTO 1720
1640 FOR I = 1 TO NU: TVA1(I) = ADV5(I): TVA2(I) =
ADV6(I): NEXT I
1642 PD1 = P(5): PD2 = P(6)
1645 GOTO 1720
1650 FOR I = 1 TO NU: TVA1(I) = ADV6(I): TVA2(I) =
ADV7(I): NEXT I
1652 PD1 = P(6): PD2 = P(7)
1655 GOTO 1720
1660 FOR I = 1 TO NU: TVA1(I) = ADV7(I): TVA2(I) =
ADV8(I): NEXT I
1662 PD1 = P(7): PD2 = P(8)
1665 GOTO 1720
1670 FOR I = 1 TO NU: TVA1(I) = ADV8(I): TVA2(I) =
ADV9(I): NEXT I
1672 PD1 = P(8): PD2 = P(9)
1675 GOTO 1720
1680 FOR I = 1 TO NU: TVA1(I) = ADV9(I): TVA2(I) =
ADV10(I): NEXT I
1682 PD1 = P(9): PD2 = P(10)
1685 GOTO 1720
1690 FOR I = 1 TO NU: TVA1(I) = ADV10(I): TVA2(I) =
ADV11(I): NEXT I
1692 PD1 = P(10): PD2 = P(11)
1695 GOTO 1702
1700 FOR I = 1 TO NU: TVA1(I) = ADV10(I): TVA2(I) =
ADV11(I): NEXT I
1702 PD1 = P(11): PD2 = P(12)

1720 REM INTERPOLATION OF TVA ARRAYS TO CREATE A NEW
ARRAY
1725 DV = PD
1730 FOR J = 1 TO NU
1735 X(J) = PD1: X(J + 1) = PD2: Y(J) = TVA1(J): Y(J +
1) = TVA2(J)
1740 TV = J: GOSUB 4000
1750 IV = ((DV - LX) / (HX - LX)) * (HY - LY) + LY
1760 TVAIV(J) = IV
1770 NEXT J
```

```
1800 REM INTERPOLATION OF ADV. DECISION PER NEW ARRAY,
TVAIV(I)
1810 FOR I = 1 TO NU: X(I) = ADV(I): Y(I) = TVAIV(I):
NEXT I
1820 DV = AD
1821 TV = 0

1830 GOSUB 2000
1840 AIPIV = IV
1860 CLS : PRINT STRINGS$(80, "_")
1870 PRINT TAB(10); "Decision - "; TAB(25); "Decision ";
TAB(45); "Interpolated"
1880 PRINT TAB(25); "Amount"; TAB(45); "  Value": PRINT
STRINGS$(80, "_")
1890 PRINT TAB(10); "Price"; USING "            ####
     ####"; PD; PQIV
1895 PRINT TAB(10); "Advertising"; USING "      ####
    ###.####"; AD; AIPIV
1900 INPUT "PRESS ENTER TO CONTINUE"; XX

2000 REM interpolation subroutine
2010 FOR J = 1 TO NU
2020 IF DV >= X(J) AND DV <= X(J + 1) THEN TV = J
2030 IF TV > 0 GOTO 2390
2040 NEXT J
2390 GOSUB 4000
2400 IV = ((DV - LX) / (HX - LX)) * (HY - LY) + LY
2410 TV = 0
2420 RETURN
4000 LX = X(TV + 1)
4010 HX = X(TV)
4020 LY = Y(TV + 1)
4030 HY = Y(TV)
4040 RETURN

10000 DATA 1,10,20,30,40,0,0,0,0,0,0
10110 DATA 10,4.0,3.0,2.0,1.0,0,0,0,0,0,0
10115 DATA 1,4,6,8,10,0,0,0,0,0,0
10120 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10130 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10140 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10150 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10170 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10180 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10190 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10200 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10210 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10220 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
10230 DATA .01,1.0,1.5,2.0,2.5,0,0,0,0,0,0
```