Using Beta Curves

By

Richard D. Whitlock
NASA Johnson Space Center

Mr. Richard D. Whitlock is a senior cost analyst in the Shuttle Program Control Office at the Johnson Space Center in Houston, Texas. Prior to joining NASA, he was a cost analyst with the Department of the Army for 8 years and an aerospace engineer with McDonnell Douglas for 5 years. He received his BS in Aerospace Engineering from the University of Texas at Austin, in 1967 and his MS in Operations Research from the University of California ay Irvine, 1971. He is responsible for conducting technical and cost analyses of the Space Shuttle and other advanced spacecraft and space vehicle concepts.

INTRODUCTION

In a typical large engineering program, parametric techniques are used to estimate total program costs which must then be distributed (or "spread") over time according to some schedule for budgetary and planning purposes. These costs are often spread according to some well-defined mathematical relationships designed to stimulate how costs are incurred as a function of time. The Beta distribution curve is one example of this kind of mathematical model. It has been used quite successfully for this purpose by parametric analysts and estimators for some time.

In addition to establishing a program cost estimate initially, a need exists to monitor and re-establish, if necessary, the estimate at completion (EAC) as cost information becomes available during the course of a program. One method to accomplish this is to compute the labor rates, overhead, and estimated materials costs required to complete the unfinished tasks. Although this method may be well-suited to the small dollar value project, it falls short when applied to the large, high cost program. This paper will demonstrate how the Beta distribution, in addition to being used to spread total program cost estimates, can also be used to model costs of an ongoing program in order that an EAC can be calculated. This technique will be shown to be a powerful tool for providing real-time feedback to management concerning progress toward meeting its program cost objectives. Other virtues of the Beta function, its simplicity (the only data required are the cumulative costs at various time points in the program) and its flexibility ("what-if's", such as schedule slips, can be evaluated quickly), will become evident in the discussion.

DESCRIPTION OF THE BETA CURVE MODEL

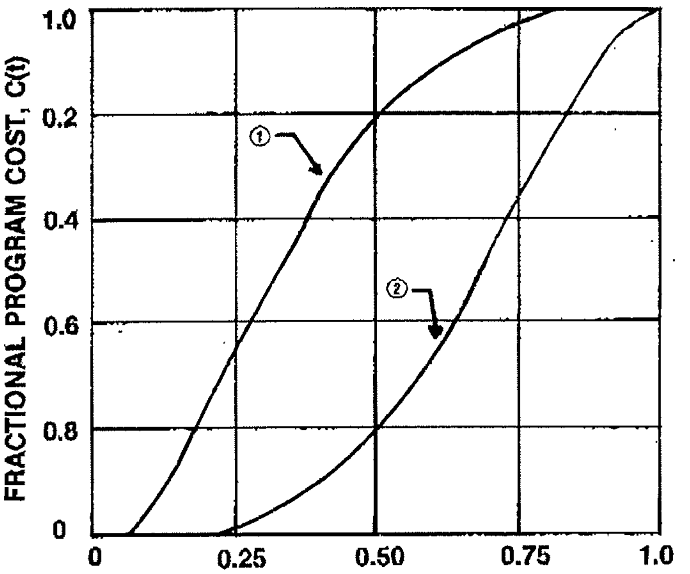
The spreading function used for distributing costs over time was originally developed by E.D. Lupo, NASA-JSC. The concept was further developed by A. Ferguson, also of NASA-JSC (Reference 1). The Cumulative cost curve, expressed as a fractional-cost, fractional-time relationships, is:

$$C(t) = A(10t^2 - 20t^3 + 10t^4) + B(10t^3 - 20t^4 + 10t^5) + (5t^4 - 4t^5) \quad [1]$$

where t is the fraction of time elapsed in the program and C(t) is the fraction of cost consumed at time t. Both t and C(t) are constrained to the interval (0,1). The cost *rate* curve (which is the derivative of the cumulative cost curve) can be expressed as:

$$C(t) = A[20t(1-t)^3] + B[30t^2(1-t)^2] + (1-A-B)[20t^3(1-t)] \quad [2]$$

It can be seen that cost rate curve is a convex combination of three integral Beta probability functions where the parameters A,B, and (1-A-B) are the relative weights given to the three Beta curves. Since the weights must be non-negative values, then A 0, B 0, and A+B 1. Although different combinations of A and B yield different cumulative cost curves, all solutions fall within the envelope shown graphically in Figure 1 and are monotonically increasing. These limits can be readily understood by referring to Figure 2 which presents the three Beta curves separately. When A=1, B=0, the cost rate curve models an extremely fast-starting program with a slow finish; when A=0, B=0, the cost rate curve models a program with a slow start and an extremely fast finish. Thus extremes in cost rate curves define the envelope of the cumulative cost curve shown in Figure 1. All other feasible combinations of A and B result in cumulative cost curves falling within these limits.

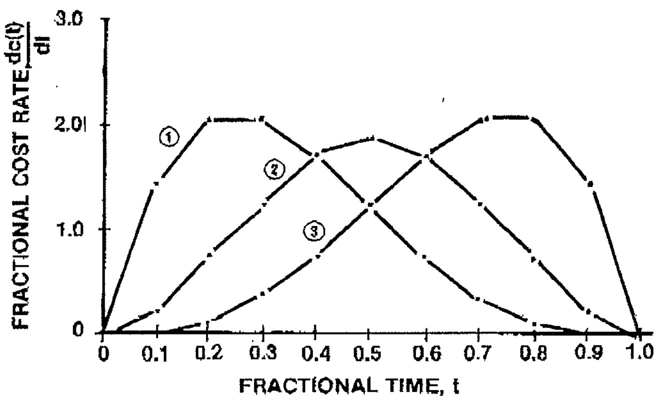


$$C(t) = A(10t^2 - 20t^3 + 10t^4) + B(10t^3 - 20t^4 + 10t^5) + (5t^4 - 4t^5)$$

① A = 1.0 and B = 0.0

② A = 0.0 and B = 0.0

Figure 1
Fractional Program Time, t



PARAMETER T: INDICATES THE % COST SPENT BY 50% TIME:
THIS SKEWS CURVE TO LEFT OR RIGHT OF CENTER.

① T = 0.8125 (A = 1; B = 0)

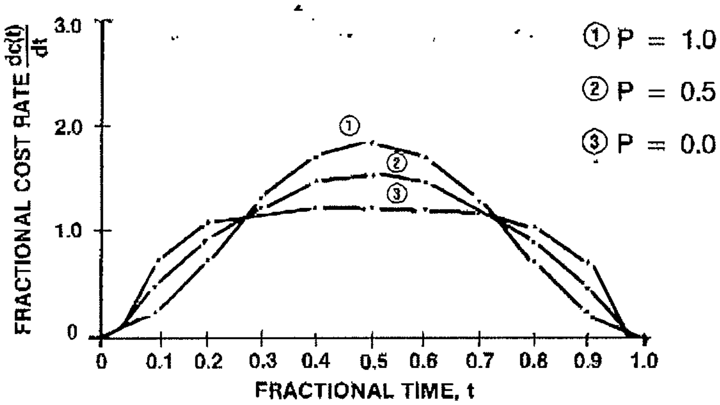
② T = 0.50 (A = 0; B =)

③ T = 0.1875 (A = 0; B = 0)

Figure 2
Effect of Parameter T On Cost Rate Curve

BETA CURVES IN TERMS OF P AND T

Beta curve characteristics are often defined in terms of a half-way cost fraction (denoted by the letter T) and a peakedness coefficient (denoted by the letter P). The half-way cost fraction is defined as the fraction of cost consumed when one-half of the program time is spent. As an illustration, the value of the parameter T is presented for the three cost rate curves shown in Figure 2. Due to the nature of the Beta curve as defined by Equation [1], T is constrained to values between 0.1875 and 0.8125, inclusive. The parameter, P, is a number between 0 and 1 which represents a relative measure of the peakedness of the cost rate curve. The most peaked (i.e. least spread out) cost rate curve is represented by P=1. The least peaked (i.e. most spread out) is represented by P=0. It is interesting to note that, as T approaches its limits of 0.1875 or 0.8125, the effect of the peakedness coefficient is reduced since, for these two values of T, unique spreading functions occur. Figure 3 presents the effect of P on the cost rate curve.



PARAMETER P:
INDICATES THE PEAKNESS OF THE COST RATE CURVE.
THIS CORRELATES TO THE BUILDUP RATE OF THE PROGRAM.

Figure 3
Effect of Parameter P On Cost Rate Curve

The following equations relate the T and P parameters to the cost rate curve weighting factors, A and B:

$$T = 0.625A + 0.3125B + 0.1875 \qquad [3]$$

$$P = 0.3125B/(0.625A + 0.3125) \quad T < 0.5 \qquad [4A]$$

$$P = 0.3125B/(0.625 - 0.3125B - 0.625A) \quad T \geq 0.5 \qquad [4B]$$

APPROACH

Typically the data that are available for an ongoing program are cumulative program costs at various dates into the program. If there are m of these data points, these can be expressed in the form (D_i, K_i) where i=1,2,3,...m and where D is the date for which the cumulative cost, K, is known. Since these are actual costs and dates, they must be converted to fractional times and fractional costs in order to be consistent with the form of Equation [1].

The first step, expressing each of the dates, D, in terms of the fractional program time, t, is accomplished with the following relationship:

$$t_i = \frac{D_i - D_{start}}{D_{end} - D_{start}} \qquad [5]$$

Finding a scaling factor which expresses each of the cumulative program costs, K, as a fractional-cost, is the next step. In fact, the problem can be formulated such that the calculation of the scaling factor will immediately yield the EAC of the program.

Let the criterion for the best fit of the actual cumulative cost data to C(t) be defined as the selection of that scaling factor which minimizes the sum of the least squares (SLS). However, since the spending pattern becomes better defined the later into a program you are, the later time points will be more indicative of program completion costs than earlier ones. Hence, the standard SLS method should be modified to give more weight to the later time points. The weighting scheme selected was to use fractional-time cubed as this weighting factor. Thus the best fit criterion becomes a weighted-sum-of-least-squares (WSLS) expression: Minimize,

$$F = WSLS = \quad [c(t_i) - YK_i]^2 t_i^3 \qquad [6]$$

where Y is the scaling factor and C(t) is defined in Equation [1].

DISCUSSION OF COMPUTATIONAL TECHNIQUES

There are several way to approach the solution to equation [6]. First, however, Equation [6] should be examined to determine what the variables are. The scaling factor, Y, is obviously a variable, whereas the K and t values are constants. Since C(t) is a function of A and B, Equation [6] implicitly requires choices for the values of A and B, as well. Whether they are treated as constants or as variables depends on the computational method used. One computational method is simply to assign arbitrary values to A and B and then solve for Y. The values to be assigned might, for example, come from another engineering program of similar size, complexity, and scope for which these values have been computed from historical data. If this method is used, A and B become constants in Equation [6]. The scaling factor, Y, corresponding to the minimum WSLS can then be computed by the standard calculus technique of setting the first derivative equal to zero and solving:

$$Y_{opt} = \frac{\Sigma C(t_i) K_i t_i^3}{t_i^3 K_i^2} \qquad [7]$$

The principal disadvantage to this method is the fact that programs do differ from one another regardless of how similar they may appear to be. Thus, assigning a particular shape of the Beta curve to a new program may not adequately allow for its uniqueness.

In recognition of this limitation another approach would be to use values for A and B which allow the shape of the Beta curve to be such that it best fits the cost-time data that the program has already experienced. Thus, unlike the previous case where the parameters A and B were specified, in this case A and B are treated as variables and the "best fit" is simply that combination of A,B, and Y which minimizes Equation [6]. There are several computational techniques that could be used to solve this problem. Three of these methods (exhaustive search, partial derivatives, and quadratic programming) are described in the following paragraphs.

In the *exhaustive search* technique, a computer algorithm is set up which ranges over all the feasible combinations of A and B, computing the corresponding scaling factor (from Equation [7]) and WSLS for each combination. Then the values for A,B and Y are selected based on which combination results in the smallest WSLS value. The major disadvantage of this technique is that the algorithm can become somewhat involved since it must be set up to search in discrete increments for A and B (such as 0.01) that are computationally manageable, and then recycled to finer increments as the true optimum is approached if increased accuracy is desired.

A more analytical approach is to use *partial derivatives* to set up a system of three unknowns. Equation [1] can be rewritten:

$$C(t) = A\alpha + B\beta + \gamma \qquad [8]$$

$$\alpha = 10t^2 - 20t^3 + 10t^4 \qquad [9A]$$

$$\beta = 10t^3 - 20t^4 + 10t^5 \qquad [9B]$$

$$\gamma = 5t^4 - 4t^5 \qquad [9C]$$

Therefore, the problem is to minimize:

$$F = WSLS = \Sigma_i [A\alpha_i + B\beta_i + \gamma_i - YK_i]^2 t_i^3 \qquad [10]$$

$$\frac{F}{A} = 2\Sigma [A_{\ i} + B\beta_i + \gamma_i - YK_i[\ \ _i t_i^3 = 0 \qquad [11A]$$

$$\frac{F}{B} = 2\Sigma [A\alpha_i + B\beta_i + \gamma_i - YK_i]\beta_i t_i^3 = 0 \qquad [11B]$$

$$\frac{F}{Y} = 2\Sigma [A\alpha_i + B\beta_i + \gamma_i - YK_i]K_i t_i^3 = 0 \qquad [11C]$$

After rearranging terms, a set of three linear equations results which can be easily solved for the unknowns A,B, and Y.

The primary drawback to this computational scheme is that there is no assurance that the solution to these equations will result in non-negative values for A and B. This disadvantage can be overcome, however, by extending the computational scheme to include boundary condition checks such that when one is exceeded, the corresponding variable is set to equal to the boundary value and the other variables recomputed.

A third approach would be to formulate the problem as a *quadratic programming* form. Using the notation introduced in Equations [8] and [9], the problem is to minimize:

$$F = WSLS = \Sigma_i [A\alpha_i + B\beta_i + \gamma_i - YK_i] \pm t_i^3 \qquad [12]$$

subject to: $G = A + B \leq 1 \qquad A \geq 0,\ B \leq 0 \qquad [13]$

This is a standard quadratic programming problem which can be restated and solved as a linear programming problem. (Reference 2). Its advantage over the previous method is that the computational algorithm automatically ensures that the solution will involve only non-negative values for the variables.

CALCUALTION OF PROGRAM EAC

Whichever method is selected to calculate the parameters, A and B and the scaling factor, Y, the following relationship exists for any data point:

$$YK_i = C(t_i) \qquad [14]$$

Equation [14] can be used to estimate the cumulative program cost for any future time interest. For the time point corresponding to the end of the program:

$$K_{END} = \frac{C(t_{END})}{Y} \qquad [15]$$

But K_{END} = EAC and $C(t_{END}) = C(1) = 1$. Therefore,

$$EAC = \frac{1}{Y} \qquad [16]$$

SUMMARY

Using the methodology described above, the Beta distribution curve, already used extensively by parametric analysts for spreading costs over time, can be as a predictive model for establishing estimates at completion for large programs. Furthermore, even for those situations where another method of computing EAC's is already established within a company or program, this method can be used as a check on the results of the existing method.

REFERENCES

1. *Estimation of Program Completion Cost,* Ferguson, Aubin F.; MSC Internal Note No. 67-ET-11, June 1967

2. *Mathematical Programming,* McMillan, Claude; John Wiley & Sons, Inc., 1970.

An Employee-Owned Company

SCIENCE APPLICATIONS INTERNATIONAL CORPORATION
6725 Odyssey Drive
Huntsville, AL 35806

REDSTAR Fax Cover Sheet

From:

Date: 8-18-05

Mary Ellen Harris
phone 256-971-6425

fax 256-971-6439

Total Pages (includes cover): 12

To:

Harrison Sterne
Recipient's Name

828-263-5901
Recipients Fax Number

REMARKS

Estimates at Completion...

7666-0030

ESTIMATES AT COMPLETION USING BETA CURVES

By: Richard D. Whitlock
NASA Johnson Space Center
Houston, Texas

ESTIMATES AT COMPLETION USING BETA CURVES

INTRODUCTION

In a typical large engineering program, parametric techniques are used to estimate total program costs which must then be distributed (or "spread") over time according to some schedule for budgetary and planning purposes. These costs are often spread according to some well-defined mathematical relationships designed to simulate how costs are incurred as a function of time. The Beta distribution curve is one example of this kind of mathematical model which has been used quite successfully for this purpose by parametric analysts and estimators for some time.

In addition to establishing a program cost estimate initially, a need exists to monitor and re-establish, if necessary, the estimate at completion (EAC) as cost information becomes available during the course of a program. One method to accomplish this is to compute the labor rates, overhead, and estimated materials costs required to complete the unfinished tasks. Although this method maybe well-suited to the small dollar value project, it falls short when applied to the large, high cost program. This paper will demonstrate how the Beta distribution, in addition to being used to spread total program cost estimates, can also be used to model costs of an on-going program in order than an EAC can be calculated. This technique will be shown to be a powerful tool for providing real-time feedback to management concerning its progress toward meeting its program cost objectives. Its other virtues; its simplicity (the only data required are the cumulative costs at various time points in the program) and its flexibility ("what-if's", such as schedule slips, can be evaluated quickly), will become evident in the discussion.

DESCRIPTION OF THE BETA CURVE MODEL

The spreading function used for distributing costs over time was originally developed by E. D. Lupo, NASA-JSC. The concept was further developed by A. Ferguson, also of NASA-JSC (Reference 1). The cumulative cost curve, expressed as a fractional-cost, fractional-time relationships, is:

$$C(t) = A(10t^2 - 20t^3 + 10t^4) + B(10t^3 - 20t^4 + 10t^5) + (5t^4 - 4t^5) \quad [1]$$

where t is the fraction of time elapsed in the program and C(t) is the fraction of cost consumed at time t. Both t and C(t) are constrained to the interval (0,1). The cost rate curve (which is the derivative of the cumulative cost curve) can be expressed as:

$$C'(t) = A[20t(1-t)^3] + B[30t^2(1-t)^2] + (1-A-B)[20t^3(1-t)] \quad [2]$$

It can be seen that the cost rate curve is a convex combination of three integral Beta probability functions where the parameters A, B, and (1-A-B) are the relative weights given to the three Beta curves. Since the weights must be non-negative values, then $A \geq 0$, $B \geq 0$, and $A+B \leq 1$. Although different combinations of A and B yield different cumulative cost curves, all solutions fall within the envelope shown graphically in Figure 1 and are monotonically increasing. These limits can be readily understood by referring to Figure 2 which presents the three Beta curves separately. When A=1, B=0, the cost rate curve models an extremely fast-starting program with a slow finish; when A=0, B=0, the cost rate curve models a program with a slow start and an extremely fast finish. Thus extremes in cost rate curves define the envelope of the cumulative cost curve shown in Figure 1. All other feasible combinations of A and B result in cumultive cost curves falling within these limits.

BETA CURVE IN TERMS OF P AND T

Beta curve characteristics are often defined in terms of a half-way cost fraction (denoted by the letter T) and a peakedness coefficient (denoted by the letter P). The half-way cost fraction is defined as the fraction of cost consumed when one-half of the program time is spent. As an illustration, the value of the parameter T is presented for the three cost rate curves shown in Figure 2. Due to the nature of the Beta curve as defined by Equation [1], T is constrained to values between 0.1875 and 0.8125, inclusive. The parameter, P, is a number between 0 and 1 which represents a relative measure of the peakedness of the cost rate curve. The most peaked (i.e., least spread out) cost rate curve is represented by P=1. The least peaked (i.e., most spread out) is represented by P=0. It is interesting to note that, as T approaches its limits of 0.1875 or 0.8125, the effect of the peakedness coefficient is reduced since, for these two values of T, unique spreading functions occur. Figure 3 presents the effect of P on the cost rate curve.

The following equations relate the T and P parameters to the cost rate curve weighting factors, A and B:

$$T = 0.625A + 0.3125B + 0.1875 \qquad [3]$$

$$P = 0.3125B/(0.625A + 0.3125) \quad T < 0.5 \qquad [4A]$$

$$P = 0.3125B/(0.625 - 0.3125A - 0.625B) \quad T \geq 0.5 \qquad [4B]$$

APPROACH

Typically the data that are available for an on-going program are cumulative program costs at various dates into the program. If there are m of these data points, these can be expressed in the form (D_i, K_i) where i=1, 2, 3, ... m and where D is the date for which the cumulative cost, K, is known. Since these are actual costs and dates, they must be converted to fractional-times and fractional-costs in order to be consistent with the form of Equation [1].

The first step, expressing each of the dates, D, in terms of the fractional program time, t, is accomplished with the following relationship:

$$t_i = \frac{D_i - D_{start}}{D_{end} - D_{start}} \qquad [5]$$

Finding a scaling factor which expresses each of the cumulative program costs, K, as a fractional-cost, is the next step. In fact, the problem can be formulated such that the calculation of the scaling factor will immediately yield the EAC of the program.

Let the criterion for the best fit of the actual cumulative cost data to C(t) be defined as the selection of that scaling factor which minimizes the sum of the least square (SLS). However, since the spending pattern becomes better defined the later into a program you are, the later time points will be more indicative of program completion costs than earlier ones. Hence, the standard SLS method should be modified to give more weight to the later time points. The weighting scheme selected was to use fractional-time cubed as this weighting factor. Thus the best fit criterion becomes a weighted-sum-of-least-squares (WSLS) expression: Minimize,

$$F = WSLS = \sum_{i=1}^{m} [C(t_i) - YK_i]^2 t_i^3 \qquad [6]$$

where Y is the scaling factor and C(t) is defined in Equation [1].

DISCUSSION OF COMPUTATIONAL TECHNIQUES

There are several ways to approach the solution to Equation [6]. First, however, Equation [6] should be examined to determine what the variables are. The scaling factor, Y, is obviously a variable, whereas the K 's and t 's are constants. Since C(t) is a function of A and B, Equation [6] implicitly requires choices for the values of A and B as well. Whether they are treated as constants or as variables depends on the computational method used. One computational method is simply to assign arbitrary values to A and B and then solve for Y. The values to be assigned might, for example, come from another engineering program of similar size, complexity, and scope for which these values have been computed from historical data. If this method is used, A and B become constants in Equation [6]. The scaling factor, Y, corresponding to the minimum WSLS can then be computed by the standard calculus technique of setting the first derivative equal to zero and solving:

$$Y_{opt} = \frac{\sum_i C(t_i) K_i t_i^3}{\sum_i t_i^3 K_i^2} \qquad [7]$$

The principle disadvantage to this method is the fact that programs do differ from one another regardless of how similar they may appear to be. Thus, assigning a particular shape of the Beta curve to a new program may not adequately allow for its uniqueness.

In recognition of this limitation another approach would be to use values for A and B which allow the shape of the Beta curve to be such that it best fits the cost-time data that the program has already experienced. Thus, unlike the previous case where the parameters A and B were specified, in this case A and B are treated as variables and the "best fit" is simply that combination of A, B, and Y which minimizes Equation [6]. There are several computational techniques that could be used to solve this problem. Three of these methods (exhaustive search, partial derivatives, and quadratic programming) are described in the following paragraphs.

In the <u>exhaustive search</u> technique, a computer algorithm is set up which ranges over all the feasible combinations of A and B, computing the corresponding scaling factor (from Equation [7]) and WSLS for each combination. Then the values for A, B, and Y are selected based on which combination results in the smallest WSLS value. The major disadvantage of this technique is that the algorithm can become somewhat involved since it must be set up to search in discrete increments for A and B (such as 0.01) that are computational manageable and then recycled to finer increments as the true optinum is approached if increased accuracy is desired.

A more analytical approach is to use partial derivatives to set up a system of three equations in three unknowns. Equation [1] can be rewritten:

$$C(t) = A\alpha + B\beta + \gamma \quad [8]$$

where;

$$\alpha = 10t^2 - 20t^3 + 10t^4 \quad [9A]$$

$$\beta = 10t^3 - 20t^4 + 10t^5 \quad [9B]$$

$$\gamma = 5t^4 - 4t^5 \quad [9C]$$

Therefore, the problem is to minimize:

$$F = WSLS = \sum_i [A\alpha_i + B\beta_i + \gamma_i - YK_i]^2 t_i^3 \quad [10]$$

$$\frac{\partial F}{\partial A} = 2\sum_i [A\alpha_i + B\beta_i + \gamma_i - YK_i]\alpha_i t_i^3 = 0 \quad [11A]$$

$$\frac{\partial F}{\partial B} = 2\sum_i [A\alpha_i + B\beta_i + \gamma_i - YK_i]\beta_i t_i^3 = 0 \quad [11B]$$

$$\frac{\partial F}{\partial Y} = 2\sum_i [A\alpha_i + B\beta_i + \gamma_i - YK_i]K_i t_i^3 = 0 \quad [11C]$$

After rearranging terms, a set of three linear equations results which can be easily solved for the unknowns A, B, and Y.

The primary drawback to this computational scheme is that there is no assurance that the solution to these equations will result in non-negative values for A and B. This disadvantage can be overcome, however, by extending the computational scheme to include boundary condition checks such that when one is exceeded, the corresponding variable is set equal to the boundary value and the other variables recomputed.

A third approach would be to formulate the problem as a quadratic programming form. Using the notation introduced in Equations [8] and [9], the problem is to minimize:

$$F = WSLS = \sum_i [A\alpha_i + B\beta_i + \gamma_i - YK_i]^2 t_i^3 \quad [12]$$

subject to: $G = A + B \leq 1$ $\quad A \geq 0,\ B \geq 0$ [13]

This is a standard quadratic programming problem which can be restated and solved as a linear programming problem. (Reference 2). Its advantage over the previous method is that the computational algorithm automatically insures that the solution will involve only non-negative values for the variables.

CALCULATION OF PROGRAM EAC

Whichever method is selected to calculate the parameters, A and B and the scaling factor, Y, the following relationship exists for any data point,

$$YK_i = C(t_i) \qquad [14]$$

Equation [14] can be used to estimate the cumulative program cost for any future time interest. For the time point corresponding to the end of the program,

$$K_{END} = \frac{C(t_{END})}{Y} \qquad [15]$$

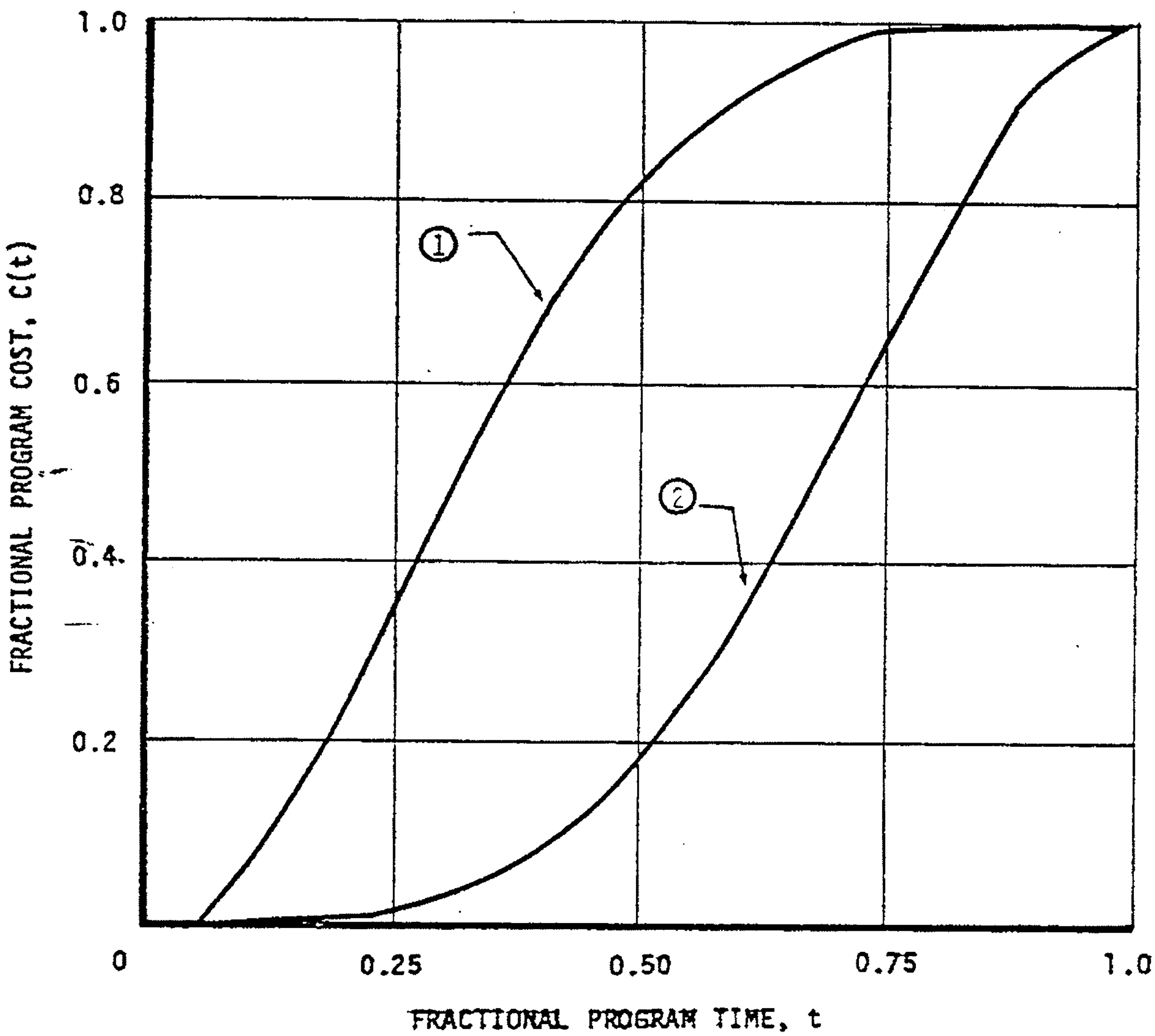
But K_{END} = EAC and $C(t_{END}) = C(1) = 1$. Therefore,

$$EAC = \frac{1}{Y} \qquad [16]$$

REFERENCES

1. Estimation of Program Completion Cost; Ferguson, Aubin F.; MSC Internal Note No. 67-ET-11, June 1967.

2. Mathematical Programming; McMillan, Claude; John Wiley & Sons, Inc., 1970.



$$C(t) = A\,(10t^2 - 20t^3 + 10t^4) + B(10t^3 - 20t^4 + 10t^5) + (5t^4 - 4t^5).$$

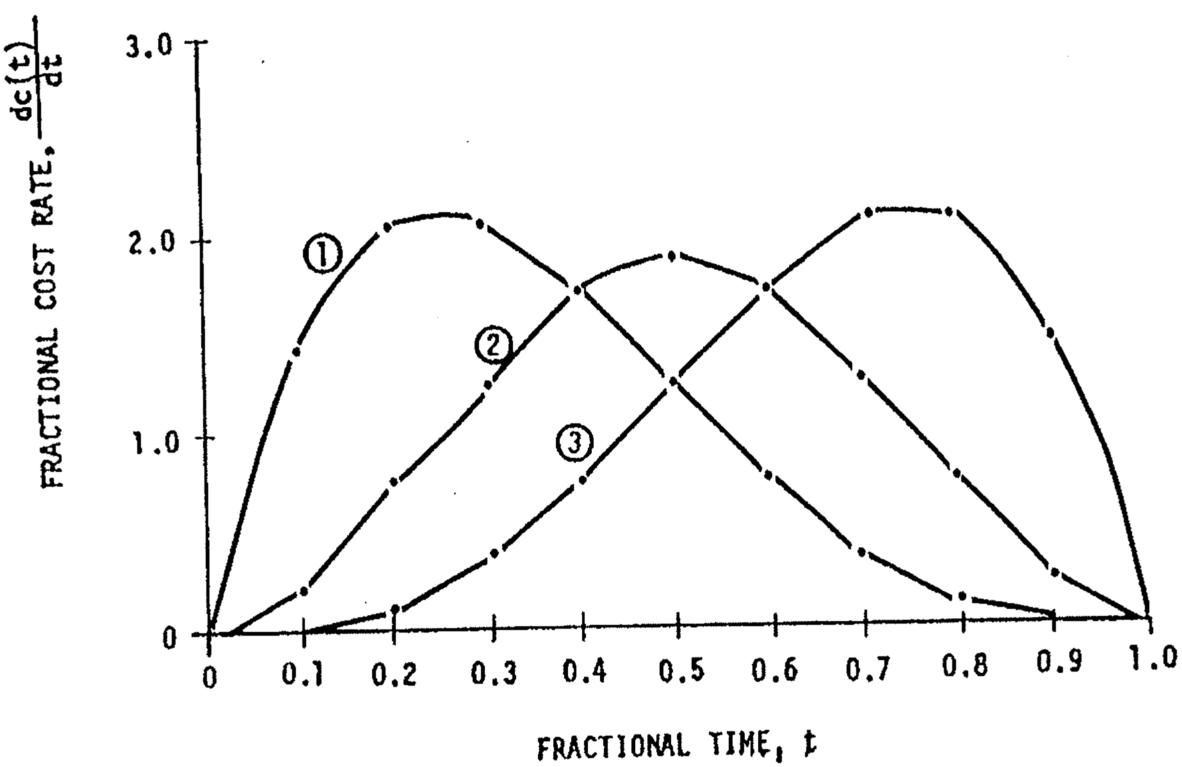
① A = 1.0 and B = 0.0

② A = 0.0 and B = 0.0

FIGURE]

EFFECT OF PARAMETER T ON COST RATE CURVE

PARAMETER T: INDICATES THE % COST SPENT BY 50% TIME.
THIS SKEWS CURVE TO LEFT OR RIGHT OF CENTER.



1 T = 0.8125 (A=1; B=0)
2 T = 0.50 (A=0; B=1)
3 T = 0.1875 (A=0; B=0)

FIGURE 2

EFFECT OF PARAMETER P ON COST RATE CURVE

PARAMETER P: INDICATES THE PEAKNESS OF THE COST RATE CURVE.
THIS CORRELATES TO THE BUILDUP RATE OF THE PROGRAM.

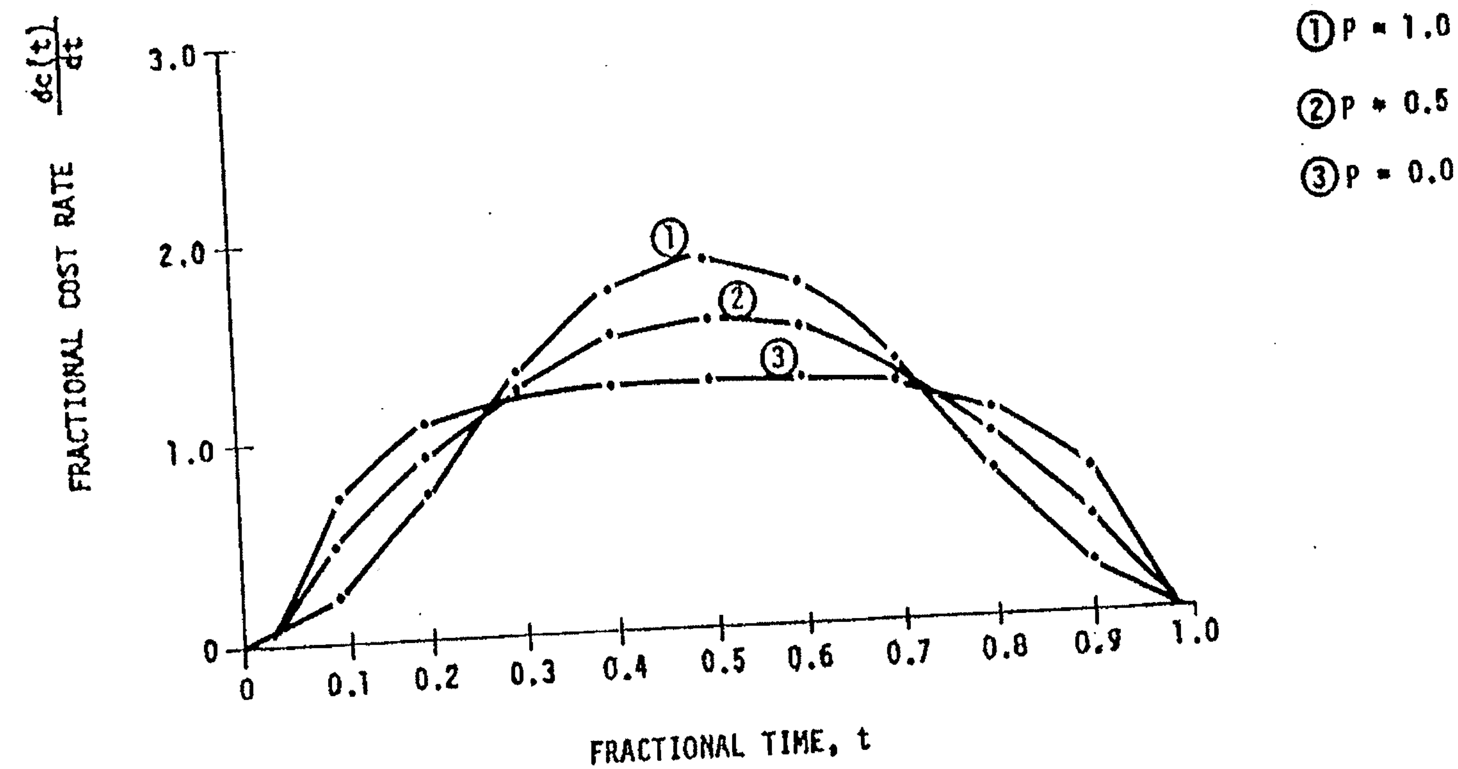


FIGURE 3