A Method of Univariate Interpolation that Has the Accuracy of a Third-Degree Polynomial

HIROSHI AKIMA
U.S. National Telecommunications and Information Administration

A method of interpolation that accurately interpolates data values that satisfy a function is said to have the accuracy of that function. The desired or required properties for a univariate interpolation method are reviewed, and the accuracy of a third-degree polynomial is found to be one of the desired properties. A method of univariate interpolation having the accuracy of a third-degree polynomial while retaining the desired properties of the method developed earlier by Akima (*J. ACM 17*, 4 (Oct. 1970), 589–602) has been developed. The newly developed method is based on a piecewise function composed of a set of polynomials, each of (at most) degree three, and applicable to successive intervals of the given data points. The method estimates the first derivative of the interpolating function (or the slope of the curve) at each given data point from the coordinates of seven data points. The resultant curve looks natural in many cases when the method is applied to curve fitting. The method is presented with examples. The possible use of a higher-degree polynomial in each interval is also examined.

Categories and Subject Descriptors: G.1.1. [**Numerical Analysis**]: Interpolation–*spline and piecewise polynomial interpolation*

General Terms: Algorithms

Additional Key Words and Phrases: Curve fitting, third-degree polynomial, univariate interpolation

1. INTRODUCTION

Interpolation is a mathematical procedure for supplying intermediate terms in a given series of terms. In this paper, we consider interpolation of univariate (one-variable) single-valued functions. We seek a method of interpolation that produces a natural-looking curve when it is applied to curve fitting. (When there is no risk of confusion, the two terms "interpolation" and "curve fitting" are used synonymously.)

Some time ago, Akima [2, 3] developed a method (hereafter referred to as the original A method) that produces natural-looking curves. In many cases

Author's address: Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U.S. Department of Commerce, 325 Broadway, Boulder, CO 80303-3328.

it suppresses excessive undulations (or wiggles) of the curves. It is described in several textbooks [5, 8] and is included in the IMSL (International Mathematical and Statistical Libraries, Inc.) Library under the routine name of IQHSCU [15]. Examinations of the method with some "hostile" examples, however, have revealed that it needs further improvement.

There are several aspects of interpolation. A method emphasizes an aspect. The method developed by Fritsch and Carlson [11] or its improved version by Fritsch [9] or by Fritsch and Butland [10] (referred to collectively as the F-C-B method) outperforms the original A method when monotonicity of data must be preserved. The method developed by Roulier [19] or by McAllister and Roulier [17, 18] (referred to collectively as the M-R method) preserves convexity of data in addition to monotonicity. There are also many other shape-preserving methods, as described by Gregory [12].

In the primitive stage of development, a given method can in most cases be superior to others. In the advanced stage, however, a given method is superior only in some cases. There are several desirable properties of interpolation, depending on the particular purposes of the user. Since some desirable properties are mutually incompatible (as discussed later), any one method cannot possess all the desired properties simultaneously.

In this paper we identify the required or desired properties for an interpolation method, discuss their mutual compatibility, and establish our goals for developing an improved method in Section 2. One of our goals is to develop a method that has the accuracy of a third-degree (cubic) polynomial, i.e., a method that interpolates accurately when the given set of data points lies on a cubic curve. We have developed a method (referred to as the improved A method) that meets these goals in Section 3. Like the original A method, the improved A method does not always preserve monotonicity or convexity. We propose the improved A method as a replacement for the original A method when the natural appearance of the resultant curve is important; we do not propose it as a replacement for the F-C-B or M-R methods when monotonicity or convexity must be preserved.

In addition, the behavior of some interpolating functions in a unit interval is examined in the Appendix, and possible use of higher-degree polynomials is considered as a variation of the improved A method in Section 4. Some examples are shown in Section 5. A summary and remarks for the use of the improved A method are given in Section 6. A Fortran subroutine subprogram that implements the improved A method is presented in the accompanying algorithm [4].

Throughout this paper, we use the following conventions:

—the x and y variables represent the independent variable and the function value;

—the x and y variables also represent the abscissa and ordinate of a two-dimensional Cartesian coordinate;

—the first derivative of the function (or the slope of the curve) is represented by y';

—the x, y, and y' values at data point P_i are represented by x_i, y_i, and y'_i; and

—the sequence $\{x_i\}$ is in an increasing order.

2. REQUIRED OR DESIRED PROPERTIES OF INTERPOLATION

In order for the curve to be smooth when interpolation is applied to curve fitting, we require that the interpolating function and its first derivative be continuous (i.e., in mathematical terms, that the function be C^1 continuous).

Since in many cases we want the curve to be affected only in a small neighborhood of the data point when a data point is added, deleted, or moved, we require that the method be based on local procedures confined to a small neighborhood of the point at which interpolation is required. (This requirement was recognized before the turn of the century.) A method based on local procedures was developed by Karup [16] and later improved by Ackland [1]. We do not consider global methods such as the spline-function method [6, 13]. Although some traditional methods such as Lagrange's and Newton's [7, 14] are based on local procedures, we do not consider them either, since they fail to produce a C^1-continuous curve.

Symmetry of the method is another required property. Symmetry means the method produces a symmetric curve when the data points are symmetric.

Invariance under certain types of coordinate transformation is a desirable property in some applications. The desirability of invariance under a linear-scale transformation

$$\begin{aligned} x &= au \\ y &= bv, \end{aligned} \tag{1}$$

where a and b are nonzero constants, is obvious. In some cases, invariance under another type of linear transformation

$$\begin{aligned} x &= au \\ y &= bu + cv, \end{aligned} \tag{2}$$

where a, b, and c are nonzero constants, is also desirable. In studying the fluctuation of a clock, for example, one may plot either the reading of the clock against the correct time (as an almost 45°-slope curve) or the error against the correct time (as an almost horizontal curve), and both plottings should represent physically the same phenomenon.

The following are other desirable properties in various applications:

—Continuity of the method. Continuity is the property that the resultant curve changes very little when a small change is made in the input data.

—Linearity of the method. Linearity is the property that the interpolated values satisfy $y(x) = af(x) + bg(x)$ if $y_i = af(x_i) + bg(x_i)$ for all i, where a and b are nonzero constants and $f(x)$ and $g(x)$ are functions of x.

—Improving the behavior of the resulting curve by insertion of an additional data point.

—Preserving the shape of data such as monotonicity and convexity. Monotonicity must be preserved, for example, when the input data point set represents a probability function.
—Producing a curve that looks natural. A skilled draftsman draws a natural-looking curve with French curves, and we want our method to do likewise. We emphasize this property in developing the improved method.

A natural-looking curve has not been defined mathematically. We describe here the behavior of a curve that makes the curve look unnatural. We know intuitively that a curve looks unnatural if it has excessive undulations, excessive inflection points, or if a line segment is embedded in a generally curved portion of the curve. Such behaviors must be avoided if possible.

Some requirements for producing a natural-looking curve need adjustment and compromise. Suppressing excessive undulations requires that, when several successive data points are on a straight line and other data points are elsewhere, the portion of the curve that connects the collinear data points be a line segment. The requirement of a line segment for several collinear data points and the requirement for suppressing embedded line segments need mutual adjustment. If we require a line segment for three collinear data points, this requirement tends to produce unnatural-looking line segments. We feel that a deviation from the line segment should be allowed when only *three* data points are collinear. We therefore require a line segment when *four* data points or more are collinear. (The original A method produces a line segment when only three data points are collinear, and sometimes fails to produce a natural-looking curve.)

In addition to the above, we also describe the property of producing a natural-looking curve in terms of the accuracy of a mathematical function. We say that an interpolation method has the accuracy of a function if the method interpolates accurately when the data points lie on a curve of the function. We also know intuitively that curves of some simple functions such as a low-degree polynomial or a sinusoidal function look good and natural. A third-degree polynomial is a polynomial of the lowest degree that can have an inflection point, so we require the accuracy of a third-degree polynomial for our method. (Ackland's osculatory method has the accuracy of a second-degree polynomial. Karup's method and the original A method have the accuracy of a second-degree polynomial conditionally, when the given data points are equally spaced in their abscissas.)

Requiring a line segment for *three* collinear data points is incompatible with the requirement of accuracy for a third-degree polynomial, while requiring a line segment for *four* collinear data points is compatible. This is because a straight line can intersect a cubic curve at three points at most.

Since a third-degree polynomial remains a third-degree polynomial under the coordinate transformations represented by (1) and (2), the requirement for the accuracy of a third-degree polynomial is consistent with the requirement for invariance under those coordinate transformations.

Regardless of the number of collinear data points, the requirement of a line segment for several collinear data points is incompatible with the

requirement for continuity of the method. For example, the curve resulting from the original A method (that produces a line segment for three collinear data points) changes abruptly when three data points P_{i-2}, P_{i-1}, and P_i are collinear and three data points P_i, P_{i+1}, and P_{i+2} are changed from almost collinear to exactly collinear. Although we have increased the number of collinear data points for a line segment from three to four, similar discontinuous behaviors can occur. We drop the requirement for continuity.

Regardless of the number of collinear data points, again, the requirement of a line segment for several collinear data points is incompatible with the requirement for linearity of the method. We do not require linearity in developing an improved method. (The original A method that produces a line segment for three collinear data points is nonlinear.)

The property of preserving monotonicity dictates that the portion of the curve between a pair of successive data points having an identical ordinate value must be a horizontal line segment, and the property of preserving convexity dictates that the portion of the curve that connects three collinear data points must be a line segment. Thus, the monotonicity or convexity requirement tends to produce embedded line segments and is incompatible with the requirement for accuracy of a third-degree polynomial. It may not be a good idea to require preserving monotonicity or convexity when such a property is not really needed. Since we already have the F-C-B and M-R methods as good interpolation methods that preserve monotonicity or convexity, we drop the requirement for preserving monotonicity or convexity.

3. THE METHOD

We first review the basic procedures of the osculatory method which has the accuracy of a second-degree polynomial [1] and the original A method [2, 3] which has some of the desired properties.

In both methods, the interpolating function is a piecewise function composed of a set of third-degree polynomials. The third-degree polynomial for the y value in the interval between x_i and x_{i+1} is represented by

$$y = a_0 + a_1(x - x_i) + a_2(x - x_i)^2 + a_3(x - x_i)^3. \tag{3}$$

The coefficients of the polynomial are determined by the given y values and the estimated y' values at the endpoints of the interval, as

$$\begin{aligned} a_0 &= y_i, \\ a_1 &= y'_i, \\ a_2 &= -\left[2(y'_i - m_i) + (y'_{i+1} - m_i)\right]/(x_{i+1} - x_i), \\ a_3 &= \left[(y'_i - m_i) + (y'_{i+1} - m_i)\right]/\left[(x_{i+1} - x_i)^2\right], \end{aligned} \tag{4}$$

where m_i is the slope of the line segment connecting P_i and P_{i+1}, and is represented by

$$m_i = (y_{i+1} - y_i)/(x_{i+1} - x_i). \tag{5}$$

The only difference between the two methods is in the procedure of estimating the first derivative of the interpolating function at each given data point.

In the osculatory method, the first derivative at P_i is estimated as the first derivative of the second-degree polynomial fitted to a set of three data points, P_{i-1}, P_i, and P_{i+1}.

In the original A method, the first derivative at P_i is estimated with a set of five data points, P_{i-2}, P_{i-1}, P_i, P_{i+1}, and P_{i+2}. Two line-segment slopes, m_{i-1} and m_i, are used as the primary estimates of the first derivative, and the final estimate is calculated as the weighted mean of the primary estimates, i.e.,

$$y'_i = (m_{i-1}w_{im} + m_i w_{ip})/(w_{im} + w_{ip}). \tag{6}$$

The weight for m_{i-1} is the reciprocal of the absolute value of the difference between m_{i-1} and m_{i-2}, and the weight for m_i is the reciprocal of the absolute value of the difference between m_{i+1} and m_i, i.e.,

$$\begin{aligned} w_{im} &= 1/\text{abs}\{m_{i-1} - m_{i-2}\}, \\ w_{ip} &= 1/\text{abs}\{m_{i+1} - m_i\}, \end{aligned} \tag{7}$$

where abs { } stands for "the absolute value of." The basic concept behind the selection of the weight is that the primary estimate based on the data points on the left (or right) side of the point in question should be given a small weight if the data points on the left (or right) side are "volatile" (or far from being collinear).

The above review suggests that the interpolation method that meets our goals have: (1) a primary estimate of the first derivative at P_i calculated as the first derivative of the third-degree polynomial fitted to every set of four consecutive data points that include P_i; (2) the final estimate of the first derivative at P_i calculated as the weighted mean of the primary estimates; and (3) the weights inversely proportional to the volatility factor of the data point set. In addition to the volatility factor, we include the distance factor in the weight; we consider that the primary estimate should be given a small weight if the data point set includes one or more data points far distant from the data point in question.

The first derivative, at data point P_i, of a third-degree polynomial fitted to a set of four data points, P_i, P_j, P_k, and P_l, is represented by

$$\begin{aligned} F(i,j,k,l) = &\big[(y_j - y_i)(x_k - x_i)^2(x_l - x_i)^2(x_l x_k) \\ &+ (y_k - y_i)(x_l - x_i)^2(x_j - x_i)^2(x_j - x_l) \\ &+ (y_l - y_i)(x_j - x_i)^2(x_k - x_i)^2(x_k - x_j)\big] \\ &/\big[(x_j - x_i)(x_k - x_i)(x_l - x_i)(x_k - x_j)(x_l - x_k)(x_l - x_j)\big]. \end{aligned} \tag{8}$$

The first index in $F(i, j, k, l)$ must be i, which is the point number of the point in question, and the remaining indices can be given in any order.

For simplicity, we take the sum of squares of deviations from a straight line of the least-square fit as the volatility factor. The volatility factor is represented by

$$V(i,j,k,l) = \Sigma\left[y - (b_0 + b_1 x)\right]^2, \tag{9}$$

where b_0 and b_1 are the coefficients of the first-degree (linear) polynomial of the least-square fit to the data points, and are represented by

$$\begin{aligned} b_0 &= \left[\Sigma x^2 \Sigma y - \Sigma x \Sigma xy\right] / \left[4\Sigma x^2 - (\Sigma x)^2\right], \\ b_1 &= \left[4\Sigma xy - \Sigma x \Sigma y\right] / \left[4\Sigma x^2 - (\Sigma x)^2\right]. \end{aligned} \tag{10}$$

In (9) and (10), Σ represents a summation over four data points, P_i, P_j, P_k, and P_l. The four indices in the expression of $V(i, j, k, l)$ in (9) can be given in any order.

For simplicity, we also take the sum of the squares of the distance from P_i to the other three data points as the distance factor. It is represented by

$$D(i,j,k,l) = (x_j - x_i)^2 + (x_k - x_i)^2 + (x_l - x_i)^2. \tag{11}$$

The first index in $D(i, j, k, l)$ must be i, and the remaining indices can be given in any order.

There are four sets of four consecutive data points that include P_i, i.e., P_{i-3} through P_i, P_{i-2} through P_{i+1}, P_{i-1}, through P_{i+2}, and P_i through P_{i+3}. The method uses four primary estimates, each calculated as the first derivative of a third-degree polynomial fitted to each set. They are represented by

$$\begin{aligned} y'_{imm} &= F(i, i-3, i-2, i-1), \\ y'_{im} &= F(i, i-2, i-1, i+1), \\ y'_{ip} &= F(i, i-1, i+1, i+2), \\ y'_{ipp} &= F(i, i+1, i+2, i+3). \end{aligned} \tag{12}$$

Since the method uses the reciprocal of the product of the volatility and distance factor, the four weights corresponding to the four primary estimates (12) are represented by

$$\begin{aligned} w_{imm} &= 1/\left[V(i, i-3, i-2, i-1)D(i, i-3, i-2, i-1)\right], \\ w_{im} &= 1/\left[V(i, i-2, i-1, i+1)D(i, i-2, i-1, i+1)\right], \\ w_{ip} &= 1/\left[V(i, i-1, i+1, i+2)D(i, i-1, i+1, i+2)\right], \\ w_{ipp} &= 1/\left[V(i, i+1, i+2, i+3)D(i, i+1, i+2, i+3)\right]. \end{aligned} \tag{13}$$

The method uses the weighted mean, that is,

$$y'_i = \left(y'_{imm}w_{imm} + y'_{im}w_{im} + y'_{ip}w_{ip} + y'_{ipp}w_{ipp}\right) / \left(w_{imm} + w_{im} + w_{ip} + w_{ipp}\right), \tag{14}$$

as the final estimate of the first derivative of the interpolating function. Since the four primary estimates are all accurate when all data points are on a curve of a third-degree polynomial, use of a weighted mean of these primary estimates is consistent with the requirement for the accuracy of a third-degree polynomial.

When a set of four data points is collinear, the V value equals zero, and the corresponding weight becomes infinite. When any weight becomes infinite, we reset infinite weights to unity and finite weights to zero before using (14) to calculate the final estimate of the first derivative.

Because of the use of the weights in (13), the method developed here has the property that, when a set of four data points P_i through P_{i+3} is collinear, the method produces a line segment across the set of data points, unless another set of four data points P_{i-3} through P_i or P_{i+3} through P_{i+6} is also collinear.

Since the method produces a line segment for four collinear data points, the curve resulting from the method changes abruptly when four data points P_{i-3} through P_i are collinear and four data points P_i through P_{i+3} are changed from almost collinear to exactly collinear. However, such discontinuous behavior occurs much less frequently for the above method than for the original A method, since the probability of having four collinear data points by chance is much less than the probability of having three collinear data points by chance.

Since the method for estimating the first derivative of the interpolating function at P_i treats the data points on both sides of P_i equally, it is consistent with the requirement for symmetry.

When the data point in question is one of the first or last three data points, all four sets of four data points are not available, and all four primary estimates cannot be calculated. In such a case, the method uses only the available primary estimate or estimates for calculating the final estimate.

Like the osculatory and original A methods, the method also uses a third-degree polynomial for the interval between each pair of successive data points. This method interpolates the y value with (3), (4), and (5). Obviously, use of a third-degree polynomial in each interval is consistent with the requirement for accuracy of a third-degree polynomial. It is also consistent with the requirement for symmetry (see Section 4).

Since the method is expected to retain the desired properties of the original A method and have the additional desired property of the accuracy of a third-degree polynomial, we call this method the improved A method.

4. USE OF A HIGHER-DEGREE POLYNOMIAL (A VARIATION)

So far we have assumed a third-degree polynomial for an interval between each pair of successive data points. A third-degree polynomial is not, however, the only function that can be used for this purpose. A higher degree polynomial and a combination of two exponential functions are some of the examples of such functions. Another example is a combination of two second-degree polynomials, which is used by the M–R method [17, 18]. The study of these functions in the Appendix indicates that the use of a higher degree polynomial or a combination of two exponential functions reduces undulations. The study also indicates that a combination of two second-degree polynomials enhances undulations and sometimes produces unnatural-looking curves. As a way of reducing undulations, we present in this section

an interpolating function based on an nth-degree polynomial, with n being equal to three or greater. (Although a higher degree polynomial and a combination of exponential functions are equally effective in reducing undulations, we prefer the former because of shorter calculation times.)

We consider an interpolating function $y = y(x)$ in the interval between P_i and P_{i+1} in a new coordinate system, called the $u - v$ coordinate system, in which u equals 0 and 1 at P_i and P_{i+1}, respectively, and v equals 0 at the two points. The linear coordinate transformation between the $u - v$ and $x - y$ coordinate systems is represented by

$$\begin{aligned} x - x_i &= (x_{i+1} - x_i)u, \\ y - y_i &= (y_{i+1} - y_i)u + v. \end{aligned} \tag{15}$$

The first derivatives in the two coordinate systems are related by

$$y' - m_i = v'/(x_{i+1} - x_i), \tag{16}$$

where

$$m_i = (y_{i+1} - y_i)/(x_{i+1} - x_i). \tag{17}$$

It is clear from (15) through (17) that

$$\begin{aligned} &u = 0, v = 0, v'_0 = (y'_i - m_i)(x_{i+1} - x_i) \qquad at \quad P_i, \\ &u = 1, v = 0, v'_1 = (y'_{i+1} - m_i)(x_{i+1} - x_i) \qquad at \quad P_{i+1}. \end{aligned} \tag{18}$$

where v'_1 and v'_0 are the v' values at $u = 0$ and $u = 1$.

As the $v(u)$ function, we present an nth-degree polynomial in u, represented by

$$v(u) = A_0[u^n - u] + A_1[(1 - u)^n - (1 - u)]. \tag{19}$$

The coefficients A_0 and A_1 are calculated by

$$\begin{aligned} A_0 &= [v'_0 + (n - 1)v'_1]/[n(n - 2)], \\ A_1 &= -[(n - 1)v'_0 + v'_1]/[n(n - 2)]. \end{aligned} \tag{20}$$

For a given x value, we can calculate the corresponding u value with the first equation in (15), the $v(u)$ value with (19), and the y value with the second equation in (15). Use of the linear combination of $f(u)$ and $f(1 - u)$ in (19) meets the requirement for symmetry in the method. When $n = 3$, the nth-degree polynomial $y(x)$ determined in this section reduces to (3) with (4) and (5).

Although use of a higher degree polynomial has an advantage in reducing undulations in the curve fitted with the interpolation method, it has a disadvantage, also. The resultant curve is sometimes too "tight," i.e., the portion of the curve between a pair of successive data points is so close to the line segment connecting the pair of data points that the whole curve looks as if it were deflected. Use of a higher degree polynomial has another disadvantage. The interpolation method does not have the accuracy of a third-degree polynomial.

We have implemented, as a user option, the use of higher degree polynomials in the accompanying algorithm [4]. Depending on the user's situation, the user can use higher degree polynomials to reduce undulations while giving up the accuracy of a third-degree polynomial. It is expected that the user will develop a general idea on the selection of the degree of the polynomial from the examples presented in Section 5.

5. EXAMPLES

This section illustrates performance of the developed methods with examples in Figures 1 through 11. In each figure, curves resulting from two existing methods are also plotted for comparison. Five curves in each figure are, from the top to bottom:

(1) the osculatory method (developed by Ackland [1]),
(2) the original A method (developed by Akima [2, 3]),
(3) the improved A method with $n = 3$ (or the improved A method without the variation described in Section 4),
(4) the improved A method with $n = 6$ (or the improved A method with the variation described in Section 4 with $n = 6$),
(5) the improved A method with $n = 10$ (or the improved A method with the variation described in Section 4 with $n = 10$).

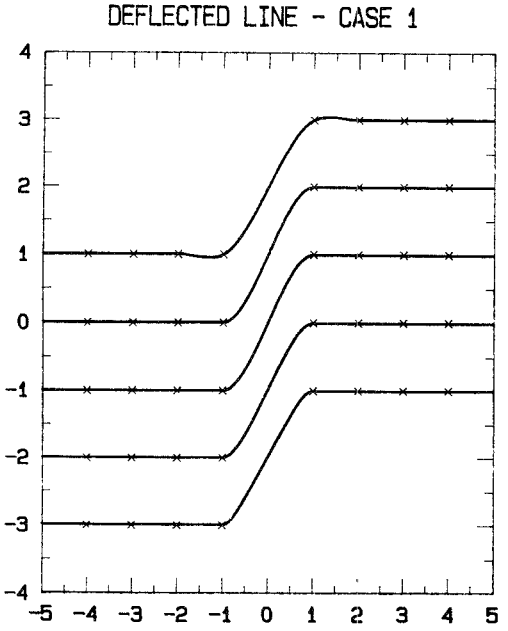
Each data point is plotted with an "$\times$" symbol. The x and y coordinate values of the data points are tabulated above the caption of each figure.

In Figure 1, data points are taken from a deflected line. The top curve (resulting from the osculatory method) exhibits overshoots in the horizontal portions of the curve, while the overshoots are nonexistent in other curves. The bottom three curves (from the improved A method) illustrate the effect of the degree of polynomials, $n = 3$ versus $n = 6$ or 10.

In Figure 2, data points are also taken from a deflected line; they consist of all data points in Figure 1 plus a data point at the center of the sloping region. The top two curves (from the osculatory method and original A method) exhibit overshoots, while the overshoots are nonexistent in the bottom three curves (from the improved A method). The bottom three curves again illustrate the effect of the degree of polynomials.

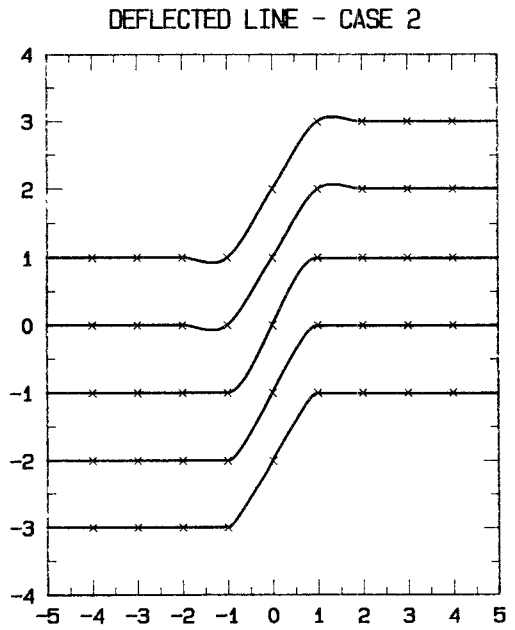
In Figure 3, the first four data points are on a horizontal straight line and the last six data points are on a curve of a third-degree polynomial, with the third and fourth points overlapping on both lines. The top curve (from the osculatory method) exhibits an overshoot, while the overshoots are nonexistent in other curves.

In Figure 4, the data points are on a cubic curve at unequal intervals. As is expected, the third curve (from the improved A method with $n = 3$) looks good, while all other curves exhibit irregularities. In the two intervals around the center point, the portion of the first curve (from the osculatory method) has inflection points. In the same intervals, the portions of the second curve (from the original A method) are line segments. Disadvantages



x =	-4	-3	-2	-1	1	2	3	4
y =	-1	-1	-1	-1	1	1	1	1

Fig. 1. Deflected-line data, Case 1.



x =	-4	-3	-2	-1	0	1	2	3	4
y =	-1	-1	-1	-1	0	1	1	1	1

Fig. 2. Deflected-line data, Case 2.

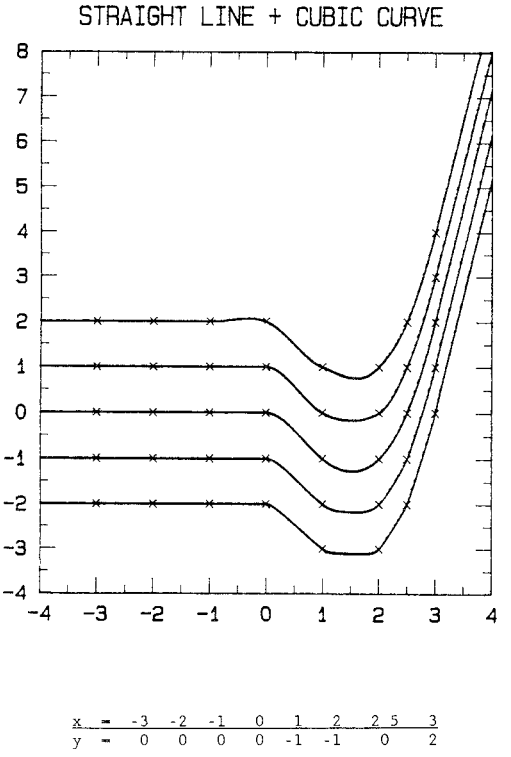


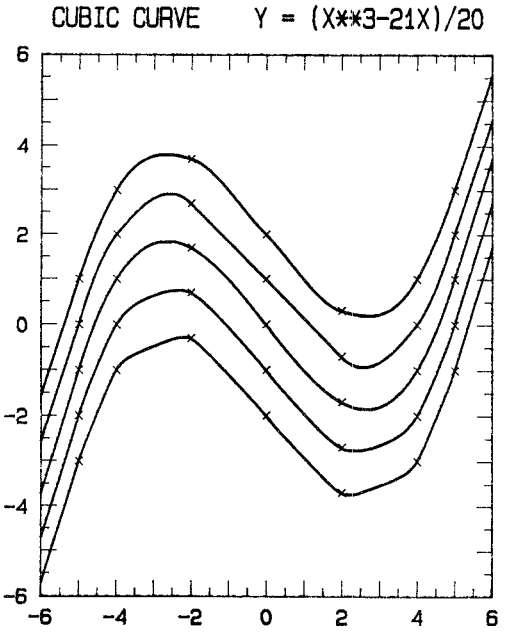
Fig. 3. Straight line plus cubic curve, $y = 0$ and $y = x^3/3 - x^2/2 - 5x/6$.

of the use of higher degree polynomials are demonstrated in the bottom two curves (from the improved A method with $n = 6$ and 10).

In Figure 5, the data points are on a sine curve at unequal intervals. In this rather contrived example, the general trends of the curves in Figure 4 are even more pronounced. Figures 4 and 5 indicate that higher degree polynomials should be used sparingly.

The data points for Figure 6 are taken from Akima [2]. The top curve (from the osculatory method) exhibits an undulation in the interval between $x = 6$ and 8, while all other curves look good. We will modify this data point set in several ways and see how the curves behave for each of the modified data point sets in the figures that follow.

The data point set for Figure 7 is Modification A. Two leftmost data points for Figure 6 (original data point set) are removed, and the remaining data points are moved horizontally. As expected, removal of the two points has no effect. The undulation in the top curve (from the osculatory method), now in the interval between $x = 8$ and 10, is more pronounced. The second curve (from the original A method) looks good. The third curve (from the improved A method with $n = 3$) exhibits a small undulation in the interval between $x = 8$ and 10, while the bottom two curves (from the improved A method with $n = 6$ and 10) do not. In the bottom three curves, the negative slopes at $x = 13$ may look a little strange, but the third curve looks good as a whole if



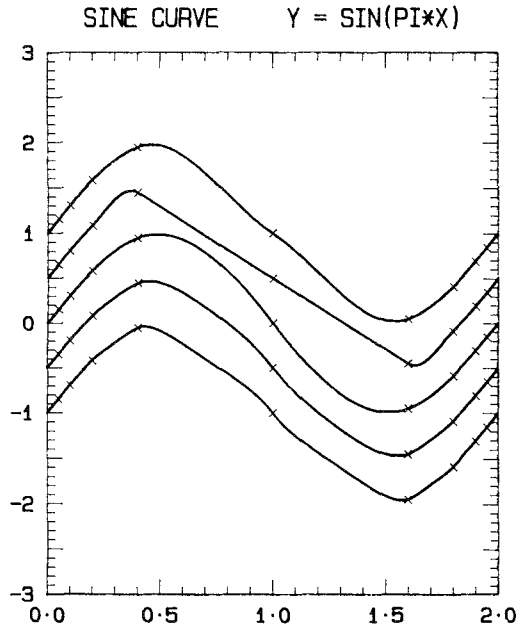
x =	-5	-4	-2	0	2	4	5
y =	-1	1	1.7	0	-1.7	-1	1

Fig. 4. Cubic curve, $y = (x^3 - 21x)/20$.

monotonicity of the curve is not required. The bottom two curves change their directions so fast around $x = 13$ that they look as if they were deflected at this point.

The data point set for Figure 8 is Modification B. It consists of the data points for Figure 7 (Modification A) and an additional data point at the center of the line segment that has the steepest slope. With this addition, the top curve (from the osculatory method) remains unacceptable. The second curve (from the original A method) exhibits a large undulation in the interval between $x = 8$ and 10. In the same interval, the undulation in the third curve (from the improved A method with $n = 3$) is a little more pronounced than in Figure 7, and a small undulation emerges even in the fourth curve (from the improved A method with $n = 6$). The behaviors of all curves around $x = 13$ remain almost unchanged from Figure 7.

The data point set for Figure 9 is Modification C. It consists of the data points for Figure 8 (Modification B) and an additional data point at $x = 9$. With this additional point, the top two curves (from the osculatory and original A methods) remain unacceptable. The third and fourth curves (from the improved A method with $n = 3$ and 6) are improved considerably; undulations that existed in the interval between $x = 8$ and 10 in Figure 8 are nonexistent in Figure 9. The slopes of all curves at $x = 13$ are unaffected by the additional data point.



x =	0 05	0 10	0 20	0 40	1 00	1.60	1.80	1 90	1 95
y =	1564	3090	5878	9511	.0000	- 9511	- 5878	- 3090	- 1564

Fig. 5. Sine curve, $y = \sin(\pi x)$.

The data point set for Figure 10 is Modification D. It consists of the data points for Figure 9 (Modification C) and an additional data point at $x = 12$. With this additional data point, the first and second curves (from the osculatory and original A methods) are degraded; an inflection point emerges in each side of the newly added data point in the first curve, and a straight line segment is embedded in a generally curved portion of the second curve. The bottom three curves (from the improved A method) are improved; the slopes of the curves at $x = 13$ look more natural than the same slopes in Figure 9. In Figure 10, the second curve from the bottom (from the improved A method with $n = 6$) is better than the third curve from the bottom (with $n = 3$); higher degree polynomials work well, without adverse side effects in this example. The bottom curve (with $n = 10$), however, is not as good as the second curve from the bottom (with $n = 6$); polynomials with an excessively high degree do not always work favorably.

The data point set for Figure 11 is Modification E, which is another modification of Modification C (but not a modification of D). It consists of the data points for Figure 9 (Modification C) and an additional data point at $x = 13.5$. With this additional data point, the first and second curves (from the osculatory and original A methods) are almost unchanged from Figure 9. The bottom three curves (from the improved A method) in this figure are

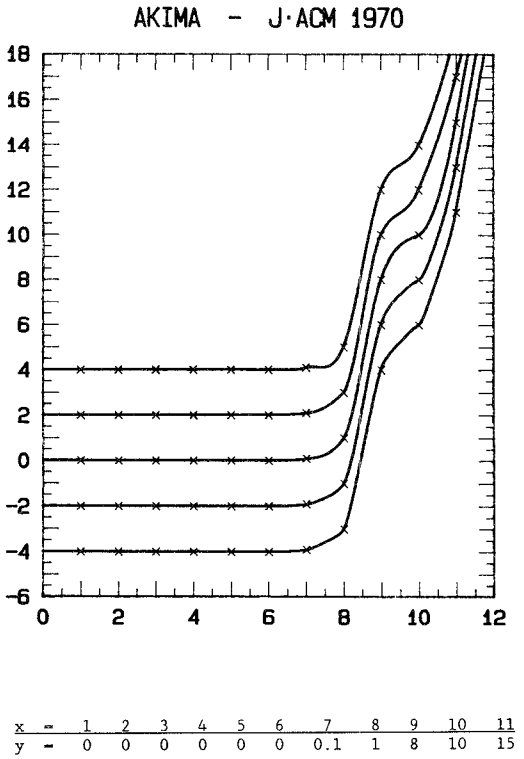


Fig. 6. Akima data [2].

better than the same curves in Figure 9; the slopes of these curves at $x = 13$ look more natural than the same slopes in Figure 9.

6. CONCLUSIONS

We have identified required or desired properties for a univariate interpolation method, discussed their mutual compatibility, and established our goals for developing a method that produces a natural-looking curve when the method is used for smooth curve fitting. We have realized that one of our goals is to develop a method that has the accuracy of a third-degree (cubic) polynomial, i.e., a method that interpolates accurately when the given data points lie on a cubic curve. We have also realized that the original A method [2, 3] has some of the desired properties. We have improved the original A method in such a way that the improved method, called the improved A method, has the accuracy of a third-degree polynomial while retaining the desired properties of the original A method. As demonstrated in the examples, the improved A method generally yields a curve that looks much more natural than what results from the original A method. We propose the improved A method as a replacement for the original A method when the natural appearance of the resultant curve is important.

Improvement has been made in the procedure of estimating the first derivative of the interpolating function at each data point. The improved A

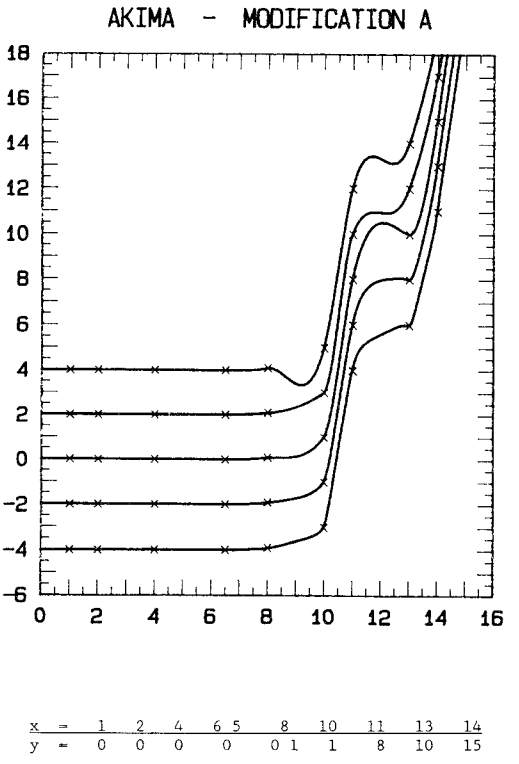
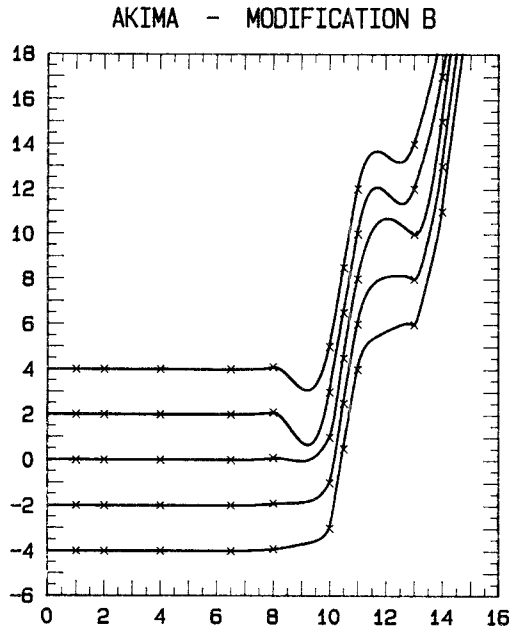


Fig. 7. Akima data, modification A.

method calculates four primary estimates for the first derivative, each as the first derivative of a third-degree polynomial fitted to a set of four consecutive data points. It calculates the final estimate of the first derivative as the weighted mean of the four primary estimates. The weight for each primary estimate is the reciprocal of the product of the volatility factor and the distance factor of the set of four data points. The sum of squares of the deviations of the ordinate values of the four data points from the straight line of least-square fit is used as the volatility factor. The sum of squares of the distances in the abscissa from the data point in question to the remaining three data points is used as the distance factor.

Like the original A method, the improved A method uses a third-degree polynomial in an interval between each pair of data points as a default. In addition, we have also implemented possible use of a higher degree polynomial for an interval as an option. Although the use of a higher degree polynomial generally reduces undulations, it sometimes distorts curves that would look good otherwise. It inevitably voids the accuracy of a third-degree polynomial of the method. A higher-degree polynomial option should therefore be exercised prudently and sparingly when naturalness of the resultant curve is important.

Like the original A method, the improved A method does not always preserve monotonicity or convexity. We did not intend to preserve it in developing the improved A method. We do not propose the improved A



x =	1	2	4	6.5	8	10	10.5	11	13	14
y =	0	0	0	0	0.1	1	4.5	8	10	15

Fig. 8. Akima data, modification B

method as a replacement for the F–C–B method [9–11] or the M–R method [17–19] when monotonicity or convexity must be preserved.

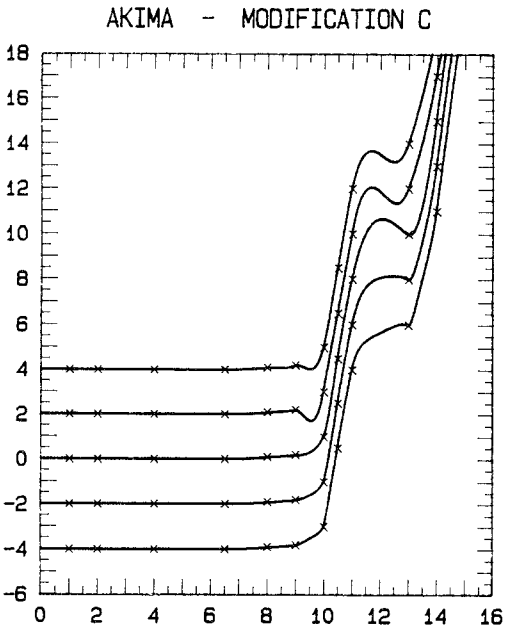
The improved A method can easily be implemented in a computer program. A FORTRAN subroutine subprogram that implements the improved A method is presented in the accompanying algorithm [4].

Since the original A method has been improved without changing its basic concept, most remarks about the original A method apply to the improved A method as well. Some remarks pertinent to proper application of the improved A method follow.

(1) The method does not smooth the data. In other words, the resultant curve passes through all the given data points if the method is applied to smooth curve fitting. Therefore, the method is applicable only when the precise y values are given or where the errors are negligible.

(2) As is true for any interpolation method, the accuracy of the improved A method cannot be guaranteed, unless it is known that the given data points lie on a curve of a third-degree polynomial.

(3) Unless the option for a higher degree polynomial is exercised, the method has the accuracy of a third-degree polynomial, i.e., the method gives exact results when y is a third-degree polynomial in x even when the data points are given at unequal intervals.



x =	1	2	4	6.5	8	9	10	10.5	11	13	14		
y =	0	0	0	0	0.1	0.2	1	4.5	8	10	15		

Fig. 9. Akima data, modification C.

(4) The method yields a smooth, natural-looking curve and is therefore useful in cases where manual curve fitting will do in principle.

(5) The method is nonlinear. In other words, if $y_i = f(x_i) + g(x_i)$ for all i, the interpolated values do not, in general, satisfy $y(x) = f(x) + g(x)$.

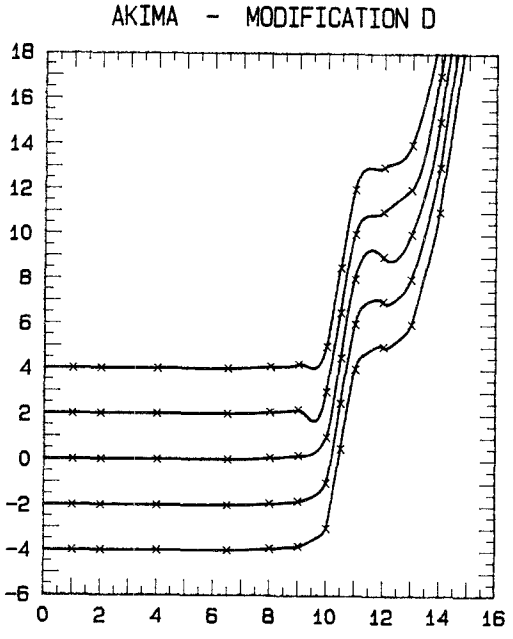
(6) The method produces a periodic curve from a set of periodic data points that covers a complete cycle if three additional data points corresponding to the preceding or following cycle are supplied on each side of the given data point set.

(7) The method requires only straightforward procedures. No problem concerning computational stability or convergence exists in the application of the method.

APPENDIX: INTERPOLATING FUNCTIONS IN A UNIT INTERVAL

In this appendix we examine the behavior of some interpolating functions in a unit interval between $x = 0$ and $x = 1$. For simplicity, we examine the behavior of an interpolating function, $y(x)$, under the following conditions:

$$x = 0: y(0) = 0,$$
$$x = 1: y(1) = 0. \tag{A-1}$$



x =	1	2	4	6.5	8	9	10	10.5	11	12	13	14
y =	0	0	0	0	0.1	0.2	1	4.5	8	9	10	15

Fig. 10. Akima data, modification D.

We consider an interpolating function that meets the symmetry requirement and has only one inflection point, at most, in the interval.

To construct an interpolating function, we first consider a basis function, $f(x)$, that satisfies the following conditions:

$$\begin{aligned} &x = 0\colon f(0) = 0, \quad f'(0) = -t, \quad 0 < t < 1, \\ &x = 1\colon f(1) = 0, \quad f'(1) = 1, \\ &0 \le x \le 1\colon f''(x) \ge 0, \quad f^{(3)}(x) \ge 0. \end{aligned} \tag{A-2}$$

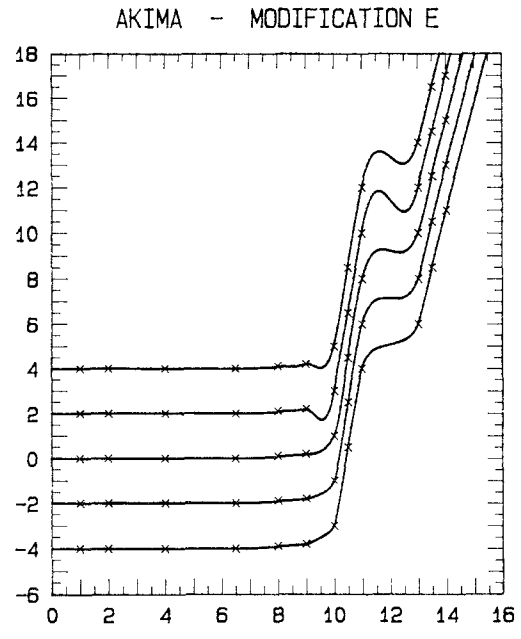
Next, we consider, as an interpolating function, a linear combination of $f(x)$ and $f(1 - x)$, represented by

$$y(x) = C_0 f(x) + C_1 f(1 - x), \tag{A-3}$$

where

$$\begin{aligned} C_0 &= \left[ty'(0) + y'(1)\right]/(1 - t^2), \\ C_1 &= -\left[y'(0) + ty'(1)\right]/(1 - t^2). \end{aligned} \tag{A-4}$$

The interpolating function thus constructed satisfies (A-1). The condition on the third derivative of $f(x)$ set in (A-2) guarantees that the second derivative



x =	1	2	4	6.5	8	9	10	10.5	11	13	13.5	14
y =	0	0	0	0	0.1	0.2	1	4.5	8	10	12.5	15

Fig. 11 Akima data, modification E.

of $y(x)$ can be zero at only one point at most, and therefore $y(x)$ can have only one inflection point at most in the interval. Equation (A-3) guarantees that $y(x)$ meets the symmetry requirement.

Although we can consider an infinite number of functions as the basis functions that satisfy (A-2), we consider only two functions here. Each function is listed with its first derivative and the t value.

(a) *nth-degree polynomial*

$$\begin{aligned} f(x) &= (x^n - x)/(n-1), \\ f'(x) &= (nx^{n-1} - 1)/(n-1), \\ t &= 1/(n-1). \end{aligned} \tag{A-5}$$

(b) *Exponential function*

$$\begin{aligned} f(x) &= \{[\exp(ax) - 1] - [\exp(a) - 1]x\}/b, \\ f'(x) &= [a\exp(ax) - \exp(a) + 1]/b, \\ t &= [\exp(a) - (1+a)]/b, \\ b &= (a-1)\exp(a) + 1. \end{aligned} \tag{A-6}$$

Perhaps the function in (a) is the simplest form we can consider as the basis

function. Because of the condition on the third derivative of $f(x)$, n must be equal to 3 or greater. When $n = 3$, it reduces to a third-degree polynomial:

$$y(x) = (x^3 - 2x^2 + x)y'(0) + (x^3 - x^2)y'(1). \tag{A-7}$$

This is commonly referred to as the cubic Hermite interpolant.

Before we proceed, we introduce the third interpolating function that does not fall in the same category as (a) and (b), but yet satisfies (A-1). It is a piecewise function composed of two second-degree polynomials joined together smoothly at $x = 0.5$, i.e., the center of the interval. It is represented as follows:

(c) *Piecewise second-degree polynomials*

$$\begin{aligned} &0 \le x \le 0.5: \quad y(x) = a_1 x + a_2 x^2, \\ &0.5 \le x \le 1: \quad y(x) = b_1(1 - x) + b_2(1 - x)^2, \end{aligned} \tag{A-8}$$

where

$$\begin{aligned} a_1 &= y'(0), \\ a_2 &= -[3y'(0) + y'(1)]/2, \\ b_1 &= -y'(1), \\ b_2 &= [y'(0) + 3y'(1)]/2. \end{aligned} \tag{A-9}$$

This function (A-8) has an advantage over the third-degree polynomial (A-7) with respect to convexity, i.e., the property that $y''(x)$ does not change its sign in the whole interval. Curves of the former are convex when $y'(0)/y'(1)$ lies between -3 and $-1/3$, while curves of the latter are convex only when $y'(0)/y'(1)$ lies -2 and $-1/2$. The interpolating function in (c) is used by McAllister and Roulier [18] and by others mainly because of this advantage.

We now present, in Figures 12–19, the behavior of these three functions graphically in a normalized form. In each figure, we plot 21 curves that correspond to the $y'(0)$ values from -1.0 to $+1.0$, with a 0.1 step from the bottom to the top. The $y'(1)$ value is set to unity for all curves. The center curve that corresponds to $y'(0) = 0$ is plotted in a heavy line. (Note that the actual scaling of the y axis is double that of the x axis.)

Curves for the function $y(x)$ based on an nth-degree polynomial (a) are plotted for the n values equal to 3, 4, 6, and 10 in Figures 12–15. These figures indicate that undulations are reduced by increasing n, but each curve approaches a straight line at the same time.

Curves for the function $y(x)$ based on an exponential function (b) are plotted for the a value equal to 1, 5, and 10 in Figures 16–18. These figures indicate that a curve of this function for an a value is very close to the curve of the nth-degree polynomial (a) for some n value. Note, however, that calculation time required for the exponential function is longer than that for the nth-degree polynomial.

Curves for the piecewise second-degree polynomials (c) are plotted in Figure 19. We notice that the curves for $y'(0) = -0.2$ and -0.3 look unnatural. Perhaps this behavior is related to the fact that, as is clear from (A-9),

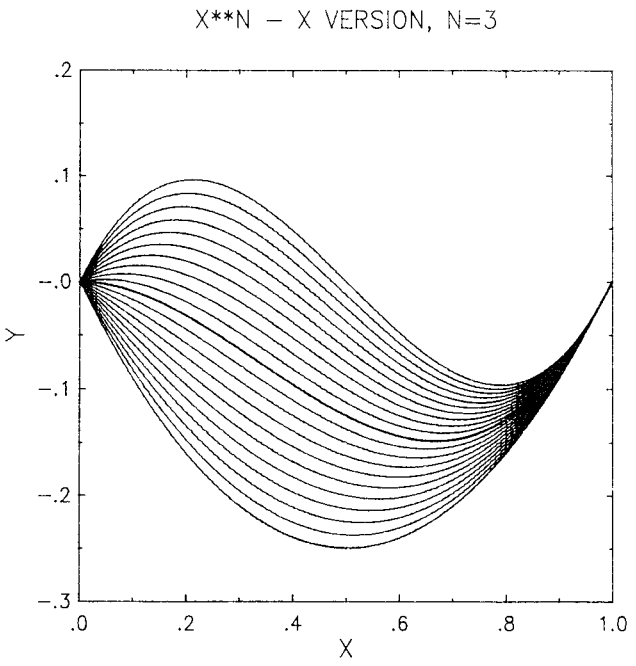


Fig. 12. Function based on an nth-degree polynomial with $n = 3$.

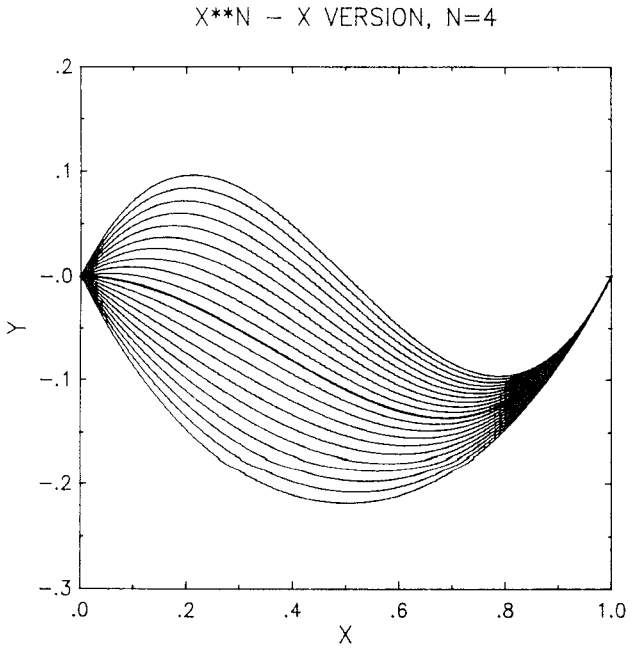


Fig. 13. Function based on an nth-degree polynomial with $n = 4$

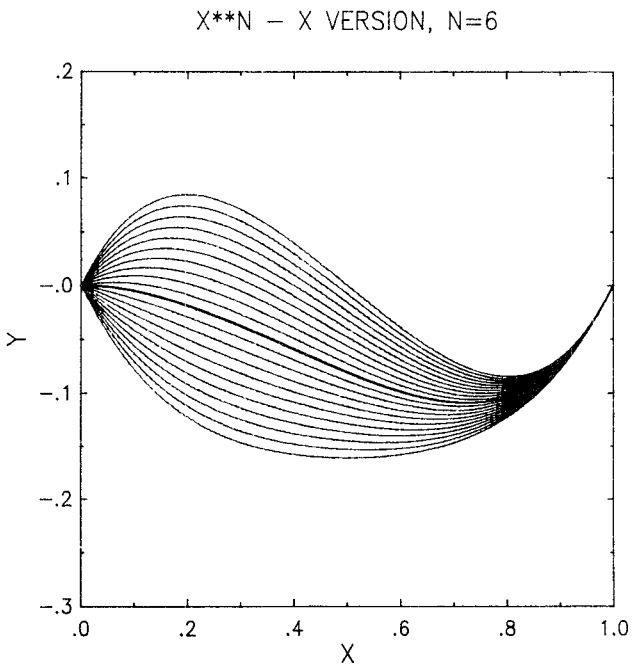


Fig. 14. Function based on an nth-degree polynomial with $n = 6$.

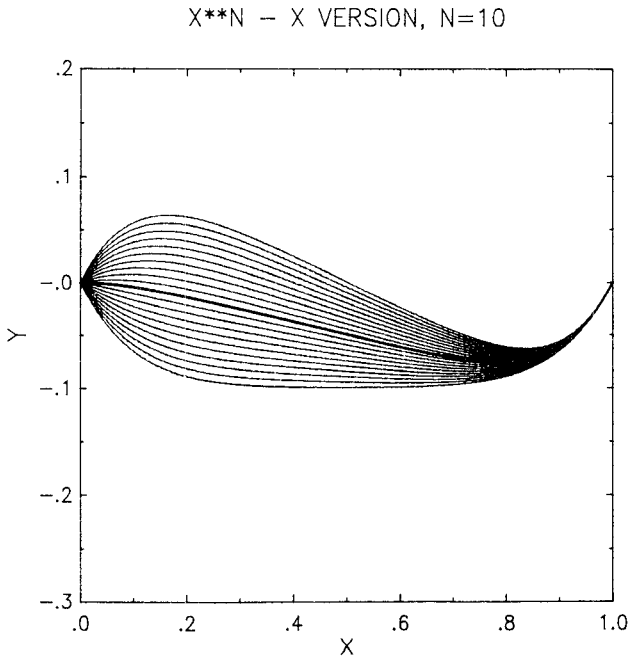


Fig. 15. Function based on an nth-degree polynomial with $n = 10$.

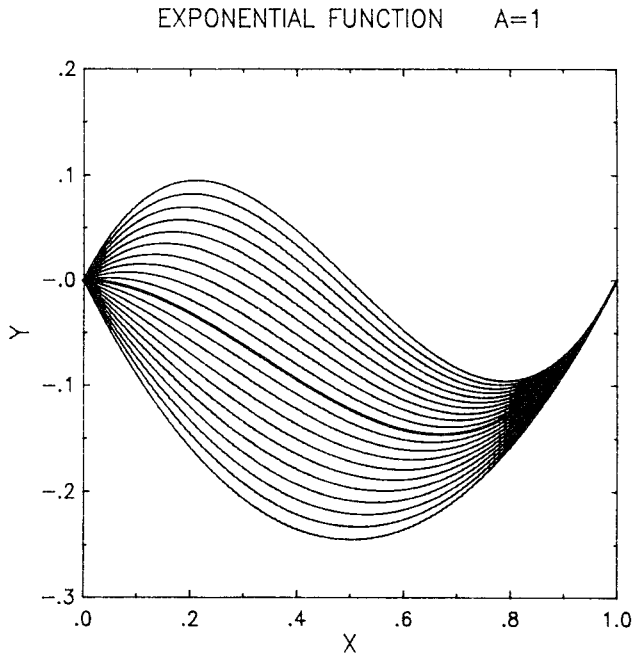


Fig. 16. Function based on an exponential function exp(ax) with $a = 1$.

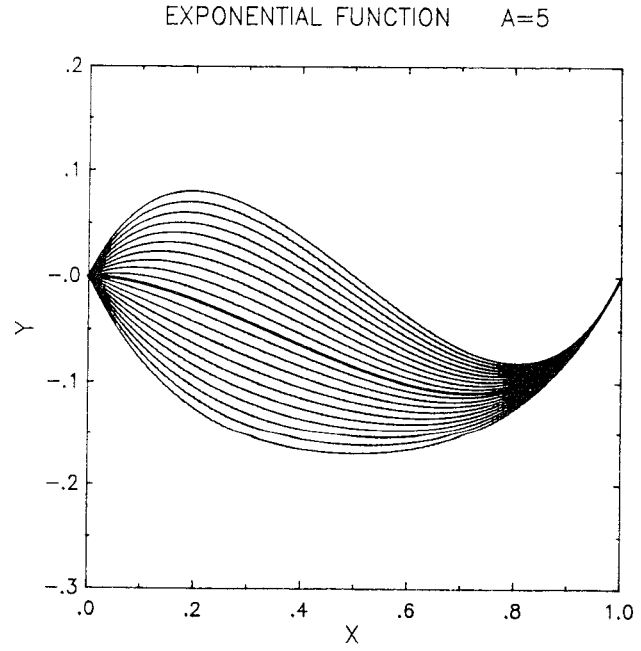


Fig. 17. Function based on an exponential function exp(ax) with $a = 5$.

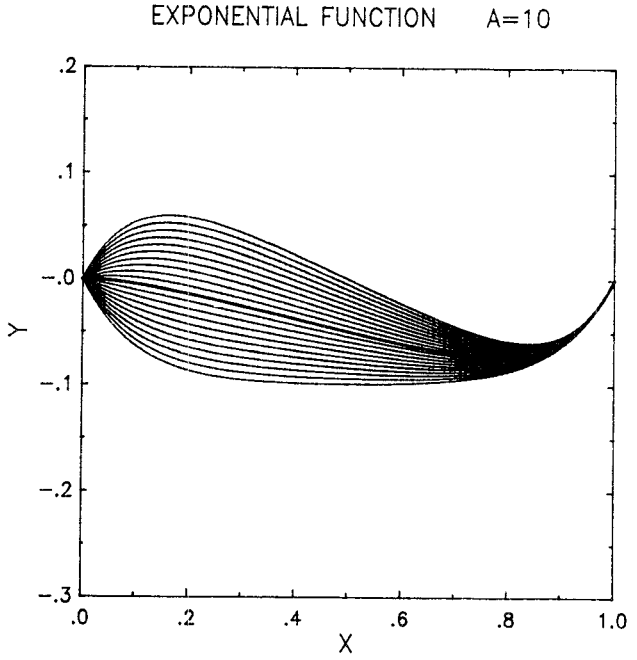


Fig. 18. Function based on an exponential function exp(ax) with $a = 10$.

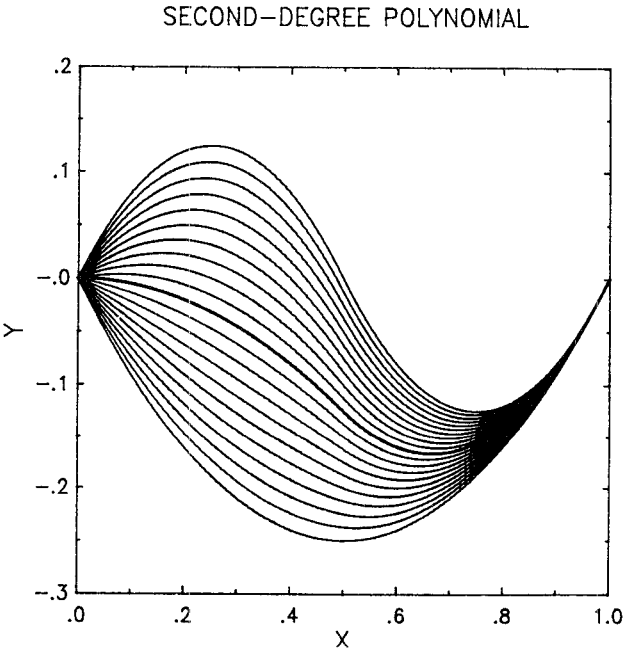


Fig. 19. Piecewise function composed of two second-degree polynomials.

the portion of the curve for $0 \leq x \leq 0.5$ is a line segment when $y'(0) = -1/3$. Also, the "amplitude" of the top curve for $y'(0) = +1$ is too large in comparison with the top curve for the third-degree polynomial in Figure 12. For these reasons, use of this piecewise function is not advisable, regardless of the advantage with respect to convexity.

ACKNOWLEDGMENT

The author is grateful to Frederick N. Fritsch of the Lawrence Livermore National Laboratory for his useful information on various methods of shape-preserving interpolation, and to George A. Hufford and A. Donald Spaulding of the Institute for Telecommunication Sciences of the National Telecommunications and Information Administration of the U.S. Department of Commerce, for their review of this paper and useful discussions on the subject of this paper.

REFERENCES

1 ACKLAND, T. G. On osculatory interpolation, where the given values of the function are at unequal intervals. *J. Inst. Actuar. 49* (Oct. 1915), 369-375.
2. AKIMA H A new method of interpolation and smooth curve fitting based on local procedures. *J. ACM 17*, 4 (Oct 1970), 589-602.
3 AKIMA, H Algorithm 433: Interpolation and smooth curve fitting based on local procedures. *Commun. ACM 15*, 10 (Oct. 1972), 914-918.
4. AKIMA, H. Algorithm 696: Univariate interpolation that has the accuracy of a third-degree polynomial. *ACM Trans. Math. Softw., 17*, 3 (Sept. 1991), 367.
5 CARNAHAN, B., AND WILKES, J. O. *Digital Computing and Numerical Methods (With FORTRAN-IV, WATFOR, and WATFIV Programming).* Wiley, New York, 1973, ch. 6.
6. CLINE, A. K. Scalar- and planar-valued curve fitting using splines under tension. *Commun. ACM 17*, 4 (Apr. 1974), 218-220
7. DAVIS, P. J. *Interpolation and Approximation.* Dover, New York, 1975.
8. DE BOOR, C. *A Practical Guide to Splines.* Springer-Verlag, New York, 1978.
9. FRITSCH, F. N. Piecewise cubic Hermite interpolation package. In *SIAM 30th Anniversary Meeting* (Stanford, Calif., July 1982), p. 71.
10. FRITSCH, F. N., AND BUTLAND, J. A method for constructing local monotone piecewise cubic interpolants. *SIAM J. Sci. Stat. Comput. 5*, 2 (June 1984), 300-304.
11. FRITSCH, F. N., AND CARLSON, R. E. Monotone piecewise cubic interpolation. *SIAM J. Numer. Anal. 17*, 2 (Apr. 1980), 238-246.
12. GREGORY, J. A review of curve interpolation with shape control. In *IMA Conference on Algorithms for Approximation of Functions and Data* (Shrivenham, England, July 1985).
13. GREVILLE, T. N. E. Spline functions, interpolation, numerical quadrature. In *Mathematical Methods for Digital Computers, Vol. 2,* A. Ralston and H. S. Wilf, Eds., Wiley, New York, 1967, ch. 8.
14. HILDEBRAND, F. B. *Introduction to Numerical Analysis.* McGraw-Hill, New York, 1956, ch. 2, 3, and 4.
15 IMSL (International Mathematical and Statistical Libraries). *IMSL Library Reference Manual,* Ed. 7 (Jan. 1979), Ch. I.
16 KARUP, J. On a new mechanical method of graduation. In *Transactions of the Second International Actuarial Congress.* C. and E. Layton, London, 1899, pp. 78-109.
17. MCALLISTER, D. F., AND ROULIER, J. A An algorithm for computing a shape-preserving osculatory quadratic spline. *ACM Trans. Math. Softw. 7*, 3 (Sept. 1981), 331-347.
18. MCALLISTER, D. F., AND ROULIER, J. A. Algorithm 574: Shape-preserving osculatory quadratic spline [E1, E2]. *ACM Trans. Math. Softw. 7*, 3 (Sept. 1981), 384-386
19. ROULIER, J. A. Constrained interpolation. *SIAM J. Sci. Stat. Comput. 1*, 3 (Sept 1980), 333-344.

Received July, 1989; accepted July, 1990