BIT 6 (1966), 277–293

SMOOTH-CURVE INTERPOLATION: A GENERALIZED SPLINE-FIT PROCEDURE

J. M. GLASS

Abstract.

A method is presented for finding the smoothest curve through a set of data points. "Smoothest" refers to the equilibrium, or minimum-energy configuration of an ideal elastic beam constrained to pass through the data points. The formulation of the smoothest curve is seen to involve a multivariable boundary-value minimization problem which makes use of a numerical solution of the beam nonlinear differential equation. The method is shown to offer considerable improvement over the spline technique for smooth-curve interpolation.

Key words: Interpolation, spline, curve.

I. Introduction.

Smooth-curve interpolation of sampled data is frequently desired for a variety of reasons. For example, the physical or mathematical process that is sampled may be known to be extremely well-behaved, with little fluctuation between sample points. As another instance, one may wish to create a smooth curve or surface between certain fixed points, a technique used in the fairing of shiplines [1, 2]. Other illustrations of the need for smooth interpolation will be readily apparent to the reader.

When the data are known precisely (no noise) an exact interpolation scheme is called for rather than a least squares data fit, particularly when considerable expense may be incurred in obtaining the data. A number of papers in the literature have described an exact smooth-curve data fit known as spline interpolation. A representative sampling of the literature on spline interpolation is listed as references [1]–[6] at the end of the paper. The spline method essentially approximates the equilibrium, or minimum energy configuration of an ideal elastic beam constrained to pass through the given data points. The approximation results from neglecting the nonlinearities in the beam equation. Thus it can be shown that for a set of N data points the spline curve consists of $N-1$ piecewise cubic polynomials which have continuous first and second derivatives at the $N-2$ interior juncture points [3]. The method to be described here gives a computational scheme for finding the equilibrium position of the beam that fully accounts for the nonlinearities in the beam equation. The method makes use of a minimization procedure

that permits computation of the minimum energy configuration to as fine a precision as desired, within the limits of computational roundoff errors.

An important advantage of the present method is that it is not restricted to single-valued data points as is the case with the spline technique. The method described here permits arbitrarily located data points (some exceptions will be noted later) in the $x-y$ plane and seeks to pass the smoothest curve through these points in the minimum energy sense. Our definition of the "energy" of a curve is directly analogous to the formulation for the strain energy of an elastic beam under flexure. That is, for a curve $y(x)$ with arc length L, we have

$$E[y(x)]_0^L = \int_0^L \psi^2(s)\,ds\,, \tag{1}$$

where $\psi(s)$ is the reciprocal of the instantaneous radius of curvature of the curve. Equation (1) is identically the expression for the strain energy of a beam, neglecting the appropriate material and geometry constants. It should be noted that the energy interpretation requires that the radius of curvature be everywhere nonzero; otherwise an infinite energy will result. Throughout the rest of this paper the *minimum-energy curve* for a set of data points will be abbreviated as MEC. The next section derives the differential equation satisfied by the MEC between each pair of data points. Section III shows the numerical solution of the differential equation and a gradient minimization method for adjusting the boundary conditions. A brief discussion of the design of the minimization algorithm for computer implementation is given in Section IV, and results and conclusions are presented in Sections V and VI, respectively.

II. Equation of the MEC.

Let the data points (boundary points) be represented by their $x-y$ coordinates, (x_i, y_i), $i = 1, 2, \ldots, N$, and consider a beam (curve) passing through all the data points: i.e., $y(x_i) = y_i$, (see Fig. 1). Letting W denote the strain-energy density, and S_i the arc length along $y(x)$ up to the i^{th} data point, the total strain energy of the curve is written as

$$E[y] = \sum_{i=1}^{N-1} \int_{S_i}^{S_{i+1}} W\,ds\,. \tag{2}$$

From (1), W is recognized as the square of the reciprocal of the instantaneous radius of curvature, which in cartesian coordinates is

$$W = \frac{(y'')^2}{[1+(y')^2]^3}. \tag{3}$$

Substituting (3) in (2), and taking

$$ds = [1+(y')^2]^{\frac{1}{2}} dx ,$$

(2) becomes

$$E[y] = \sum_{i=1}^{N-1} \int_{x_i}^{x_{i+1}} \frac{(y'')^2}{[1+(y')^2]^{5/2}} dx . \tag{4}$$

The MEC is then that function $y(x)$, with $y(x_i)=y_i$, that minimizes (4). At all interior boundary points, $i=2,3,\ldots,N-1$, the first derivative must be continuous; otherwise infinite strain energy will result, an impossibility for a physical beam.

Between each pair of boundary points the MEC must satisfy the differential equation that minimizes the integral in (4). This differential equation is obtained by using the calculus of variations and is found to be [7, 8]

$$y^{\mathrm{iv}} = \frac{5(y'')^3 + 20y'y''y'''}{2[1+(y')^2]} - \frac{35(y')^2(y'')^3}{2[1+(y')^2]^2}, \tag{5}$$

which is recognized as a fourth-order nonlinear differential equation. It can be further shown that for a meaningful solution to exist the four boundary conditions that must be specified are: $y(x_i)$, $y(x_{i+1})$, $y'(x_i)$ and $y'(x_{i+1})$ [7]. The minimization of (4) is thus recognized as a multipoint boundary-value problem. When the boundary coordinates are fixed, then only the boundary slopes, $y'(x_i)$, must be determined. Thus the minimization involves a search for the $y'(x_i)$, $i=1,2,\ldots,N$, that minimize (4), where $y(x)$ satisfies (5) and $y(x_i)=y_i$, $i=1,2,\ldots,N$. Note that the solution must provide for continuity in function value and slope across each of the boundary points in order to faithfully represent the continuous beam. If all the boundary points were to lie along a straight line then the minimizing solution would itself be a straight line. When the boundary points are located such that the change in slope of the solution within each interval is small, then the denominator in (4) can be accurately approximated by a constant in each interval and the minimization yields the spline interpolation.

It has not been found possible to integrate (5) analytically, and hence a numerical solution is indicated. The next section describes the numerical solution of (5) subject to the four boundary conditions specified above, and presents the gradient method for adjusting the slopes at the boundary points so as to minimize (4).

III. Numerical solution of the minimum-energy curve.

3.1. *Solution of the Differential Equation.*

A numerical solution of (5) will be given for the interval (0,0)–(1,0), subject to the given boundary conditions $y'(0)$ and $y'(1)$. Figure 2 shows the arrangement to be used for evaluating quantities at discrete points in the range, where the x-interval $[0,1]$ is subdivided at N points, including the points $x=0$ and $x=1$. The notation used is largely that due to Fox [9]. The points of subdivision are called "pivotal points" and the solution values at these points are called "pivotal values". The pivotal points are indexed from $j=1$ to $j=N$ and the pivotal values denoted by y_j. The uniform mesh spacing will be represented by $H=x_{j+1}-x_j$.

There are two basic methods for handling the solution of a boundary-value problem: 1) converting the boundary-value problem into an initial-value problem; 2) direct solution of the difference-equation representation of the differential equation by forcing it to satisfy the boundary conditions. The present work has followed the second approach.

The second method involves writing the finite-difference equation at each of the $N-2$ interior pivotal points in terms of the unknown pivotal values, y_j. One thus obtains $N-2$ algebraic equations in the $N-2$ interior pivotal values and those exterior pivotal values which are involved when the difference equation is written at pivotal points near the end of the range. The boundary conditions must be sufficient in number and kind to supply the remaining equations needed to obtain a consistent set. The y_j can then be solved for, either directly, as when the difference equation is linear, or by iteration, for a nonlinear difference equation. Alternatively, the original differential equation can first be linearized and an iterative approach used to converge to the solution. The latter method is used in this paper.

The linearization of (5) is accomplished as follows. Suppose that $y^{(0)}$ is an approximate solution of (5) and that a better approximation is given by $y^{(0)}+\eta^{(0)}$. Let (5) be written in abbreviated form as $y^{\mathrm{iv}}=f(x,y',y'',y''')$. Expanding (5) about $y^{(0)}$ in a Taylor series and retaining only the first-order terms, we get

$$(y^{(0)}+\eta^{(0)})^{\mathrm{iv}} = f(x,y^{(0)\prime},y^{(0)\prime\prime},y^{(0)\prime\prime\prime})+\eta^{(0)\prime}\frac{\partial f}{\partial y^{(0)\prime}}+\eta^{(0)\prime\prime}\frac{\partial f}{\partial y^{(0)\prime\prime}}+\eta^{(0)\prime\prime\prime}\frac{\partial f}{\partial y^{(0)\prime\prime\prime}}. \quad (6)$$

Rewriting (6), dropping the superscript notation for convenience, and letting

$$(y^{(0)})^{\mathrm{iv}}-f(x,y^{(0)\prime},y^{(0)\prime\prime},y^{(0)\prime\prime\prime}) = b\,,$$

one obtains

$$-\eta^{iv}+\eta'''\frac{\partial f}{\partial y'''}+\eta''\frac{\partial f}{\partial y''}+\eta'\frac{\partial f}{\partial y'}=y^{iv}-f(x,y',y'',y''')=b \qquad (7)$$

Now (7) represents a linear equation in η, and solving one next takes $y^{(0)}+\eta^{(0)}\rightarrow y^{(1)}$ as a better approximation. The procedure can then be repeated using the improved approximation $y^{(1)}$ in (7). Succeeding iterates, $y^{(i)}$, should converge to the solution if the approximate solution at any stage in the iteration is sufficiently close to the true solution so that the neglected higher-order terms in (6) are negligible. The process terminates when the i^{th} correction, $\eta^{(i)}$, becomes negligibly small, as determined by some desired tolerance. At this stage the mesh size, H, can be halved, the calculated pivotal values used to interpolate for the intermediate values, and the iterations repeated. The mesh size can be halved as many times as is required in order to obtain a desired precision for the solution.

The conversion of the foregoing procedure into a finite-difference scheme is shown in Appendix A. It is shown there that one can arrive at the following difference equation version of (7), namely,

$$A_{2j}\eta_{j+2}+A_{1j}\eta_{j+1}+A_{0j}\eta_j+A_{-1j}\eta_{j-1}+A_{-2j}\eta_{j-2}=B_j \qquad j=2,3,\ldots,N-1\,. \qquad (8)$$

One observes from (8) that the equations written at the pivotal points $j=2$ and $j=N-1$ will each involve a pivotal point exterior to the range. Auxiliary relations are needed to express the two corresponding exterior pivotal values in terms of interior values and the boundary conditions are used for this purpose. For example, writing a Taylor series expansion at $j=N$ about y_N, one obtains

$$y_{N+1}=y_N+Hy_N'+\frac{H^2}{2!}y_N''+\ldots\,. \qquad (9)$$

Both y_N and y_N' are given quantities at each stage of the iteration, being supplied from the given boundary conditions. If H is sufficiently small, all terms beyond first-order will be negligible and (9) can be written as

$$y_{N+1}\cong y_N+Hy_N'\,. \qquad (10)$$

The value of η at the external pivotal points can be similarly evaluated. Thus

$$\eta_{N+1}=\eta_N+H\eta_N'+\frac{H^2}{2!}\eta_N''+\ldots\,. \qquad (11)$$

Since η_N and η_N' must vanish because of the boundary conditions fixing

y_N and y_N', and neglecting the higher-order terms for the same reason as in (9)–(11), there results

$$\eta_{N+1} \cong 0 . \tag{12}$$

With successive iterations of (8) the η_j will become progressively smaller if the process is converging, and the approximation (12) will improve. Expressions identical to (10) and (12) are obtained at the other end of the range, $j = 1$.

Thus the system of equations (8) will consist of $N-2$ linear equations in the $N-2$ unknowns, η_j, $j = 2, 3, \ldots, N-1$. In matrix notation (8) can be written

$$\boldsymbol{A}\eta = \boldsymbol{B} \tag{13}$$

$\boldsymbol{A}$ being a sparse matrix, and (13) is rapidly solved on a digital computer by straightforward Gauss elimination.

Successive iterations with (13) will give progressively decreasing values of η_j provided that at any stage the iterate values, y_j, are sufficiently close to the true solution so that the neglected higher-order terms in the expansion (6) are indeed negligible compared to the terms retained. Thus convergence is strongly dependent on the initial approximation being a good one. After each iteration for η_j, the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are recomputed using the corrected pivotal values $y_j + \eta_j \to y_j$, the iterations being continued until the η_j have been reduced below a desired tolerance. The final pivotal values are then used to compute the energy via a numerical integration of (4), provided the mesh size used (H) is sufficiently small to give a desired precision. Because of the smooth nature of the solution, a zero-order integration formula was found adequate for a mesh size slightly less than $\frac{1}{100}$ of the interval range. The observed difference in the computed energy values using higher-order formulas was confined to at most the fourth decimal place.

The initial approximation ($y^{(0)}$) for the pivotal values is provided by a cubic polynomial, determined by the four boundary conditions. Justification for use of a cubic rests on the fact that the solution itself will behave as a cubic over small intervals and also over larger intervals if the slope change between boundary points is small (i.e., spline interpolation). With the cubic approximation convergence was always obtained provided the specified boundary slopes did not give too "severe" a slope change per unit range distance or provided there was not a severely asymmetrical slope difference. In such cases the cubic approximation is not accurate enough to permit neglecting the higher-order terms in the expansion of (5) and the solution becomes unstable. A way of frequently overcoming this difficulty was found through the use of an "intermediate" variable

boundary point, whose location can be adjusted so as to minimize the energy of the curve passing through the given boundary points and the intermediate point. If the solution is still unstable then additional intermediate points must be added. The computation time, of course, increases appreciably. Thus the data points can be arbitrarily located (see Section I) as long as the solution remains stable.

3.2. *The Gradient Method for Minimization.*

Having obtained a solution for the energy in any interval, we can proceed to show the gradient method for adjusting the boundary-values slopes in order to give a minimum-energy curve through a set of data points (see Fig. 1). The problem can be considered as one in which a

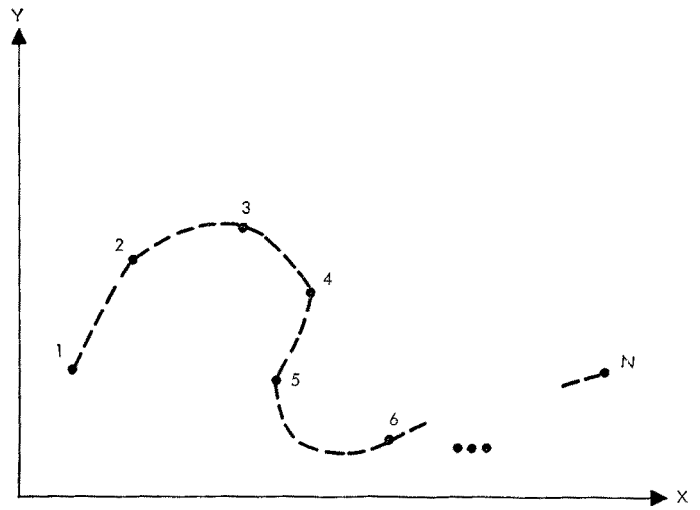

Figure 1. Minimum-energy curve through a set of N data points.

criterion function, the total strain energy, is expressed as a function of the N boundary slopes, or

$$E[y] = f[y_1', y_2', \ldots, y_N'] , \tag{14}$$

the boundary positions being fixed.

We let the n-tuple

$$\bar{S}_k = [y'_{1k}, y'_{2k}, \ldots, y'_{Nk}]$$

represent the slopes at each of the N boundary points at the k^{th} iteration of the minimization procedure, and we take E_k as the corresponding energy of the solution. The gradient method results in a sequence of slope vectors, $\bar{S}_1, \bar{S}_2, \ldots, \bar{S}_k, \bar{S}_{k+1}, \ldots, \bar{S}_l$, such that $E_1 > E_2 > \ldots > E_k > E_{k+1} >$

$\ldots > E_t$, where $\bar{S}_t$ corresponds to the slope vector of the minimum-energy solution. At the minimum, $\bar{S}_t$, the gradient of the energy with respect to the slope vector must vanish, or

$$\operatorname{grad} E_t = \left[\frac{\partial E_t}{\partial y'_{1t}}, \frac{\partial E_t}{\partial y'_{2t}}, \ldots, \frac{\partial E_t}{\partial y'_{Nt}}\right] \equiv 0 . \tag{15}$$

One thus obtains the slope vector for the $k+1^{st}$ iteration as

$$\bar{S}_{k+1} = \bar{S}_k - \lambda_k \operatorname{grad} E_k , \tag{16}$$

where λ_k is a constant adjusted to ensure that $E_{k+1} < E_k$, and the minus sign is required because the gradient vector is in a direction to maximize the criterion function. Use of (16) is known as the method of steepest descent [10].

Because an analytic expression for the energy in each interval is not available, the partial derivative components of the gradient vector at the k^{th} iteration must be approximated by terms

$$\frac{\Delta E_k}{\Delta y'_{ik}}, \qquad i = 1, 2, \ldots, N .$$

One notes that the gradient components can be evaluated by making a small change in slope at each boundary point and evaluating the resulting energy change in the two adjacent intervals (or a single interval in the case of the end points). In the actual computation the gradient vector itself was not used in the iterations of (16). Rather, the gradient components were quantized into $+1$, -1, or 0, according as the components increase, decrease, or do not change the energy beyond the tolerance set for the gradient. Thus all boundary slopes undergo the same magnitude of change, or none at all, for each iteration of (16). The reason for this procedure was to avoid instability which may result from the large value of some of the gradient components and the accompanying large boundary slope change in the corresponding intervals. Although it seems as if the method used would be highly inefficient, the comparatively few iterations required in most of the examples performed testifies to its efficiency.

The efficiency mentioned in the preceding paragraph is due in large measure to the good initial approximation $\bar{S}_1$ that can be obtained to the solution $\bar{S}_t$. It will be recalled from Section 3.1 that convergence is critically dependent on the approximate solution at any point being suitably close to the true solution to permit the linearized expansion of (5). It has been found that the slopes obtained by passing "overlap-

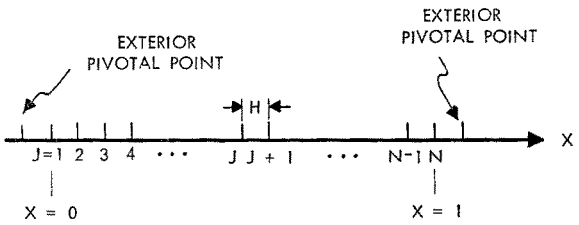

Figure 2. Arrangement for the location of the pivotal points for the difference-equation solution. The uniform mesh size is H.

ping" parabolas through the data points give a fairly accurate initial approximation. Fig. 3 illustrates the idea. The initial slope at the i^{th} data point is taken as the slope of the parabola passing through the points $i-1$, i, and $i+1$. It has been observed that in those intervals for which the overlapping parabolas have second derivatives of opposite sign, the MEC will have a point of inflection.

Comparison of the final slope values achieved in the minimization with the initial approximation reveals good agreement, as may be seen from an example included in Section V. Physical intuition strongly suggests that the problem has an isolated minimum, provided the initial approximation is not too far removed from the stationary point. This intuition has been confirmed computationally with the examples tested as well as with a physical simulator [7, 8]. However, no uniqueness proof has yet been obtained for the MEC.

The use of any numerical gradient technique always evokes consideration of the convergence problem; that is, finding an efficient choice for the λ_k in (16) and use of a suitable metric (e.g., Newton's method) to give

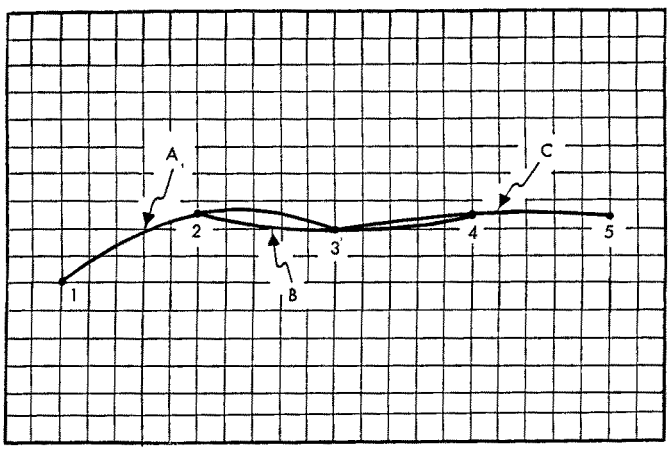

Figure 3. Parabola fitting for determining the initial slope approximation.

quadratic convergence near the minimum. Methods for giving improved convergence near the minimum were not considered in this work; nor was any attempt made to design a sophisticated searching algorithm for the optimum λ_k. Rather, the value of λ_1 at the first iteration was successively increased until the greatest possible energy reduction was achieved; thereafter the iterations used $\lambda_{k+1}=\lambda_k$, where the value was repeatedly halved until the energy relation $E_{k+1}<E_k$ was satisfied. This procedure seemed to work quite well in most cases, giving satisfactory convergence (less than 1 percent change in energy) within four or five iterations. The bulk of the computation time consists of evaluating the gradient and the criterion function via solution of the basic difference equation (13) in each interval. Thus it was felt that computation time per iteration would be substantially increased* by use of accelerating methods and by use of any of a variety of schemes for interpolating for the optimum λ_k in the direction $-\text{grad}\,E_k$. The author suggests, however, that there is considerable opportunity for the application of acceleration techniques to give improved convergence of the minimum-energy solution.

In each interval the solution is carried out by rotating coordinates so as to bring the x-axis into correspondence with the line joining the two data points. This procedure allows the number of pivotal points in each interval to be made directly proportional to the interval length (the distance between two data points along the line joining them), thus ensuring that a nearly uniform precision is obtained throughout the solution. In the interval $(x_i, y_i)-(x_{i+1}, y_{i+1})$ the angle of rotation is

$$\theta_i = \tan^{-1}\frac{y_{i+1}-y_i}{x_{i+1}-x_i}. \tag{17}$$

Hence if θ_{ik} is the trial solution angle corresponding to the i^{th} data point at the k^{th} iteration of (16), the boundary slopes used in the numerical solution of (5) are given by

$$\begin{aligned} y'_{ik} &= \frac{\tan\theta_{ik}-\tan\theta_i}{1+(\tan\theta_{ik})(\tan\theta_i)} \\ y'_{i+1,k} &= \frac{\tan\theta_{i+1,k}-\tan\theta_i}{1+(\tan\theta_{i+1,k})(\tan\theta_i)}. \end{aligned} \tag{18}$$

When minimization is reached the coordinates of the pivotal points in

* The *iteration* time per *interval* averaged approximately 6 seconds for the examples chosen. The time, of course, is strongly dependent on the interval size and the fineness of the subdivisions used. The computations were performed on an IBM 7094.

each interval can be transformed back into the absolute coordinates to facilitate graphing of the solution.

IV. Selection of parameters for the MEC algorithm.

There are three competing factors influencing the choice of the various parameters to be supplied to the MEC algorithm: 1) we desire as precise a solution as possible; 2) we desire as rapid convergence as possible; 3) we must guard against instability of the solution. The last-mentioned factor is the most important and decisions affecting stability must override considerations of precision and convergence.

The tolerance values used in the algorithm were found largely through experience gained from worked examples and the intuition acquired therein. Setting a tolerance for determining when a stationary point is reached depends on several factors: 1) the incremental value of slope change used to calculate the gradient components; 2) the number of iterations of (16) required to give a value E_{k+1} whose j^{th} decimal place remains unchanged from the E_k value; 3) the tolerance value used to terminate the η correction-factor iteration. All these factors are in turn influenced by the mesh size H and the roundoff error propagated through the entire solution. It was found from worked examples that when the energy change due to an incremental slope change at any data point was approximately 0.2 percent of the total curve energy, then succeeding iterations of (16) did not reduce the total energy by more than approximately 0.5 percent. For the examples tested the curve energies were all within an order of magnitude of each other, ranging from about 2 to 20, and a value of 0.001 for the gradient energy tolerance was sufficiently small to generally give 1 percent accuracy for the minimum.

The incremental slope change used to calculate the gradient components was usually set at 0.005. This value was found to be sufficiently large to give a measurable change even near a stationary point and sufficiently small to approximate the taking of a derivative. It was also large enough to prevent the gradient tolerance from being satisfied before the solution had come satisfactorily close to the stationary point. The average number of iterations of (16) required to satisfy the gradient tolerance averaged about five for the examples tested. Reduction of the gradient tolerance resulted in an appreciable increase in the number of iterations. The initial trial value of λ, i.e., λ_1, was taken to be four times the incremental slope value. Successive doubling of λ_1 was permitted to obtain that value resulting in the largest energy reduction. If the initial

trial value failed to reduce the energy, then λ_1 was successively halved until reduction occurred.

The range between the data points influences the selection of an appropriate tolerance for the η correction factors. For all test examples the range was within two units and an acceptable tolerance was found to be 0.00002. When the tolerance was reduced to 0.000002 the resulting energy values differed only in the fifth decimal place from those obtained with the larger value. Reducing the tolerance by still another order of magnitude failed to produce convergence, thus establishing the effect of round-off-error propagation in the calculation and indicating that results computed with the 0.00002 value were sufficiently protected from roundoff.

Selection of the mesh size, H, can critically affect the precision of the energy calculation. In most of the test examples the initial value of H was chosen to give about 60 pivotal points per unit distance in each data interval. Generally one interval subdivision was used, resulting in about 120 pivotal points per unit distance for the final energy calculation. Such a mesh size appears to be sufficient for maintaining the precision established by the gradient and η tolerances. Any finer mesh affects the energy calculations only in the fourth decimal place or beyond. If the mesh is made too fine it can have an adverse effect by tending to amplify at any stage in the iterations the difference between the current calculated pivotal values and the true values. The reason for this effect can be seen by considering the expressions obtained for the a_{1j}, a_{2j}, and a_{3j} terms as defined by (A-1). Carrying out the operations indicated in (A-1) shows that H appears raised to the sixth and eighth power in the denominator of some terms. Where these are not effectively cancelled by corresponding multiplying powers of H the result can be an oscillation in the solution, or even instability.

V. Results.

Several comparisons of MEC and spline interpolation are shown in Fig. 4 to 7, with the respective curve energies listed. One notes in Fig. 5

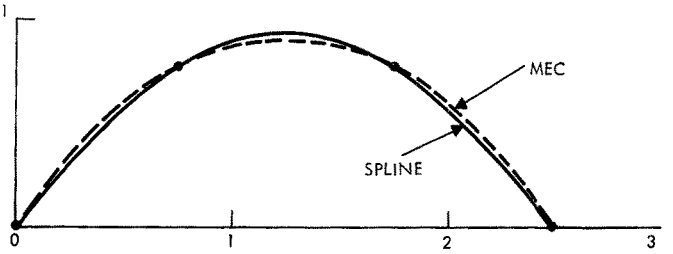

Figure 4. Example 1. Spline energy = 2.249. Mec energy = 1.383.

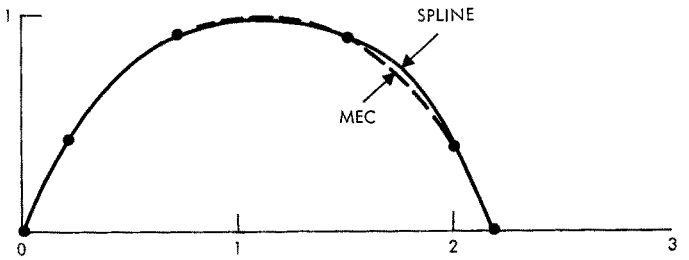

Figure 5. Example 2. Spline energy = 18.63. Mec energy = 2.005.

and 7 that the spline method does not give a completely symmetrical curve even though the samples (shown in heavy black dots) are symmetrically placed*. In all cases the spline curve energy is substantially greater than that of the corresponding MEC.

To give an idea of the accuracy of the initial slope approximation provided by fitting overlapping parabolas (see Fig. 3) to the samples consider Table I. This table lists the initial and final values of the boundary slopes in each of the nine intervals for the example of Fig. 7. The slopes are given with respect to the straight line joining the two boundary points of each interval. The sequence of energy values obtained in the minimization is listed in Table II. At the end of the tenth iteration all

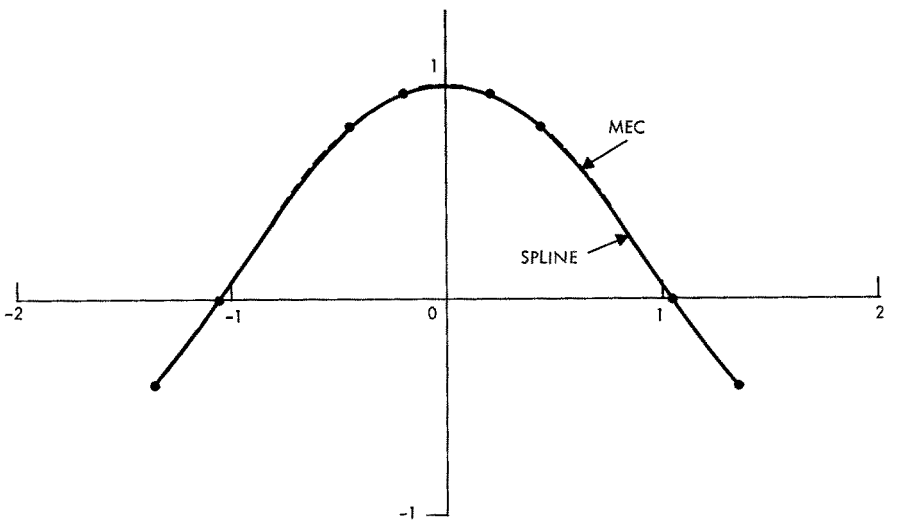

Figure 6. Example 3. Spline energy = 5.501. Mec energy = 2.305.

* There is no special significance to the use of symmetric samples. The data used here were obtained in connection with the study of a related quantization problem [7, 8] that resulted in symmetric data.

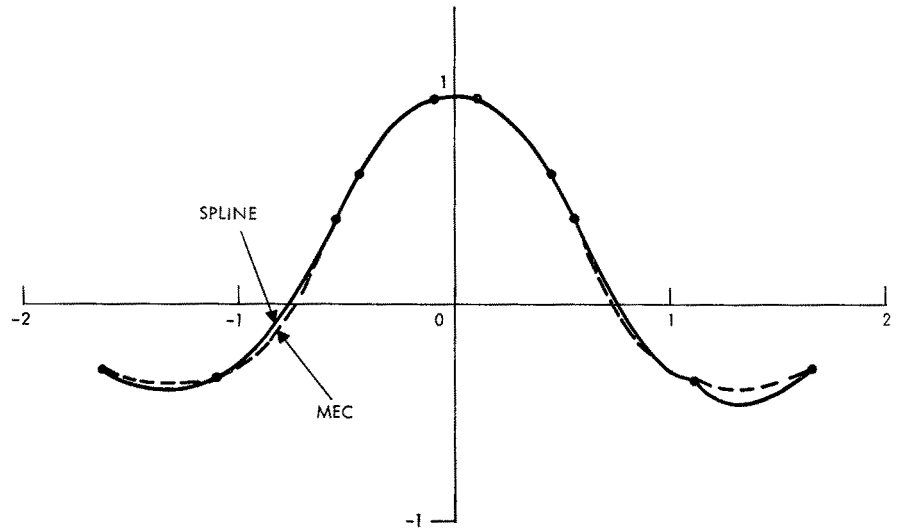

Figure 7. Example 4. Spline energy = 37.13. Mec energy = 8.353.

components of the gradient vector were below 0.001 except for the component associated with the sample (0.1, 0.95), which was 0.001295. The total computation time for this example was approximately 10 minutes on the IBM 7094.

Table I

Comparison of initial and final slopes for the example of fig. 7.

Interval	*Initial Slopes*		*Final Slopes*	
1	−0.676,	0.771	−0.201,	0.419
2	−0.389,	0.149	−0.731,	0.190
3	−0.020,	−0.048	0.019,	−0.030
4	0.280,	−0.466	0.300,	−0.558
5	0.363,	−0.363	0.283,	−0.288
6	0.466,	−0.280	0.552,	−0.300
7	0.048,	0.020	0.030,	−0.026
8	−0.149,	0.389	−0.198,	0.731
9	−0.771,	0.676	−0.419,	0.201

VI. Conclusions.

A method has been presented for obtaining the smoothest curve through a set of data points in the sense of minimizing the strain energy of an ideal elastic beam constrained to pass through the points. The method represents an advance over the usual spline interpolation procedure in that it fully accounts for the nonlinearities in the beam equation

Table II

Sequence of MEC energy values obtained for the example of fig. 7.

Iteration	*Energy*
1	10.465
2	9.552
3	8.957
4	8.857
5	8.756
6	8.489
7	8.483
8	8.392
9	8.354
10	8.353

and is not restricted to single-valued data points. The drawback of the method is the enormous computation time, compared to the spline fit, as well as possible stability problems when the data are such as to result in large slope change per unit distance of the solution. The stability problem can always be overcome by the addition of sufficient numbers of adjustable, intermediate boundary points, but at the expense of a substantial increase in computation time.

Acknowledgment.

The author wishes to thank Professor H. Freeman, Department of Electrical Engineering, New York University, for his helpful suggestions. The support of the Air Force Office of Scientific Research, Information Sciences Directorate, under Grants AF-AFOSR-24-64 and AF-AFOSR-24-65, is gratefully acknowledged.

APPENDIX A
FINITE-DIFFERENCE FORMULAE FOR NUMERICAL SOLUTION OF THE MINIMIZING DIFFERENTIAL EQUATION

We adopt the following notation:

$$y_j^{\mathrm{iv}} - f(x_j, y_j', y_j'', y_j''') = b_j$$

$$\frac{\partial f}{\partial y_j'} = a_{1j}; \quad \frac{\partial f}{\partial y_j''} = a_{2j}; \quad \frac{\partial f}{\partial y_j'''} = a_{3j} \tag{A-1}$$

$$\eta_j' \to \eta'; \ \eta_j'' \to \eta''; \ \eta_j''' \to \eta'''; \ \eta_j^{\mathrm{iv}} \to \eta^{\mathrm{iv}}, \quad j = 2, 3, \ldots, N-1 .$$

Equation (7) in the text can then be written

$$-\eta_j^{\mathrm{iv}} + a_{1j}\eta_j' + a_{2j}\eta_j'' + a_{3j}\eta_j''' = b_j, \quad j = 2,3,\ldots,N-1 . \quad \text{(A-2)}$$

Defining the "central-difference" and "averaging" operators respectively by

$$\delta\eta_j \triangleq (\eta_{j+\frac{1}{2}} - \eta_{j-\frac{1}{2}})/H$$
$$\mu\eta_j \triangleq (\eta_{j+\frac{1}{2}} + \eta_{j-\frac{1}{2}})/2 ,$$

the following "balanced" finite difference formulae are obtained:

$$\begin{aligned}
\eta_j' \triangleq \mu\delta\eta_j &= (\eta_{j+1} - \eta_{j-1})/H \\
\eta_j'' \triangleq \delta^2\eta_j &= (\eta_{j+1} - 2\eta_j + \eta_{j-1})/H^2 \\
\eta_j''' \triangleq \mu\delta^3\eta_j &= (\eta_{j+2} - 2\eta_{j+1} + 2\eta_{j-1} - \eta_{j-2})/2H^3 \\
\eta_j^{\mathrm{iv}} \triangleq \delta\eta_j^4 &= (\eta_{j+2} - 4\eta_{j+1} + 6\eta_j - 4\eta_{j-1} + \eta_{j-2})/H^4 . \qquad \text{(A-3)}
\end{aligned}$$

Substituting from (A-3) into (A-2), and collecting like terms we get:

$$\begin{aligned}
&\eta_{j+2}(-1 + a_{3j}H/2) + \eta_{j+1}(4 + a_{1j}H^3/2 + a_{2j}H^2 - a_{3j}H) \\
&\quad + \eta_j(-6 - 2a_{2j}H^2) + \eta_{j-1}(4 - a_{1j}H^3/2 + a_{2j}H^2 + a_{3j}H) \\
&\quad + \eta_{j-2}(-1 - a_{3j}H/2) = H^4 b_j = B_j, \quad j = 2,3,\ldots,N-1 . \quad \text{(A-4)}
\end{aligned}$$

The terms a_{1j}, a_{2j} and a_{3j} are readily obtained using (5) and (A-1), and finite-difference formulae as in (A-3) to replace the y_j', y_j'' and y_j'''. The expressions for the a_{1j}, a_{2j} and a_{3j} are then substituted into (A-4) to obtain the system of algebraic equations in the unknowns η_j, which can be written compactly as

$$A_{2j}\eta_{j+2} + A_{1j}\eta_{j+1} + A_{0j}\eta_j + A_{-1j}\eta_{j-1} + A_{-2j}\eta_{j-2} = B_j ,$$
$$j = 2,3,\ldots,N-1 \qquad \text{(A-5)}$$

(A-5) is recognized as (8) in the text.

REFERENCES

1. Theilheimer, F. and W. Starkweather, *The Fairing of Ship Lines on a High-Speed Computer*, Math. Comp., 15, pp. 338–355, 1961.
2. Asker, B., *The Spline Curve, A Smooth Interpolation Function Used in Numerical Design of Ship-Lines*, BIT 2, pp. 76–82, 1962.
3. Walsh, J. L., Ahlberg, J. H., and Nilson, E. N., *Best Approximation Properties of the Spline Fit*, J. Math. Mech. 11, pp. 225–234, 1962.
4. Greville, T. N. E., *Numerical Procedures for Interpolation by Spline Functions*, Mathematics Research Center, University of Wisconsin, Technical Summary Report No. 450, January 1964 (Madison, Wis.).
5. Mehlum, E., *A Curve-Fitting Method Based on a Variational Criterion*, BIT 4, pp. 213–223, 1964.

6. Holladay, J. C., *A Smoothest Curve Approximation*, Math. Comp. 11, pp. 233–243, 1957.
7. Glass, J. M., *A Criterion for the Quantization of Line-Drawing Data*, Technical Report 400-112, Laboratory for Electroscience Research, Department of Electrical Engineering, New York University, New York 10453, N. Y., 1965.
8. Freeman, H. and Glass J. M., *On the Quantization of Line-Drawing Data*, to be published.
9. Fox, L., *The Numerical Solution of Two-Point Boundary Problems in Ordinary Differential Equations*, Oxford University Press, London, 1957.
10. Curry, H. B., *The Method of Steepest Descent for Nonlinear Minimization Problems*, Q. of Appl. Math. 2, pp. 258–261, 1944.

APPLIED RESEARCH LABORATORY
SYLVANIA ELECTRONIC SYSTEMS
WALTHAM, MASSACHUSETTS