Applications of Exponential Splines in Computational Fluid Dynamics

B. J. McCartin*
Pratt & Whitney Aircraft, East Hartford, Connecticut

Recurrent problems in the use of cubic spline interpolation are investigated. It is shown how to automatically resolve any difficulties that do arise, by the alternative use of exponential spline interpolation. Numerical examples of a general nature are presented first illustrating the inherent superiority of the exponential spline to the cubic spline. Attention is then focused on a variety of problems from computational fluid dynamics. The discussion culminates in the detailed exposition of a typical boundary-layer calculation.

Nomenclature

a $= x_1 < \ldots < x_{N+1} = b$, spline nodes
b_i $= (f_{i+1} - f_i)/h_i - (f_i - f_{i-1})/h_{i-1}$, $(i = 2,\ldots,N)$
b_1 $= (f_2 - f_1)/h_1 - f'(a)$
b_{N+1} $= f'(b) - (f_{N+1} - f_N)/h_N$
C_i $= \cosh(p_i h_i)$, $(i = 1,\ldots,N)$
d_i $= (p_i C_i/S_i - 1/h_i)/p_i^2$, $(i = 1,\ldots,N)$
D = differentiation
e_i $= (1/h_i - p_i/S_i)/p_i^2$, $(i = 1,\ldots,N)$
f = data
h_i = length of ith spline interval, $(i = 1,\ldots,N)$
N = number of spline intervals
p_i = tension parameter on ith spline interval, $(i = 1,\ldots,N)$
s = cubic spline interpolant
S_i $= \sinh(p_i h_i)$, $(i = 1,\ldots,N)$
τ = exponential spline interpolant
ω = relaxation parameter

Introduction

As if the attendant mathematical difficulties were not inherently severe enough, the calculation of fluid flowfields invariably requires, at one stage or another, the construction of smooth curves from a prescribed set of tabulated data points. Examples that immediately come to mind are the definition of body geometry, the numerical enforcement of boundary conditions, and the generation of a computational mesh. Hence, reliable interpolation procedures must be developed. In what follows, we restrict our attention to the two-dimensional case for ease of exposition. The techniques developed have no such intrinsic limitation.

Classically (dating back at least to Newton and Lagrange), polynomial interpolation was the prevalent method of constructing smooth interpolants to discrete data.[1] In fact, polynomials are so smooth that strictly local behavior cannot be controlled without affecting the curve globally. To be sure, anomalies such as Runge's phenomenon are additional drawbacks.[2]

Hence, it was a major step forward when cubic splines were developed.[3] At the sacrifice of smoothness, they afford a high degree of local control. However, many times "wiggles" are present in the cubic spline interpolant. The reason for this is heuristically motivated as follows. The cubic spline interpolant with appropriate end conditions minimizes the functional

$$\int_\alpha^\beta [\psi''(x)]^2 dx$$

where $[\alpha,\beta]$ is the interval of interpolation (closely related to the potential energy of a statically deflected beam forced to pass through the data points). Therefore, the cubic spline will behave however it must to diminish this integral, including inserting inflection points whether or not the data points reflect such changes in curvature. Such a spurious oscillation is known in the literature as an extraneous inflection point (EIP).[4]

Späth[5] first proposed using exponential spline interpolants as a means of overcoming this difficulty. The exponential spline can be shown to be free of EIP's for appropriate values of certain free parameters in the interpolant.[6] That is, if the data are locally convex, then the interpolant likewise will be locally convex. Once these "tension" parameters are specified, the interpolant is uniquely defined. These parameters appear nonlinearly, thus requiring an iterative procedure for their determination.

In the following we review the results on exponential splines that are germane to our discussion. This is followed by a sequence of applications to computational fluid dynamics.

Theory

Recall that a cubic spline is a cubic polynomial on each subinterval $[x_i,x_{i+1}]$ $(i = 1,\ldots,N)$. These cubics are joined at the knots in such a fashion that the resulting interpolant has global C^2 continuity. Note that we must determine the coefficients of each cubic in order to specify a unique cubic spline. This amounts to $4N$ parameters to be determined. The interpolatory constraints number $N+1$ while the continuity constraints on $s(x)$, $s'(x)$ and $s''(x)$ number $3(N-1)$. Thus, we have a total of $4N-2$ constraints and $4N$ parameters to determine. We clearly need to impose two additional constraints for uniqueness. While there is a multitude of ways to accomplish this we assume that we know the exact end conditions $s'(a) = f'(a)$ and $s'(b) = f'(b)$.

An alternative view of this definition is as follows. We require that the cubic spline be the solution to the boundary-value problem on $[x_i,x_{i+1}]$ $(i = 1,\ldots,N)$

$$[D^4]s = 0, \quad s(x_i) = f_i, \quad s(x_{i+1}) = f_{i+1}$$

$$s''(x_i) = s_i'', \quad s''(x_{i+1}) = s_{i+1}''$$

Presented as Paper 81-0995 at the AIAA Fifth Computational Fluid Dynamics Conference, Palo Alto, Calif., June 22-23, 1981; submitted June 24, 1981; revision received Oct. 25, 1982.

*Research Scientist; at present, Research Mathematician, United Technologies Research Center, East Hartford, Conn.

Where s_i'' and s_{i+1}'' are chosen to ensure that $s(x) \in C^2[a,b]$ when $s'(a)$ and $s'(b)$ are specified. This results in

$$S(X) = f_i\left[\frac{X_{i+1}-X}{h_i}\right] + f_{i+1}\left[\frac{X-X_i}{h_i}\right]$$
$$- \frac{1}{6}h_i(X_{i+1}-X)S_i''\left[1-\frac{(X_{i+1}-X)^2}{h_i^2}\right]$$
$$- \frac{1}{6}h_i(X-X_i)S_{i+1}''\left[1-\frac{(X-X_i)^2}{h_i^2}\right]$$

S_i'' $(i=1,\ldots,N+1)$ are determined from the tridiagonal system

$$\frac{h_1}{3}S_1'' + \frac{h_1}{6}S_2'' = b_1$$
$$\frac{h_{i-1}}{6}S_{i-1}'' + \frac{h_{i-1}+h_i}{3}S_i'' + \frac{h_i}{6}S_{i+1}'' = b_i \quad (i=2,\ldots,N)$$
$$\frac{h_N}{6}S_N'' + \frac{h_N}{3}S_{N+1}'' = b_{N+1}$$

Note that this boundary-value problem is that of a simply supported, linear, elastic beam.[8]

In practice the cubic spline exhibits unwanted undulations. Specifically, we call an inflection point of $s(x)$ in $[x_i,x_{i+1}]$ extraneous if the second central differences of the data points at x_i and x_{i+1} are of the same sign, i.e., $b_ib_{i+1}>0$. When such an oscillation does appear we attempt to remedy the situation by adding uniform tension to the beam analogy.

This leads us to define the exponential spline as the solution to the boundary-value problem on $[x_i,x_{i+1}]$ $(i=1,\ldots,N)$.

$$[D^4 - p_i^2D^2]\tau = 0, \quad \tau(X_i)=f_i, \quad \tau(X_{i+1})=f_{i+1}$$
$$\tau''(X_i)=\tau_i'', \quad \tau''(X_{i+1})=\tau_{i+1}''$$

where τ_i'' and τ_{i+1}'' are chosen to ensure that $\tau(X)\in C^2[a,b]$ when $\tau'(a)$ and $\tau'(b)$ are specified. This results in

$$\tau(X) = \frac{1}{p_i^2S_i}\{\tau_i''\sinh p_i(X_{i+1}-X) + \tau_{i+1}''\sinh p_i(X-X_i)\}$$
$$+\left(f_i - \frac{\tau_i''}{p_i^2}\right)\frac{(X_{i+1}-X)}{h_i} + \left(f_{i+1}-\frac{\tau_{i+1}''}{p_i^2}\right)\frac{(X-X_i)}{h_i}$$

τ_i'' $(i=1,\ldots,N+1)$ are determined from the tridiagonal system

$$d_1\tau_1'' + e_1\tau_2'' = b_1$$
$$e_{i-1}\tau_{i-1}'' + (d_{i-1}+d_i)\tau_i'' + e_i\tau_{i+1}'' = b_i \quad (i=2,\ldots,N)$$
$$e_N\tau_N'' + d_N\tau_{N+1}'' = b_{N+1}$$

As is well known, such tridiagonal systems can be solved in $O(N)$ arithmetic operations.

Two limiting cases are of interest to us.

1) $p_i \to 0 \Rightarrow [D^4 - p_i^2D^2]\tau = 0 \Rightarrow [D^4]\tau = 0$. Hence, in this case we recover the cubic spline.

2) $p_i \to \infty \Rightarrow [D^4 - p_i^2D^2]\tau = 0 \Rightarrow [(1/p_i^2)D^4 - D^2]\tau = 0 \Rightarrow [D^2]\tau = 0$. Hence, this limit yields the polygonal or broken line interpolant. Although this argument is not rigorous, the conclusions are valid and can be substantiated by a more intricate argument.

The beam analogy led us to suspect that we could eliminate extraneous inflection points (EIP's) by resorting to exponential splines. Späth[9] and Pruess[6] have indeed established that for "sufficiently large" tension parameters $(p_i;$ $i=1,\ldots,N)$ $\tau(x)$ is free of EIP's. The question next arises as to how to produce such a set $\{p_i\}_{i=1}^N$. Moreover, we do not wish to choose these tension parameters very much larger than required to eliminate EIP's. Otherwise we obtain an interpolant that is overly kinky in appearance.

Observe that, on $[x_i,x_{i+1}]$,

$$\tau''(X) = \tau_i''\frac{\sinh p_i(X_{i+1}-X)}{\sinh p_ih_i} + \tau_{i+1}''\frac{\sinh p_i(X-X_i)}{\sinh p_ih_i}$$

so that if $b_ib_{i+1}>0$ we need only require that $\tau_i''b_i>0$ and $\tau_{i+1}''b_{i+1}>0$ in order to guarantee that $\tau''(x)$ has this common sign. We thus are led to search for $\{p_i\}_{i=1}^N$ such that $\tau_i''b_i>0 (i=1,\ldots,N+1)$. There are certain special cases that must be considered but these need not concern us here. For those details and the derivation of the tension parameter selection algorithm see Ref. 7. In the event that $\tau_K''b_K<0$ we define

$$\bar{\lambda} = \frac{\max(|b_K|, (d_{K-1}+d_K)|\tau_K''|)}{2\max(|\tau_{K-1}''|, |\tau_{K+1}''|)}$$

and let $\tilde{p}_i = \max[(\bar{\lambda}h_i)^{-1/2}, p_i]$; $i=K-1, K$. We then update the tension parameters via

$$p_i = p_i + \omega(\tilde{p}_i - p_i); \quad i=K-1, \quad K$$

The algorithm essentially proceeds as follows.

1) Fit the data with $p_i=0\forall i$; i.e., use the cubic spline as the zeroth iterate.

2) Check $\tau_i''b_i$ $(i=1,\ldots,N+1)$ for positivity.

3) If there are EIP's, iteratively enforce $\tau_i''b_i>0$ $(i=1,\ldots,N+1)$.

Previous treatments of exponential spline interpolation provided no viable means for choosing the tension parameters. The use of the proposed procedure has rendered interpolation by exponential splines not only practicable but also highly desirable.

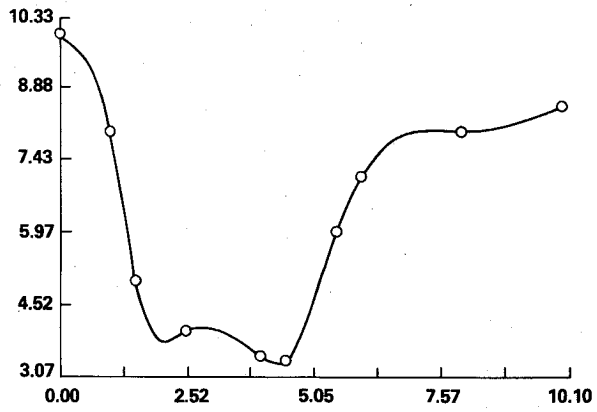
We note that the interpolant so produced provides a fourth-order accurate approximation to the function, a third-order accurate approximation to the first derivative, a second-order accurate approximation to the second derivative, and a first-order accurate approximation to the third derivative. However, for uniform mesh and tension the first-derivative approximation becomes fourth-order accurate. Moreover, under these conditions, we can use this interpolant to produce a fourth-order accurate approximation to the second derivative and second-order accurate approximations to the third and fourth derivatives. See Ref. 7 for proofs and details.

Examples

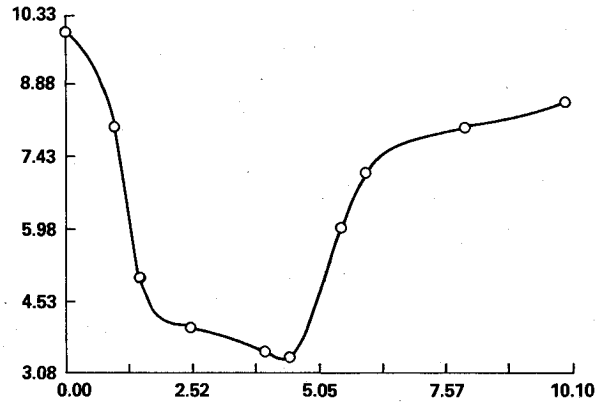
We next present a sequence of carefully selected examples. They are chosen to illustrate both the efficiency of the new parameter selection algorithm and the inherent superiority to cubic spline interpolation.

The first test case (Fig. 1) is taken from Späth's original paper.[9] The cubic spline interpolant exhibits extraneous inflection points in the first, third, fourth, and eighth intervals. The converged exponential spline interpolant is seen to be free of such aberrations. The general behavior of our parameter selection scheme is amply portrayed in this example. The first iteration captures the gross features while subsequent iterations essentially "fine-tune" the first. We note that the scheme proposed by Späth required twelve iterations as opposed to our three iterations with no visible difference in the final interpolants.

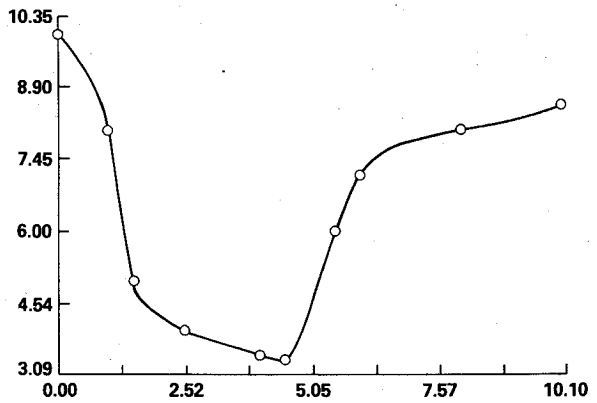
The second test case (Fig. 2) is a unit impulse function. Note the wiggles present in the cubic spline interpolant. This example demonstrates the insensitivity to "outliers" that the exponential spline interpolant possesses.



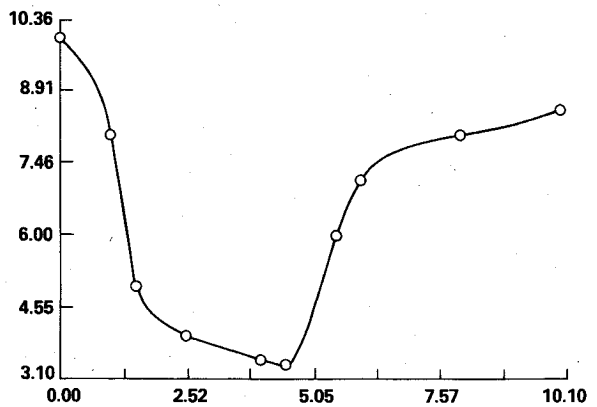
a) Cubic spline, iteration 0.



b) Exponential spline, iteration 1.

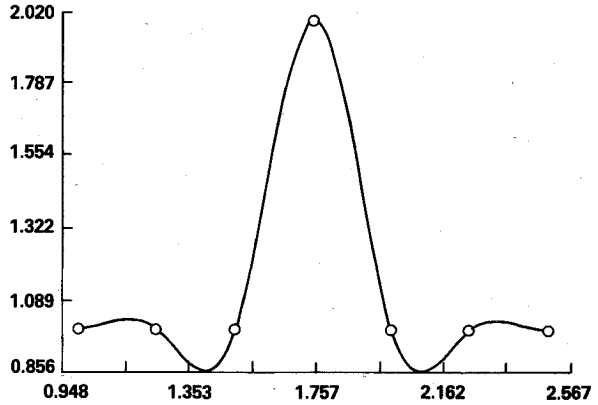


c) Exponential spline, iteration 2.

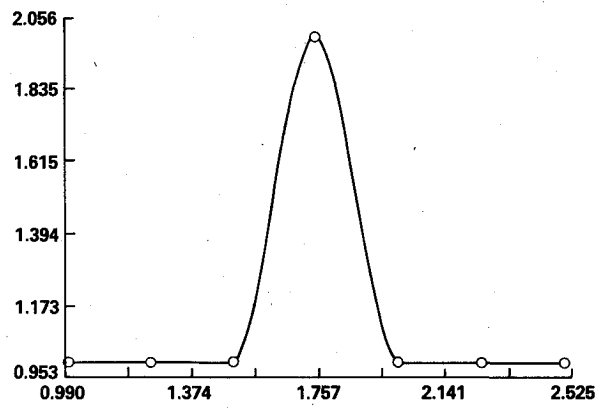


d) Exponential spline, iteration 3.

Fig. 1 Späth test case.

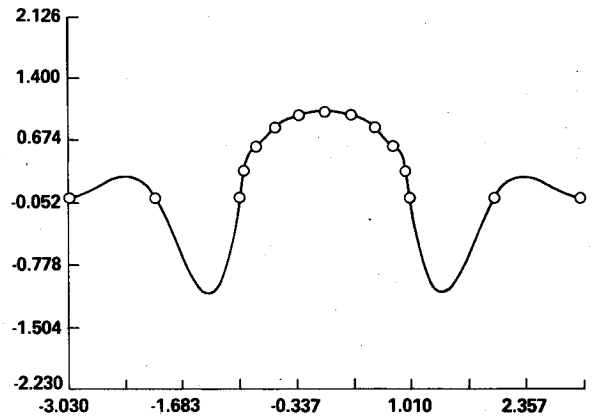


a) Cubic spline.

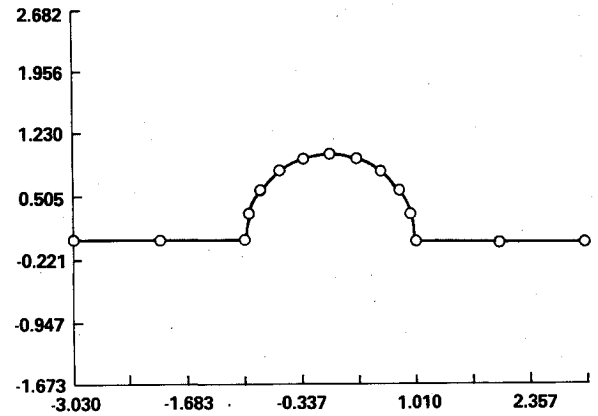


b) Exponential spline.

Fig. 2 Outlier test case.

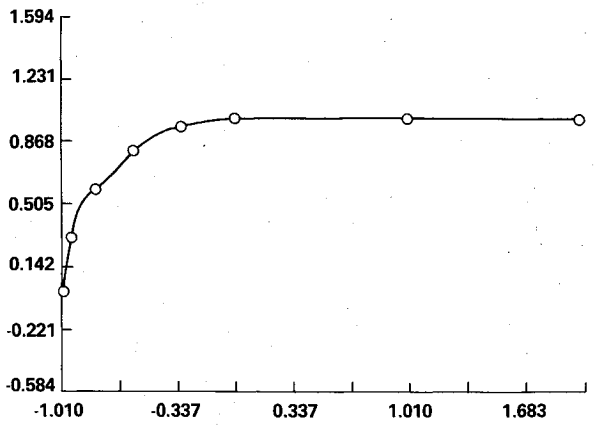


a) Cubic spline.

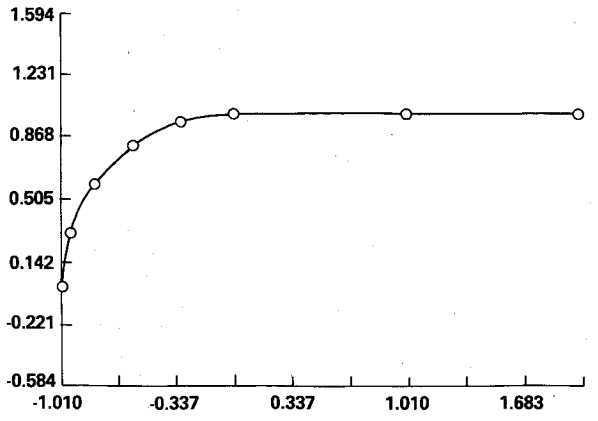


b) Exponential spline.

Fig. 3 Slope discontinuity test case.

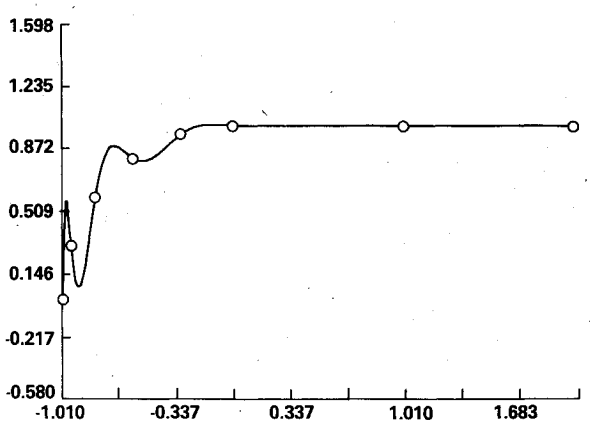


a) Cubic spline.

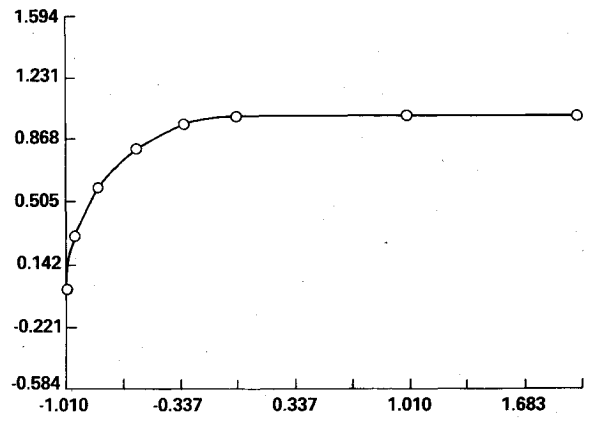


b) Exponential spline.

Fig. 4 Curvature discontinuity test case.



a) Cubic spline.



b) Exponential spline.

Fig. 5 Vertical tangent test case.

The third test case (Fig. 3) is a semicircle joined to two straight line segments in such a way as to produce discontinuities in the first derivative. This example begins to implicate the cubic spline interpolant as being deficient as a means of geometric representation. The exponential spline, on the other hand, performs ideally in this instance. Such a geometry would arise as the computational domain for flow past a cylinder.

The fourth test case (Fig. 4) is a quarter-circle joined by a straight line segment with a discontinuous second derivative at their junction. Once again the cubic spline interpolant falls far short of geometric requirements while the exponential spline does not falter. If we were to try and calculate supersonic flow past this projectile, the cubic spline inflection point would likely induce a shock wave.

The fifth test case (Fig. 5) displays the critical sensitivity of the cubic spline interpolant to the end conditions imposed. It is an additional advantage of the exponential spline interpolant that it automatically compensates for poor end conditions, thus restricting them to local influence. If this geometry were axisymmetric the cubic spline would surely replace the blunt body by a pointed one.

Furthermore, we note that preliminary tests have revealed wildly oscillating cubic spline interpolants in the presence of nonuniform knot placement. This defect appears to be absent from the exponential spline interpolant.

An additional comment deserves to be made at this point. That is, in the applications where cubic splines are typically used, a great deal of effort is generally expended in the smoothing ("fudging") of geometric data and in the preoccupation with end conditions. Both of these difficulties are peculiar to the cubic spline. In general, exponential spline interpolation obviates the need for such diversions.

Applications

The remainder of this paper presents a wide variety of applications of exponential splines in computational fluid dynamics. The emphasis throughout is on areas of application where the use of cubic splines would result in a significant degradation if not an utter breakdown of the computation.

The utility of exponential splines as a means of geometric representation has already been intimated. The resulting loss of accuracy in the numerical prescription of initial and/or boundary conditions could distort the solution throughout the entire flowfield. In a boundary conforming computational mesh, any abnormalities along the boundary will propagate into the grid. Smith and Wiegel[10] and Eiseman and Smith[11] have advocated the use of hyperbolic splines in this context. This approach requires that uniform tension be applied, thus resulting in unnecessarily kinky curves.

Along these same lines, conformal mapping techniques often require ex post facto stretchings to concentrate grid points in regions of high gradients. Such C^2 shearings may be

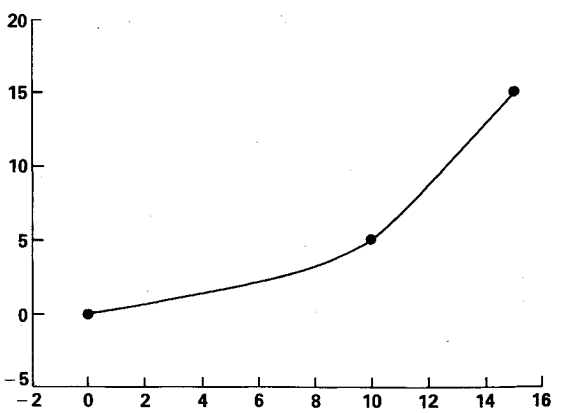


Fig. 6 Stretching function; exponential spline.

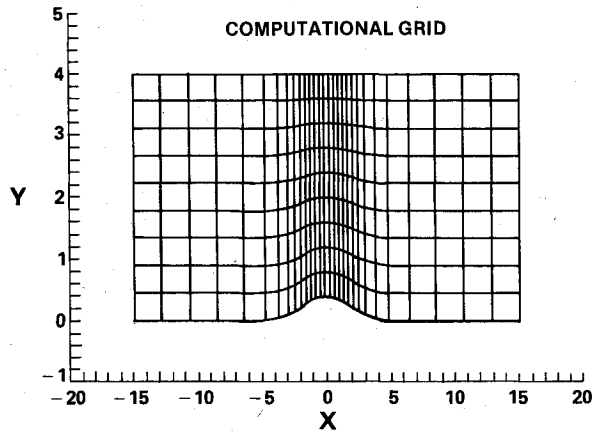


Fig. 7 Stretched mesh; computational grid.

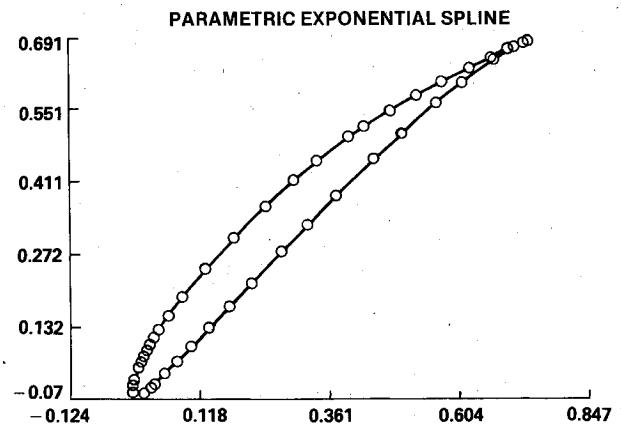


Fig. 8 Korn cascade airfoil; parametric exponential spline.

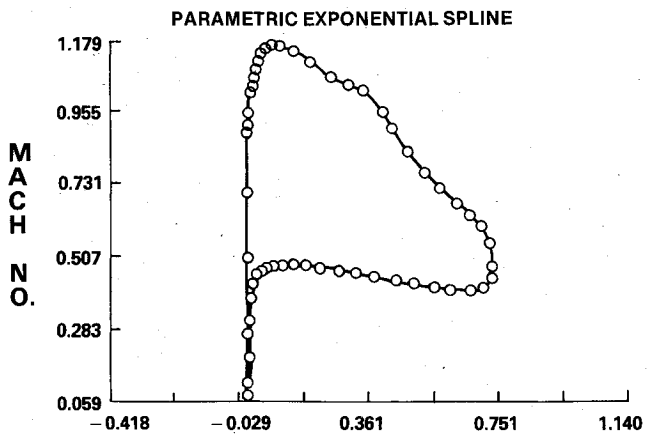


Fig. 9 Computed Mach number distribution; parameteric exponential spline.

conveniently constructed and controlled by employing exponential splines. Figure 6 displays such a stretching function while Fig. 7 illustrates its application to a channel-with-bump geometry.

In a slightly more general vein, we recall that strict adherence to convexity constraints is required by many existence[12] and uniqueness[13] results in the theory of plane subsonic fluid flow. It is also well known that in supersonic flows, there is an inherent difference between concave and convex geometries.[14] These considerations imply that convexity must be preserved in the discrete analogs to these problems, e.g., through the use of exponential splines.

The role of exponential splines in the numerical solution of singular perturbation problems has recently been investigated by Flaherty.[15] We present an application of our exponential spline code[7] to the fluid dynamicist's favorite singular perturbation problem; the boundary-layer equations.

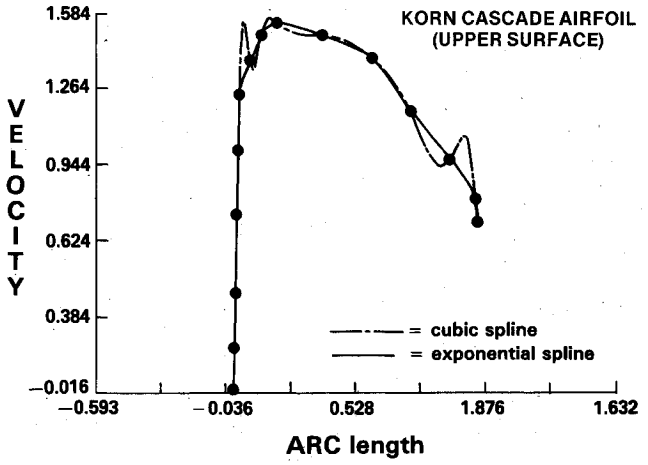


Fig. 10 Velocity schedules (15 points).

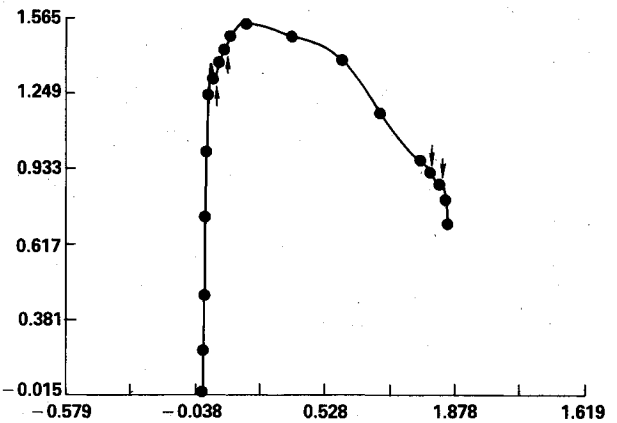


Fig. 11 Cubic spline velocity schedule (19 points).

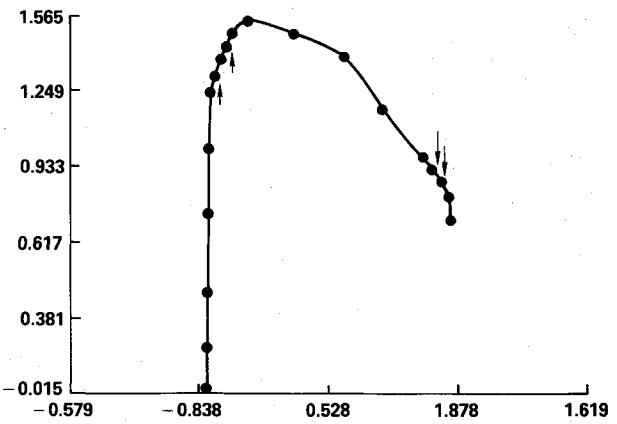


Fig. 12 Exponential spline velocity schedule (19 points).

Specifically, we consider the transonic flow (peak Mach number $\simeq 1.2$) over a compressor cascade with 10 deg of flow turning as supplied by Korn[16] (see Fig. 8). The computed velocity distribution along the upper surface (obtained from a hodograph method) is shown in Fig. 9. Fifteen points were selected and fit with both cubic and exponential splines (see Fig. 10). The cubic spline oscillates wildly in the vicinities of the supersonic zone and the trailing edge.

The resulting velocity distributions were then input to the STAN5 finite difference boundary-layer code.[17] The anomalies present in the cubic spline interpolant cause the boundary layer to separate at the leading edge overspeed thus halting the calculation. On the other hand, the well-behaved

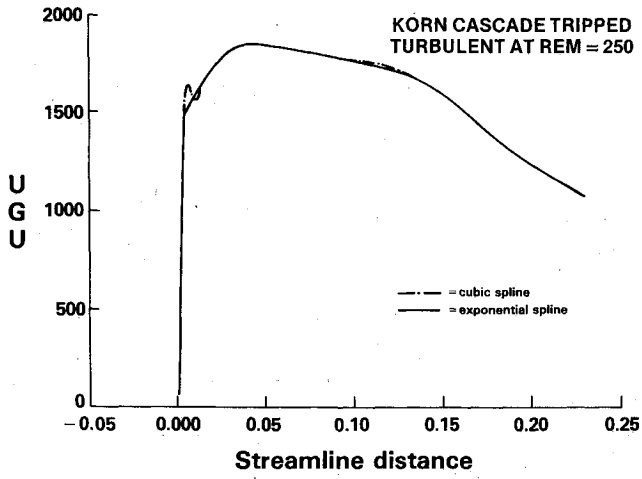


Fig. 13 Input speed distributions (19 points).

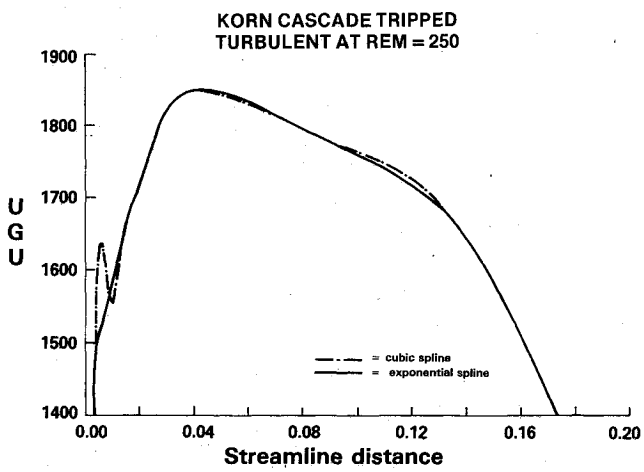


Fig. 14 Blowup of regions of variation.

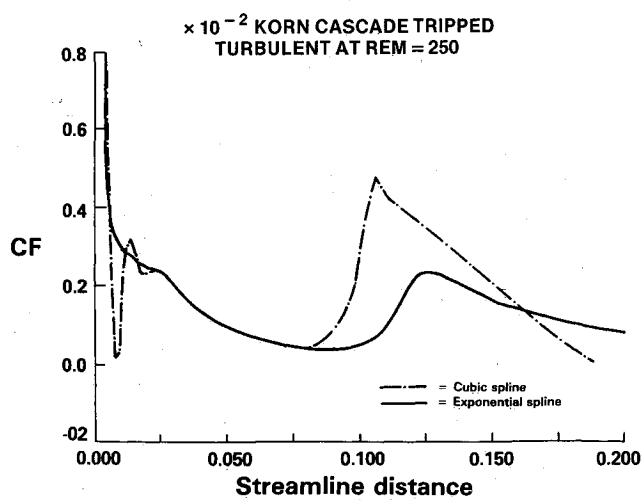


Fig. 15 Skin friction coefficient.

nature of the exponential spline interpolant allows the calculation of a fully attached flow.

In order to obtain a more substantial comparison we include four additional points on the velocity schedule (see Figs. 11 and 12). These are then used as input to STAN5 (see Figs. 13 and 14). The output, displayed in Figs. 15 and 16, clearly indicates that the cubic spline can produce nonphysical features in the computed flowfield.

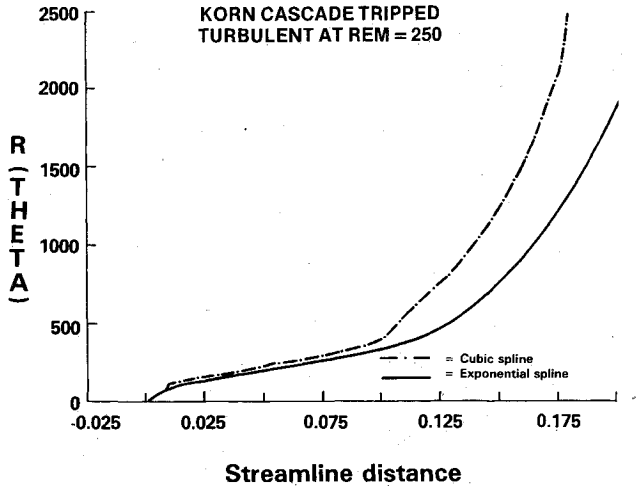


Fig. 16 Momentum thickness Reynolds number.

In contrast, the exponential spline, because of its stricter adherence to the features of the data, is seen to be more reliable. One can well imagine the additional complications which the presence of a shock wave would entail.

Obviously, the use of an extremely fine mesh would rectify the situation. However, a high cost could be incurred in the form of vastly increased computation time. This is especially true when the boundary-layer calculation is not being performed a posteriori but rather as part of an inviscid-viscous iteration. Furthermore, the use of the exponential spline would allow the specification of less data than the cubic spline in an inverse design procedure.

Conclusion

In summary, the examples considered indicate that the cubic spline may cause difficulties in a broad spectrum of practical applications. In the past, the lack of a satisfactory scheme for tension parameter selection has hindered the widespread use of the exponential spline as a viable alternative. However, our exponential spline tension parameter selection algorithm has removed this impediment thus paving the way for it to become an important tool of the computational fluid dynamicist.

Acknowledgments

Thanks are due to Prof. A. Jameson for his guidance, Dr. T. J. Barber for suggesting and consulting in this endeavor, and Mr. S. Dunne for collaboration on the boundary-layer calculations. This work was supported in part under Contact DE-AC02-76ER03077 with the U.S. Department of Energy, Division of Basic Energy Sciences, Applied Mathematical Sciences Program; and NASA Grants NGT-33-016-800 and NGT-33-016-201.

References

[1] Davis, P. J., *Interpolation and Approximation*, Dover Press, New York, 1975.

[2] Dahlquist, G. and Björck, A., *Numerical Methods*, Prentice-Hall, Englewood Cliffs, N.J., 1974.

[3] Ahlberg, J. H., Nilson, E. N., and Walsh, J. L., *The Theory of Splines and Their Applications*, Academic Press, New York, 1967.

[4] Schweikert, D. G., "An Interpolation Curve Using a Spline in Tension," *Journal of Mathematics and Physics*, Vol. 45, 1966, pp. 312-317.

[5] Späth, H., *Spline Algorithms for Curves and Surfaces*, Utilitias Mathematica, Winnipeg, Manitoba, 1974.

[6] Pruess, S., "Properties of Splines in Tension," *Journal of Approximation Theory*, Vol. 17, 1976, pp. 86-96.

[7] McCartin, B. J., "Theory, Computation and Application of Exponential Splines," DOE/ER/03077-171, Oct. 1981.

[8] Pratt, T. K. and Seitelman, L. H., *Practical Numerical Methods*, unpublished manuscript, 1981.

[9] Späth, H., "Exponential Spline Interpolation," *Computing*, Vol. 4, 1969, pp. 225-233.

[10] Smith, R. E. and Wiegel, B. L., "Analytical and Approximate Boundary Fitted Coordinate Systems for Fluid Flow Simulation," AIAA Paper 80-0192, 1980.

[11] Eiseman, P. R. and Smith, R. E., "Mesh Generation Using Algebraic Techniques," NASA/ICASE Workshop on Numerical Grid Generation Techniques for Partial Differential Equations, Oct. 1980, pp. 73-120.

[12] Birkhoff, G. and Zarantonello, E. H., *Jets, Wakes, and Cavities*, Academic Press, New York, 1957, pp. 139 and 153.

[13] Garabedian, P. R., *Partial Differential Equations*, John Wiley & Sons, New York, 1964, p. 599.

[14] Courant, R. and Friedrichs, K. O., *Supersonic Flow and Shock Waves*, Springer-Verlag, 1976, pp. 273-282.

[15] Flaherty, J. E. and Mathon, W., "Collocation with Polynomial and Tension Splines for Singularly-Perturbed Boundary Value Problems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 1, No. 2, June 1980.

[16] Korn, D. G., "Numerical Design of Transonic Cascades," Courant Institute of Mathematical Sciences, ERDA Mathematics and Computing Laboratory, New York University, Rept. C00-3077-72, Jan. 1975.

[17] Crawford, M. E. and Kays, W. M., "STAN5—A Program for Numerical Computation of Two Dimensional Internal and External Boundary Layer Flows," NASA CR-2742, 1976.