Pseudospline Interpolation for Space Curves

By D. H. Thomas

Abstract. A method for interpolating a curve through points in space is described. It is the direct analogue of Fowler-Wilson or pseudospline interpolation for plane curves in that local coordinate systems, cubic polynomials of suitable parameters, and mildly nonlinear equations are used to obtain a continuous interpolating curve with continuous tangent and curvature vectors.

1. **Introduction.** In the last two decades advances in computer-aided design and numerical control technology have stimulated considerable interest in the development and use of practical algorithms for the interpolation and approximation of points representing plane and, to some extent, space curves. Much of this interest surrounds the design and representation of smooth free form shapes by curves exhibiting at least "visual" C^2 continuity. For problems of this type it has become fashionable to adopt techniques based on various analogies and approximations to the draftsman's or the mechanical spline. A milestone in the history of mechanics, Bernoulli-Euler's eighteenth century treatment of the elastica was, in effect, the first definitive mathematical result on the latter subject [10]. In 1906 the physicist Max Born gave an elegant account of both the two-and three-dimensional elastica using an intrinsic coordinate formulation to "solve" the problem (in terms of elliptic functions) under a variety of boundary conditions [3]. In recent years the nonlinear spline and, more generally, the elastica have attracted renewed attention, this time, strictly from the point of view of interpolation theory with obvious implications for curve fitting methodology, [2], [8], [9], and [12].

On the more mundane side, most of today's users and developers of curve fitting routines continue to rely heavily on new or established methods usually connected with the linear theory of interpolation and approximation of functions (whether motivated by splines or not). For example, if the intended curve is "functional" in some fixed Cartesian frame of reference, the approximation method (whatever it may be) is most easily applied once and/or twice, respectively, using the special parametric representations $(x, f(x))$ for plane curves and/or $(x, f(x), g(x))$ for space curves. Of course, more generally and usually at no additional programming expense (only computational) the parametric representations $(f(s), g(s))$ and $(f(s), g(s), h(s))$ are employed, where s is some parameter monotone with the assigned ordering of the data points. Although now the available procedure is called

---

on two and three times, respectively, the resulting method makes no distinction between functional and arbitrarily varying curves. It is in the folklore to most commonly use accumulated chord length for s, perhaps because of the obvious interpretation of it as an approximation (posterior) to the intrinsic parameter (arc length) of the interpolant itself. This parametrization using linear cubic spline interpolants for the coordinate functions was recommended by Ahlberg in [2] and recently Cline [4] combined it with Schweikert's spline in tension method [13]. Among other approaches which are closer in spirit to the one described below we mention the parametric or vector-valued cubic splines in which the interpolating curve is represented by cubic polynomial coordinate functions in a variable u between every pair of data points (u varying between 0 and 1). The imposition of C^2 continuity requirements does not determine this parametrization uniquely but only up to certain normalization factors for each segment which, when varied, give rise to different cubic spline interpolants [5], [6], [11].

As a departure from the above philosophy, A. H. Fowler and C. W. Wilson in an earlier unpublished laboratory report [7] describe a C^2 planar curve fitting algorithm based on a nonlinear interpolation method, what we have called pseudospline interpolation in [2], which uses cubic polynomials in local coordinate systems.

As is well known, linearized cubic spline interpolation is derivable from the theory of the slight bending of thin beams via the interpretation of $\int y''^2 \, dx$ as a good approximation to the line integral $\oint \kappa^2 ds$ (where κ is the curvature as a function of arc length). The cubic polynomial provides a minimum of the first integral, and subject to the interpolatory and C^2 smoothness conditions, the linearized cubic spline results as the solution of a simpler variational problem (cf. [1]). Fowler-Wilson's use of the rotated system of cubic polynomials more fully exploits the optimum variational property of a basic cubic polynomial in that large variations in slope from one end of the curve to the other are removed. Thus, not only are arbitrarily varying curves well modeled but, roughly, one expects the "strain energy" in a Fowler-Wilson interpolant to be more accurately represented by $\Sigma \int y''^2 \, dx$ than in its linearized counterpart even to the same interpolatory conditions. Although effecting the interpolation depends on solving a set of nonlinear rather than linear equations, these are of a special type, and the computational process is still very rapid when compared with true nonlinear spline interpolation recently considered, for example, by Forsythe and Lee in [9]. In this author's opinion the Fowler-Wilson method has proven to be a usable and competitive approach to planar curve fitting of the type which exhibits the mechanical spline syndrome.

In this note we describe a direct extension of their basic scheme to points in space, detailing the relevant formulas and offering some remarks on the resulting method. The reader can review planar pseudospline interpolation as a special case of the discussion given below.

Let $P_0, P_1, \ldots, P_n$ be an ordered sequence of points with coordinates (x_i, y_i, z_i). Let tangent vectors $\vec{\tau}_0, \vec{\tau}_n$ be prescribed at P_0 and P_n. Each segment of the curve between points P_{i-1}, P_i will be described in the parametric form ($\xi, \eta(\xi), \hat{\eta}(\xi)$) where $\eta(\xi), \hat{\eta}(\xi)$ are cubic polynomials of ξ relative to Cartesian coordinates with

ξ-axis parallel to the chord $\overline{P_{i-1}P_i}$. These cubics are determined by requiring continuity of tangent and curvature vectors at each joint (i.e., interior point). The detailed description of the scheme is divided into a number of steps and is most conveniently given in the language of vector analysis.

2. **Constants for Local Coordinate Systems.** One of the main problems in establishing local orthogonal systems stems from their nonuniqueness, specifically, the $\eta, \hat{\eta}$ coordinate directions, which are unique only up to a plane rotation. The following choice of coordinate systems seems to have the most computational merit. Let $\vec{u}_i$ be $\vec{i}, \vec{j}$, or $\vec{k}$, the unit vector in the direction of the component of $\vec{\delta}_i = (x_i - x_{i-1}, y_i - y_{i-1}, z_i - z_{i-1})$ of least magnitude. Define $\vec{\eta}_i, \vec{\hat{\eta}}_i$ by the equations

(1a)
$$\vec{\eta}_i = \vec{u}_i \times \vec{\delta}_i / \| \vec{u}_i \times \vec{\delta}_i \|$$

and

(1b)
$$\vec{\hat{\eta}}_i = \vec{\delta}_i \times \vec{\eta}_i, \qquad i = 1, 2, \ldots, n,$$

where "$\times$" denotes vector or cross product and "$\| \ \|$" denotes ordinary Euclidean length. The system $\vec{\delta}_i, \vec{\eta}_i, \vec{\hat{\eta}}_i$ then represents three mutually orthogonal unit vectors, i.e. a local Cartesian coordinate system with origin at $(P_{i-1} + P_i)/2$. By construction, $\vec{u}_i$ always makes an angle of at least $60°$ with $\vec{\delta}_i$. Hence, the vector products in (1) are mathematically (and numerically) well defined for any given set of distinct points $P_0, P_1, \ldots, P_n$.

The 3×3 matrices A_i with columns given by the components of the vectors $\vec{\delta}_i, \vec{\eta}_i, \vec{\hat{\eta}}_i$, i.e.,

(2)
$$A_i = \begin{pmatrix} \delta_i^1 & \eta_i^1 & \hat{\eta}_i^1 \\ \delta_i^2 & \eta_i^2 & \hat{\eta}_i^2 \\ \delta_i^3 & \eta_i^3 & \hat{\eta}_i^3 \end{pmatrix}, \qquad i = 1, 2, \ldots, n,$$

are orthogonal transformation matrices. Any vector $\vec{u}$ with coordinates $(\alpha_1, \alpha_2, \alpha_3)$ given in the original system is represented in the ith local coordinate system with coordinates $(\beta_1, \beta_2, \beta_3)$ given by

(3)
$$(\beta_1, \beta_2, \beta_3) = (\alpha_1, \alpha_2, \alpha_3) \cdot A_i.$$

The matrices A_i are needed later to represent evaluated points (off the space curve) in the original coordinate system and hence, should be retained, once computed, as a part of the final complete description of the interpolating curve.

In order to obtain coupling conditions at joints, it is also convenient to compute the $n - 1$ 3×3 matrices B_i defined by the matrix equations

(4)
$$B_i = A_i' A_{i+1}, \qquad i = 1, 2, \ldots, n - 1,$$

(where A_i' is the transpose of A_i) which relate conditions in one local coordinate system (the ith) with those of the immediately adjoining system (the $(i + 1)$st).

3. **Coupling Equations from Continuity Conditions.** Let R be the position vector of points on the *intended* space curve relative to the given local coordinate system. Thus, as before, $\vec{R}$ has the form

(5) $$\vec{R} = (\xi,\, \eta(\xi),\, \hat{\eta}(\xi)),$$

where $\eta(\xi)$, $\hat{\eta}(\xi)$ are cubic polynomials of ξ. The unit tangent vector $d\vec{R}/ds$ (where s is arc length) is given by

(6) $$\vec{\tau} \equiv \frac{d\vec{R}}{ds} = \left(1, \frac{d\eta}{d\xi}, \frac{d\hat{\eta}}{d\xi}\right) \Big/ \left[1 + \left(\frac{d\eta}{d\xi}\right)^2 + \left(\frac{d\hat{\eta}}{d\xi}\right)^2\right]^{1/2}$$

and the curvature vector, $d^2\vec{R}/ds^2$, by (cf. [14])

(7) $$\vec{\kappa} \equiv \frac{d^2\vec{R}}{ds^2} = [(\dot{\vec{R}} \cdot \dot{\vec{R}})\ddot{\vec{R}} - (\dot{\vec{R}} \cdot \ddot{\vec{R}})\dot{\vec{R}}]/(\dot{\vec{R}} \cdot \dot{\vec{R}})^2,$$

where $\dot{\vec{R}} = d\vec{R}/d\xi$, $\ddot{\vec{R}} = d^2\vec{R}/d\xi^2$ and the notation $(\vec{u} \cdot \vec{v})$ denotes the inner or dot product of vectors $\vec{u}$, $\vec{v}$.

Relations (6) and (7) will now be used to obtain two functionally independent conditions for each interior point of interpolation. As in [2], the conditions that $\eta = \eta(\xi)$ and $\hat{\eta} = \hat{\eta}(\xi)$ be cubic functions of ξ are satisfied by setting

(8a) $$\eta(\xi) = (\xi^2 - c_i^2)(\alpha_i \xi + \beta_i), \qquad |\xi| \leqslant c_i,$$

and

(8b) $$\hat{\eta}(\xi) = (\xi^2 - c_i^2)(\hat{\alpha}_i \xi + \hat{\beta}_i), \qquad |\xi| \leqslant c_i,$$

where the constants α_i, $\hat{\alpha}_i$, β_i, $\hat{\beta}_i$ determine the tangent vectors at the $(i-1)$st and ith points and $c_i = \|\vec{\xi}_i\|/2$. If we define λ_i, $\hat{\lambda}_i$, μ_i, $\hat{\mu}_i$ by

(9a) $$\frac{d\eta}{d\xi}(-c_i) = \lambda_i, \qquad \frac{d\hat{\eta}}{d\xi}(-c_i) = \hat{\lambda}_i$$

and

(9b) $$\frac{d\eta}{d\xi}(c_i) = \mu_i, \qquad \frac{d\hat{\eta}}{d\xi}(c_i) = \hat{\mu}_i,$$

then these constants depend linearly on λ_i, μ_i, $\hat{\lambda}_i$, $\hat{\mu}_i$ and are explicitly given by

(10a) $$\alpha_i = \frac{\mu_i + \lambda_i}{4c_i^2}, \qquad \beta_i = \frac{\mu_i - \lambda_i}{4c_i}$$

and

(10b) $$\hat{\alpha}_i = \frac{\hat{\mu}_i + \hat{\lambda}_i}{4c_i^2}, \qquad \hat{\beta}_i = \frac{\hat{\mu}_i - \hat{\lambda}_i}{4c_i}$$

At the ith interior point the condition of tangent vector continuity means that the vector represented by

(11a) $$\vec{\tau}_{i-} = (1, \mu_i, \hat{\mu}_i)/(1 + \mu_i^2 + \hat{\mu}_i^2)^{1/2}$$

in the ith coordinate system is one and the same as the vector represented by

(11b) $\vec{\tau}_{i+} = (1, \lambda_{i+1}, \hat{\lambda}_{i+1})/(1 + \lambda^2_{i+1} + \hat{\lambda}^2_{i+1})^{1/2}$

in the $(i + 1)$st local system. Therefore, (11a) and (11b) are related by the orthogonal matrix B_i, i.e.,

(12) $\vec{\tau}_{i-} = \vec{\tau}_{i+} \cdot B_i.$

Equation (12) is a vector equation equivalent to three scalar equations, of which only two, however, are independent. Equating the ratios of the second to the first components in (12), one obtains

(13a) $$\mu_i = \frac{b^i_{21} + b^i_{22} \lambda_{i+1} + b^i_{23} \hat{\lambda}_{i+1}}{b^i_{11} + b^i_{12} \lambda_{i+1} + b^i_{13} \hat{\lambda}_{i+1}}$$

(the b^i_{lk}, $l, k = 1, 2, 3$ are the elements of B_i). Similarly, equating the ratios of the third to the first components in (12) gives

(13b) $$\hat{\mu}_i = \frac{b^i_{31} + b^i_{32} \lambda_{i+1} + b^i_{33} \hat{\lambda}_{i+1}}{b^i_{11} + b^i_{12} \lambda_{i+1} + b^i_{13} \hat{\lambda}_{i+1}}.$$

This pair of fractional linear transformations connecting the μ_i, $\hat{\mu}_i$ to the λ_{i+1}, $\hat{\lambda}_{i+1}$ is the exact analogue of the single bilinear transformation in [2, Eq. 11]. By virtue of these relations, (13a) and (13b), the original set of $4n - 4$ unknown parameters have been reduced to half that number. The remaining $2n - 2$ parameters λ_2, $\hat{\lambda}_2$, λ_3, $\hat{\lambda}_3$, . . . , λ_n, $\hat{\lambda}_n$ are now considered as "the unknowns" to be computed by requiring continuity of the curvature vector. This latter condition leads to $2n - 2$ nonlinear determining equations which we now derive.

Relation (7) for the curvature vector may be rewritten in terms of the cubic polynomials $\eta(\xi)$, $\hat{\eta}(\xi)$ in the form

(14) $$\vec{\kappa} = \frac{[-(\dot{\eta}\ddot{\eta} + \dot{\hat{\eta}}\ddot{\hat{\eta}}), \ddot{\eta}(1 + \dot{\hat{\eta}}^2) - \dot{\eta}\dot{\hat{\eta}}\ddot{\hat{\eta}}, (1 + \dot{\eta}^2)\ddot{\hat{\eta}} - \dot{\eta}\dot{\hat{\eta}}\ddot{\eta}]}{(1 + \dot{\eta}^2 + \dot{\hat{\eta}}^2)^2}$$

(the dots denote differentiation with respect to ξ). From (8)–(10) we obtain (cf. [2, Eq. 13])

(15a) $\ddot{\eta}(-c_i) = (2\mu_i + \lambda_i)/c_i, \quad \ddot{\hat{\eta}}(-c_i) = (2\hat{\mu}_i + \hat{\lambda}_i)/c_i$

and

(15b) $\ddot{\eta}(c_i) = -(2\lambda_i + \mu_i)/c_i, \quad \ddot{\hat{\eta}}(c_i) = -(2\hat{\lambda}_i + \hat{\mu}_i)/c_i.$

The curvature vector $\vec{\kappa}_i$ at the ith point may now be computed in two distinct ways: namely, in the ith coordinate system by substituting (9b) and (15b) into (14), and then in the $(i + 1)$st system by substituting from (9a) and (15a) (with i replaced by $i + 1$) into (14). The results of these substitutions are

(16a) $\vec{\kappa}_{i-} = (y^i_1, y^i_2, y^i_3)/c_i(1 + \mu^2_i + \hat{\mu}^2_i)^2$

and

(16b) $$\vec{\kappa}_{i+} = (x^i_1, x^i_2, x^i_3)/c_{i+1}(1 + \lambda^2_{i+1} + \hat{\lambda}^2_{i+1})^2,$$

where the y^i_1, y^i_2, y^i_3 and x^i_1, x^i_2, x^i_3 are given by

(17a) $$y^i_1 = 2\mu^2_i + 2\hat{\mu}^2_i + \lambda_i\mu_i + \lambda_i\hat{\mu}_i,$$

(17b) $$y^i_2 = -(2\mu_i + \lambda_i - \mu_i\hat{\lambda}_i\hat{\mu}_i + \lambda_i\hat{\mu}^2_i),$$

(17c) $$y^i_3 = -(2\hat{\mu}_i + \hat{\lambda}_i - \hat{\mu}_i\lambda_i\mu_i + \mu^2_i\hat{\lambda}_i)$$

and

(18a) $$x^i_1 = -(2\lambda^2_{i+1} + 2\hat{\lambda}^2_{i+1} + \mu_{i+1}\lambda_{i+1} + \hat{\mu}_{i+1}\hat{\lambda}_{i+1}),$$

(18b) $$x^i_2 = 2\lambda_{i+1} + \mu_{i+1} - \hat{\mu}_{i+1}\hat{\lambda}_{i+1}\lambda_{i+1} + \mu_{i+1}\hat{\lambda}^2_{i+1},$$

(18c) $$x^i_3 = 2\hat{\lambda}_{i+1} + \hat{\mu}_{i+1} - \mu_{i+1}\lambda_{i+1}\hat{\lambda}_{i+1} + \hat{\mu}_{i+1}\lambda^2_{i+1},$$

respectively. As before, (16a) and (16b) are related by an orthogonal transformation:

(19) $$\vec{\kappa}_{i-} = \vec{\kappa}_{i+} \cdot B_i, \qquad i = 1, 2, \ldots, n - 1.$$

Each of the $n - 1$ vector equations in (19) expresses only two functionally independent conditions (not three). It is again advantageous to select these conditions by first equating the ratios of the second to the first components in (19), and then the ratios of the third to the first components. The fourth degree polynomial factors $c_i(1 + \mu^2_i + \hat{\mu}^2_i)^2$ and $c_{i+1}(1 + \lambda^2_{i+1} + \hat{\lambda}^2_{i+1})^2$ thereby drop out of the equations (and hence out of consideration). After this has been done and the equations rationalized, one obtains

(20a) $$F_i = y^i_2(b^i_{11}x^i_1 + b^i_{12}x^i_2 + b^i_{13}x^i_3) - y^i_1(b^i_{21}x^i_1 + b^i_{22}x^i_2 + b^i_{23}x^i_3) = 0$$

and

(20b) $$G_i = y^i_3(b^i_{11}x^i_1 + b^i_{12}x^i_2 + b^i_{13}x^i_3) - y^i_1(b^i_{31}x^i_1 + b^i_{32}x^i_2 + b^i_{33}x^i_3) = 0$$

for $i = 1, 2, \ldots, n - 1$. This pair of nonlinear equations, (20a) and (20b), for each interior point of interpolation is the desired analogue of the single nonlinear equation per point in [2, Eq. 16].

4. The Nonlinear Equations and the Interpolation Problem. The system to be solved consists of the $2n - 2$ nonlinear equations of (20a) and (20b) which may be written (after the μ_i, $\hat{\mu}_i$ have been substituted for from (13)) in the form

(21a) $$F_i = F_i(\lambda_{i-1}, \hat{\lambda}_{i-1}, \lambda_i, \hat{\lambda}_i, \lambda_{i+1}, \hat{\lambda}_{i+1}) = 0$$

and

(21b) $$G_i = G_i(\lambda_{i-1}, \hat{\lambda}_{i-1}, \lambda_i, \hat{\lambda}_i, \lambda_{i+1}, \hat{\lambda}_{i+1}) = 0,$$

or in vector form as

(21c) $$\vec{F}_i = \vec{F}_i(\vec{\lambda}_{i-1}, \vec{\lambda}_i, \vec{\lambda}_{i+1}) = 0$$

for $i = 1, 2, \ldots, n - 1$, with four end conditions specified $(\lambda_0, \hat{\lambda}_0, \lambda_n, \hat{\lambda}_n)$; i.e., the two end tangent vectors. This vector "tridiagonal" system is the nonlinear analogue of the simple tridiagonal linear system one associates with ordinary cubic spline interpolation [1].

The author is unable to offer a general theorem on existence and/or uniqueness of solution of this system. Moreover, real solutions of the equations may or may not lead to the determination and uniqueness of an actual C^2 interpolating curve. In order to gain some insight into the complications that arise, consider the situation depicted in Figure 1.
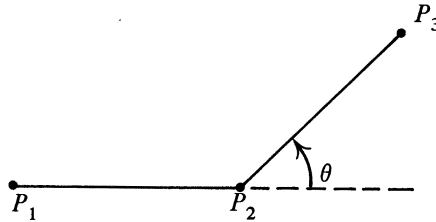

FIGURE 1

P_1, P_2 and P_3 are three points a unit distance apart with end tangent vectors given parallel to $\overrightarrow{P_1P_2}$ and $\overrightarrow{P_3P_2}$, respectively; and θ is the turning angle between $\overrightarrow{P_1P_2}$ and $\overrightarrow{P_2P_3}$ $(-\pi \leqslant \theta \leqslant \pi)$. The system of nonlinear coupling equations reduces to the single cubic equation

$$(22) \qquad (\mu \sin \theta - \cos \theta)^2(\mu \cos \theta - \sin \theta) + \mu = 0$$

for the slope μ (at P_2) referenced in the first coordinate system.

At $\theta = \pm \pi$ any real μ satisfies the coupling condition but, obviously, no C^2 interpolant exists. For $\theta = \pi/2$ $(-\pi/2)$ the coupling condition leads to essentially three possibilities for μ: $0, \infty$, and 1 $(0, \infty, -1)$. The first two possibilities do not lead to a proper interpolant as one of the cubics is undefined, leaving $\mu = 1$ $(\mu = -1)$ as the only candidate which does, in fact, determine a C^2 interpolant. For arbitrary θ the three solutions of (22) are given explicitly by

$$\mu = \tan \theta/2,$$

$$(23) \qquad \mu = \left[(1 - 2 \cos \theta) \cot \theta/2 \pm \sqrt{\frac{1 - 3 \cos \theta}{1 + \cos \theta}}\right] \bigg/ 2 \cos \theta.$$

From (23) we see that for $|\theta| \geqslant \cos^{-1}(1/3)$ there will be three distinct real solutions of (22). A more careful examination will reveal that for "very large" turning angles, namely, in the range $\pi/2 < |\theta| < \pi$ only $\mu = \tan \theta/2$ defines a legitimate C^2 interpolant. The other two solutions for μ lead to curves with cusps at P_2 (although the segments have the same right- and left-handed curvature at P_2). For "large" turning angles, specifically, in the range $\cos^{-1}(1/3) < |\theta| < \pi/2$ it can be verified that each of the three roots leads to a distinct C^2 interpolant. Finally, for any "reasonable" turning angle, defined here to be in the range $0 \leqslant |\theta| \leqslant \cos^{-1}(1/3)$, $\mu = \tan \theta/2$ is the only real root, and it (happily) leads to the existence and uniqueness

of solution to the interpolation problem. As one perturbs the end slopes and unit spacing of Figure 1, a similar situation occurs although conditions on the data are accordingly more complicated. For N points ($N > 3$) we have a concrete existence and uniqueness result in the trivial case of all points lying in a straight line (albeit with arbitrary end slopes given). In this situation the equations in (21) specialize to the familiar linear ones determining an ordinary cubic spline interpolating zero values and the given end conditions. Combining this observation with the local considerations above, it follows by a continuity argument that existence and uniqueness also holds for data perturbed from a straight line. Although these results are of limited interest, they suggest that more generally one might anticipate existence and uniqueness if the data are organized in an "agreeable" way, meaning, for example, when the points are regularly spaced, and turning angles between neighboring points and planes are sufficiently small. Computational experiments tend to support this loosely worded conjecture. However, in practice, the method seems to be applicable to more bizarre data configurations than might be expected from what we have just stated. As matters stand, the author knows of no procedure for a given data set, of resolving the question other than by obtaining the real solutions of the algebraic system in (21) and then, subsequently, examining their validity.

In spite of the nonlinearity and reservations on the question of existence and uniqueness, the method has a number of practical merits. Numerical experimentation suggests the system is rapidly solvable by ordinary Newton iteration once reasonably good initial estimates for the $\{\vec{\lambda}_i\}$ have been made. This is true even for large numbers of points because the basic iterative step involves only a banded (6-diagonal) linear system. Moreover, for this particular system of equations there is a geometrically natural way of obtaining good initial estimates for the unknown parameters, namely through local interpolation methods. For example, for data exhibiting reasonable smoothness the author has found adequate estimates obtained from local interpolation with circles as follows:

$$(24) \qquad \vec{\tau}_i^0 = \frac{\overrightarrow{P_{i-1}P_i}}{\|\overrightarrow{P_{i-1}P_i}\|^2} + \frac{\overrightarrow{P_iP_{i+1}}}{\|\overrightarrow{P_iP_{i+1}}\|^2}$$

is the tangent vector at P_i to the circle interpolating to three points P_{i-1}, P_i, P_{i+1}. This vector should be represented in the $(i + 1)$st coordinate system (by multiplying its components relative to the original system by the transformation matrix A_{i+1}). After this has been done the tangent vector in (24) will have components u_1^i, u_2^i, u_3^i. The initial $\lambda_i, \hat{\lambda}_i$ are now given by

$$(25a) \qquad \lambda_{i+1} = u_2^i / u_1^i$$

and

$$(25b) \qquad \hat{\lambda}_{i+1} = u_3^i / u_1^i \quad \text{for} \quad i = 1, 3, \ldots, n - 1.$$

Once the $\lambda_i, \hat{\lambda}_i$ have been found that satisfy (21a) and (21b), the μ_i and $\hat{\mu}_i$ can be obtained from (13). The $\alpha_i, \beta_i, \hat{\alpha}_i, \hat{\beta}_i$ and hence the cubics $\eta_i, \hat{\eta}_i$ are then deter-

mined by means of (10). The final numerical description (output) of the interpolated curve consists of the coordinates of the original points of interpolation, P_0, P_1, . . . , P_n, the $2n$ cubic polynomials (i.e., the parameters α_i, β_i, $\hat{\alpha}_i$, $\hat{\beta}_i$, c_i, $i = 1, 2, \ldots, n$), and the n 3 × 3 orthogonal matrices A_i given by (2).

5. Discussion. The salient features (both good and bad) of this generalization are comparable to those of the plane curve method. Briefly, it is a class $C^{(2)}$ method which is invariant under translations and rotations of the data (and, more generally, the Euclidean group). The well-known practical and visual advantages of cubic spline curves have been preserved, as "cubics" are used exclusively and in a highly intrinsic way. The nonlinear equations involved are somewhat more complicated than what one might expect from [2], but are still rapidly solved by the usual iterative techniques. Moreover, the method has the ability to interpolate through arbitrarily bent and twisted data point configurations (one of the main advantages in resorting to the parametric and nonlinear techniques). Finally, if points all lie in a plane, the method (and any computer program) reduces to planar pseudospline interpolation.

This interpolation scheme may also be considered related (loosely) to the three-dimensional elastica with clamped end conditions. This interpretation rests on the validity of the representation of "localized strain energy" by the expressions $\int(\eta_i''^2 + \hat{\eta}_i''^2)d\xi$. This connection with the elastica can perhaps be more fully exploited in the data smoothing problem. Specifically, the question of approximation (rather than interpolation) has not been addressed here. However, the method described above could conceivably be made the basis of a general fitting and smoothing procedure in much the same manner that the Fowler-Wilson code [7] "fits" points rather than interpolates them. Individual point movement based on minimizing certain functionals related to the "strain energy" is a natural way in which this might be accomplished. In general, such a functional involves both curvature, μ, and the torsion, τ, and in its simplest form is given by the line integral $\oint (A\mu^2 + B\tau^2)ds$ [3].

6. Acknowledgment. The author wishes to thank G. Birkhoff for originally suggesting this topic in 1965 as a follow-up to [2] and, of course, the referee for helpful comments in preparing this manuscript.

Mathematics Department
General Motors Research Laboratories
General Motors Corporation
Warren, Michigan 48019

1. J. H. AHLBERG, E. N. NILSON & J. E. WALSH, *The Theory of Splines and Their Applications*, Academic Press, New York and London, 1967. MR 39 #684.

2. G. BIRKHOFF, H. BURCHARD & D. THOMAS, *Nonlinear Interpolation by Splines, Pseudosplines, and Elastica*, GMR-468, General Motors Research Laboratories, Warren, Michigan, February 3, 1965.

3. M. BORN, *Untersuchungen über die Stabilität der elastischen Linie in Ebene und Raum, unter verschiedenen Grenzbedingungen*, Inaugural Dissertation, Göttingen, 1906.

4. A. K. CLINE, "Scalar-and-planar-valued curve fitting using splines under tension," *Comm. Assoc. Comput. Mach.*, v. 17, 1974, pp. 218–223.

5. J. FERGUSON, "Multivariable curve interpolation," *J. Assoc. Comput. Mach.*, v. 11, 1964, pp. 221–228. MR 28 #5551.

6. A. R. FORREST, *Curves and Surfaces for Computer-Aided Design*, Ph.D Thesis, Computer Laboratory, Cambridge University, 1968.

7. A. H. FOWLER & C. W. WILSON, *Cubic Spline, A Curve Fitting Routine,* Report Y-1400, Oak Ridge, 1963.

8. J. W. JEROME, "Smooth interpolating curves of prescribed length and minimum curvature," *Proc. Amer. Math. Soc.,* v. 51, 1974, pp. 62–66.

9. E. H. LEE & G. E. FORSYTHE, "Variational study of nonlinear spline curves," *SIAM Rev.,* v. 15, 1973, pp. 120–133. MR 48 #10048.

10. A. E. H. LOVE, *A Treatise on the Mathematical Theory of Elasticity,* 4th ed., Cambridge Univ. Press, London, 1927.

11. J. R. MANNING, "Continuity conditions for spline curves," *Comput. J.,* v. 17, 1974, pp. 181–186.

12. E. MEHLAM, "Nonlinear splines," in *Proceedings of the Conference on Computer-Aided Design,* University of Utah, 1974. (To appear.)

13. D. G. SCHWEIKERT, "An interpolation curve using a spline in tension," *J. Math. and Phys.,* v. 45, 1966, pp. 312–317. MR 34 #6990.

14. D. J. STRUIK, *Lectures on Classical Differential Geometry,* Addison-Wesley, Reading, Mass., 1950. MR 12, 127.

15. D. H. THOMAS, *Pseudospline Interpolation in Space,* MA-13 (1966) and GMR-468 (1974), General Motors Research Laboratories, Warren, Michigan.