PII: S0010-4485(96)00087-5

Computer-Aided Design, Vol. 29, No. 7, pp. 485-496, 1997

0010-4485/97/$17.00+0.00

Energy and B-spline interproximation

Xuefu Wang, Fuhua (Frank) Cheng and Brian A Barsky*

In this paper, we study B-spline curve interproximation with different energy forms and parametrization techniques, and present an interproximation scheme for B-spline surfaces. It shows that the energy form has a much bigger impact on the generated curve than the parametrization technique. With the same energy form, different parametrization techniques generate relatively small difference on the corresponding curves. With the same parametrization technique, however, different energy forms make significant difference on the shape and smoothness of the resulting curves. Furthermore, interproximating B-spline curves generated by minimizing approximated energy forms are far from being good approximations to the optimal curves. They tend to generate flatter regions and sharper turns than curves generated by minimizing the exact energy form. The interproximation scheme for surfaces is aimed at generating a smooth surface to interpolate a grid of data which could either be a point or a region. This is achieved by minimizing a strain energy based on squared principal curvatures for bicubic B-spline surfaces. The surface interproximation process is also studied with different energy forms and parametrization techniques. The test results of the surface interproximation process also show the same conclusion as the curve interproximation process. © 1997 Elsevier Science Ltd.

Keywords: B-splines, interpolation, approximation, interproximation, non-linear programming, centripetal model, relative chord length parametrization, constrained optimization

1. INTRODUCTION

E. T. Y. Lee pointed out in 1990 that some popular strain energy approximation methods for parametric curves do not work as well as might be expected[17]. Unfortunately, probably because the examples given by Lee did not show serious effect in real-life applications, or probably because better alternatives are not available, those strain energy approximation methods continue to be used in various applications. Actually, in some occasions, the results are considered to be acceptable[3–5,14]. One certainly might wonder if what was pointed out by Lee might not be so serious.

For the geometric shape design industry, a better understanding of the performance of the strain energy approximation methods is important since some of the design processes use the energy as a means to optimize the shape (*geometric smoothness*) of a curve or surface[10,12–16,24–27]. One example is the process of constructing a smooth surface to interpolate a network of curves[14], where an energy function is minimized to find the optimal twist vectors for the interpolating surface. Another example is the curve *interproximation* process[4,5] where a curve with the smoothest shape is sought to interpolate given date. The data could be points or regions. The curve interpolates the points and passes through the regions. The curve interproximation process is carried out by minimizing the energy of the curve to achieve geometric smoothness. A surface interproximation technique is not available in the literature yet.

Actually, the geometric smoothness of an interpolating curve or surface depends on knot parametrization as well. Appropriately parametrized knots reduce the energy of the resulting curve/surface and avoid the occurrence of 'oscillations' and 'loops'[5]. It was pointed out by Lee that an interpolating curve whose notes are defined by the *centripetal model* is usually 'fairer' (closer to the data polygon) than curves obtained with the uniform or the chord length model[18]. However, the impact of various parametrization techniques on the curve interproximation process is not known yet.

In this paper, we study the impact of different energy forms and parametrization techniques on the (B-spline) curve interproximation process, and present an interproximation scheme for B-spline surfaces; we are especially interested in the performance of the strain energy approximation methods. Our findings show that, with the same energy form, different parametrization techniques generate relatively small difference on the corresponding curves. With the same parametrization technique, however, different energy forms make significant difference on the shape and smoothness of the resulting curves. Furthermore, interproximating B-spline curves generated by minimizing approximated energy forms are far from being good approximations to the optimal curves. They tend to generate sharper turns and flatter areas in between than curves generated by minimizing the exact energy form. The interproximation scheme for surfaces is aimed at generating a smooth surface to interpolate a grid of data which could either be a point or a region. This is achieved by minimizing a strain energy based on squared principal curvatures for bicubic B-spline surfaces. The surface interproximation

Department of Computer Science, University of Kentucky, Lexington, KY 40506-0046, USA
*Computer Science Division-EECS, University of California, Berkeley, CA 94720-1776, USA
Paper received: 8 July 1995. Revised: 7 October 1996

of D_i. The curve we are seeking for can be expressed as[5]:

$$C(t) = (x(t), y(t), z(t)) = \sum_{i=1}^{n} P_i w_i(t), \qquad t \in [0, 1] \tag{11}$$

with

$$\begin{cases} w_1(t) = \dfrac{t_3 + t_2 - 2t_1}{t_3 - t_1} B_{-2,3}(t) + B_{-1,3}(t) \\ w_2(t) = -\dfrac{t_2 - t_1}{t_3 - t_1} B_{-2,3}(t) + B_{0,3}(t) \\ w_i(t) = B_{i-2,3}(t), \quad i = 3, \ldots, n-2 \\ w_{n-1}(t) = -\dfrac{t_{n-1} - t_n}{t_{n-2} - t_n} B_{n-1,3}(t) + B_{n-3,3}(t) \\ w_n(t) = \dfrac{t_{n-2} + t_{n-1} - 2t_n}{t_{n-2} - t_n} B_{n-1,3}(t) + B_{n-2,3}(t) \end{cases} \tag{12}$$

where $B_{i,3}(t)$ are cubic B-spline basis functions and $P_i = (x_i, y_i, z_i)$ are 3D control points. The P_i are to be chosen so that the energy of $C(t)$ over the parameter space is minimum with the constraint that $C(t_i) \in D_i$ for $i = 1, 2, \ldots, n$. The $w_i(t)$ are computed based on the 'natural' end conditions (i.e. the second derivatives of $C(t)$ at the end points of the parameter space are set to zero). The computation process can be found in Reference 5.

If (5) is considered with (3) being a special case, the minimization process is a quadratic programming problem. By defining $X = (x_1, x_2, \ldots, x_n)$, $Y = (y_1, y_2, \ldots, y_n)$, $Z = (z_1, z_2, \ldots, z_n)$, and $W_{tt} = ((w_1)_{tt}, (w_2)_{tt}, \ldots, (w_n)_{tt})^{\mathrm{T}}$, the energy of $C(t)$ can be expressed as follows:

$$E_1(C) = (X, Y, Z) \cdot \int_I \Omega \mathrm{d}t \cdot \begin{pmatrix} X^{\mathrm{T}} \\ Y^{\mathrm{T}} \\ Z^{\mathrm{T}} \end{pmatrix} \tag{13}$$

where Ω is a $3n \times 3n$ symmetric matrix defined as follows:

$$\Omega = \begin{bmatrix} W_{tt} & 0 & 0 \\ 0 & W_{tt} & 0 \\ 0 & 0 & W_{tt} \end{bmatrix} \cdot Q \cdot \begin{bmatrix} (W_{tt})^{\mathrm{T}} & 0 & 0 \\ 0 & (W_{tt})^{\mathrm{T}} & 0 \\ 0 & 0 & (W_{tt})^{\mathrm{T}} \end{bmatrix} \tag{14}$$

The quadratic programming problem has (13) as its objective function with the constraint:

$$(X, Y, Z) \begin{bmatrix} G & 0 & 0 \\ 0 & G & 0 \\ 0 & 0 & G \end{bmatrix} \in (A_1, A_2, \ldots, A_n, B_1, B_2, \ldots, B_n, C_1, C_2, \ldots, C_n) \tag{15}$$

where A_i, B_i and C_i are defined in (1) and $G = (g_{i,j})$ is the generating matrix of the curve $C(t)$ with $g_{i,j} = w_i(t_j)$. The dimension of G is $n \times n$.

If (6) is considered, note that

$$\Lambda = (\Lambda_1, \Lambda_2, \Lambda_3) \cdot \Theta; \quad \Theta = \begin{bmatrix} H & I & 0 & 0 & 0 & 0 \\ 0 & 0 & H & I & 0 & 0 \\ 0 & 0 & 0 & 0 & H & I \end{bmatrix}$$

with

$$H = ((w_1)_t (w_1)_{tt}, \ldots, (w_1)_t (w_n)_{tt}, \ldots, (w_n)_t (w_1)_{tt}, \ldots, (w_n)_t (w_n)_{tt})^{\mathrm{T}}$$

$$I = ((w_1)_{tt} (w_1)_t, \ldots, (w_1)_{tt} (w_n)_t, \ldots, (w_n)_{tt} (w_1)_t, \ldots, (w_n)_{tt} (w_n)_t)^{\mathrm{T}}$$

and

$$\Lambda_1 = (y_1 z_1, \ldots, y_1 z_n, \ldots, y_n z_1, \ldots, y_n z_n)$$
$$\Lambda_2 = (x_1 z_1, \ldots, x_1 z_n, \ldots, x_n z_1, \ldots, x_n z_n)$$
$$\Lambda_3 = (x_1 y_1, \ldots, x_1 y_n, \ldots, x_n y_1, \ldots, x_n y_n)$$

Hence, the energy of $C(t)$ can be expressed as

$$E_2(C) = (\Lambda_1, \Lambda_2, \Lambda_3) \cdot \int_I (\Theta \cdot R \cdot \Theta^{\mathrm{T}}) \mathrm{d}t \cdot \begin{pmatrix} \Lambda_1^{\mathrm{T}} \\ \Lambda_2^{\mathrm{T}} \\ \Lambda_3^{\mathrm{T}} \end{pmatrix} \tag{16}$$

The minimization of (16) is a quartic programming problem with the same constraint (15).

If (2) is considered, we have a non-linear programming problem with linear constraints. Since there is no enclosed form for the integrand, one has to use the trapezoid method to do the evaluation. The process is tedious, but not difficult.

3. IMPLEMENTATION

The above energy forms, (2), (3), (5) and (6), have been implemented in C in the B-spline curve interproximaton process on the following platform: HPUX level 9.05 on a Hewlett Packard HP 735 machine using a SUN Sparc 20 machine as the display device. The software package used for the optimization process is the *NAG Fortran Library*[21]. The three parametrization techniques: the uniform model, the centripetal model, and the relative chord length model, have all been used in the construction of the parameter knots (t_i) in the interproximation process for each energy form.

A number of test cases have been carried out. Four of them are shown in *Figures 1–12*. In each case, the four energy forms are implemented for each of the parametrization techniques. The resulting curves whose parameter knots are generated using the uniform model, the centripetal model, and the relative chord length model are shown in (a), (b), and (c), respectively. In each subcase, the curves generated by minimizing the energy forms (2), (3), (5) and (6) are shown in width three solid, on off dash, double dash, and width one solid, respectively. The curves are displayed in the following order: on off dash, double dash, width one solid, and width three solid. Therefore, overlapping areas will be shown in the style of the curve which is last rendered.

For each case, three knot vectors are computed using (8), (9), and (10), respectively, first. Based on these values and the corresponding cubic B-spline basis functions defined in (12), the objective functions for (2), (3), (5) and (6) are constructed and minimized subject to constraints (14) to find the control points (P_i) for the corresponding interproximating curves $C(t)$. The objective functions for (3) and (5) are constructed using (13): for (3) the matrix

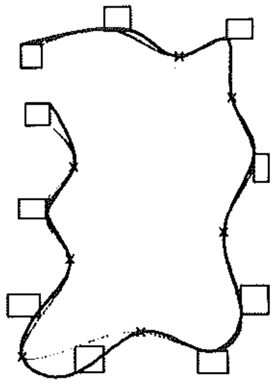

Figure 5 Case 2(b): cubic B-spline curve interproximation

has a much bigger impact on the generated curve than the parametrization technique. With the same energy form, different parametrization techniques generate relatively small difference on the corresponding interproximating curves. With the same parametrization technique, however, different energy forms make significant difference on the shape and smoothness of the resulting curves. This is especially true between curves generated with the exact energy form and the approximated forms. Furthermore, curves generated by minimizing approximated energy forms are far from being good approximations to the optimal curves; they tend to have flatter regions and sharper turns while the curves generated by the exact energy form tend to be more circular and have smoother turns.

The reason that the curves generated by minimizing the approximated energy forms tend to have flatter regions and sharper turns can be explained as follows. First, note that for a flat region, the cross product of the first derivative and the second derivative, $C_t \times C_{tt}$, tends to zero. In addition, if the parametrization has been determined, the second derivative C_{tt} tends to be small since the first derivative would be smaller than a more circular region. Therefore, the values of the energy forms (3), (5), and (6) are small for flat regions. Second, for a sharp turn, although the curvature would be large, the magnitude of the first derivative has to be small to avoid overshooting. This leads to small magnitude of the second derivative. But then the magnitude of $C_t \times C_{tt}$ would be small too. Therefore, the values of the energy forms (3), (5), and (6) are also small for sharp turns. On the other hand, the values of the energy forms (3), (5), and (6) are larger for curves with more circular regions and smoother turns. Consequently, to make the value of the energy forms (3), (5), or (6) small, the curve is forced

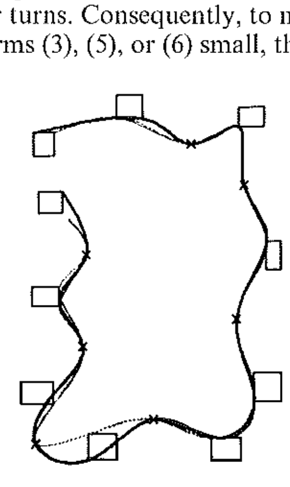

Figure 6 Case 2(c): cubic B-spline curve interproximation

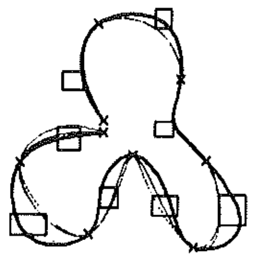

Figure 7 Case 3(a): cubic B-spline curve interproximation

to have flatter regions and sharper turns. This phenomenon has also been proved in the test cases. For instance, in *Figure Case 1(a)*, the curve generated by the exact energy form (2) (called a 2–2 curve) has smoother turns and more circular regions than the curve generated by the approximated energy form (5) (called a 2–5 curve). But the (5) energy of the 2–2 curve is bigger than the (5) energy of the 2–5 curve (recall that *E04NFF* is a global optimization procedure and the 2–2 curve is an element of the solution domain). Actually, we have used the control points of the 2–2 curve as initial configuration for (3) and (5), and we get the 2–3 curve and the 2–5 curve again, respectively.

The test cases show that none of the approximation methods really provide good approximation of the strain energy. Unless flatter regions and sharper turns are what are expected, one should avoid using approximated curve energy forms in a curve interproximation process since they are far from being a good approximation to the exact energy. After all, computing the exact energy can be done in a reasonable amount of time (see *Table 2* for the computation times).

4. B-SPLINE SURFACE INTERPROXIMATION

Given a set of 3D data $D_{i,j}, i = 1, \ldots, m;\ j = 1, \ldots, n$ where

$$D_{i,j} = A_{i,j} \times B_{i,j} \times C_{i,j} = [a_{i,j}, b_{i,j}] \times [c_{i,j}, d_{i,j}] \times [e_{i,j}, f_{i,j}] \quad (17)$$

with $a_{i,j} \leqslant b_{i,j}$, $c_{i,j} \leqslant d_{i,j}$, and $e_{i,j} \leqslant f_{i,j}$, the objective is to construct a bicubic B-spline surface that passes through all $D_{i,j}$ with the smoothest shape. This will be accomplished by finding the interproximating B-spline surface with the smallest *strain energy*.

The interproximation will be performed at the corners of the surface patches. Hence, the interproximating surface needs $(m-1) \times (n-1)$ patches and, consequently, $(m+2) \times (n+2)$ control points and $(m+6)$

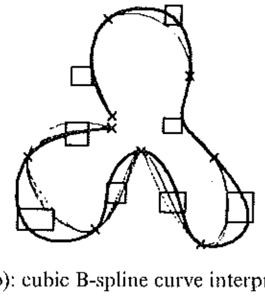

Figure 8 Case 3(b): cubic B-spline curve interproximation

Table 1 (Exact) strain energies of the interproximating curves

Case	Energy form (2)	Energy form (3)	Energy form (5)	Energy form (6)
1(a)	0.128665	0.208134	0.199694	0.251445
1(b)	0.119079	0.203338	0.192632	0.238918
1(c)	0.111950	0.204435	0.197048	0.227865
2(a)	0.567010	0.806265	0.776907	0.783750
2(b)	0.550515	0.757523	0.742433	0.749167
2(c)	0.529589	0.724953	0.707721	0.728271
3(a)	0.366207	0.620149	0.614252	0.743856
3(b)	0.355651	0.625134	0.577276	0.842426
3(c)	0.347007	0.604094	0.541312	0.713505
4(a)	0.411886	0.792969	0.689443	0.835857
4(b)	0.411755	0.751548	0.659106	0.894739
4(c)	0.411470	0.853501	0.625680	0.802757

To find the $P_{i,j}$ that minimize (20), one may proceed as follows. Let P^x, P^y, and P^z be the x-, y-, and z-components of the control points listed by columns, respectively:

$$P^x = (x_{1,1}, \ldots, x_{1,n}, x_{2,1}, \ldots, x_{2,n}, \ldots, x_{m,1}, \ldots x_{m,n})$$

$$P^y = (y_{1,1}, \ldots, y_{1,n}, y_{2,1}, \ldots, y_{2,n}, \ldots, y_{m,1}, \ldots, y_{m,n})$$

$$P^z = (z_{1,1}, \ldots, z_{1,n}, z_{2,1}, \ldots z_{2,n}, \ldots, z_{m,1}, \ldots, z_{m,n})$$

Then the bicubic B-spline surface $S(u,v)$ can be expressed as

$$S(u,v) = (x(u,v), y(u,v), z(u,v)) = (P^x, P^y, P^z) \begin{pmatrix} W & 0 & 0 \\ 0 & W & 0 \\ 0 & 0 & W \end{pmatrix} \quad (21)$$

where

$$W = (w_{1,3}(u)w_{1,3}(v), w_{1,3}(u)w_{2,3}(v), \ldots, w_{1,3}(u)w_{n,3}(v), \ldots \ldots, w_{m,3}(u)w_{1,3}(v), \ldots, w_{m,3}(u)w_{n,3}(v))^{\mathrm{T}} \quad (22)$$

Let $L = (A_L, B_L, C_L)$ and $U = (A_U, B_U, C_U)$ be the vectors of the lower bounds and upper bounds of the x-, y- and z-components of $S(u,v)$ at the interpolation points by columns, i.e.

$$A_L = (a_{1,1}, a_{1,2}, \ldots, a_{1,n}, \ldots, a_{m,1}, a_{m,2}, \ldots, a_{m,n})$$

$$A_U = (b_{1,1}, b_{1,2}, \ldots, b_{1,n}, \ldots, b_{m,1}, b_{m,2}, \ldots, b_{m,n})$$

where $a_{i,j}, b_{i,j}, \ldots$, are given in (17). Let Ω be a matrix of dimension $mn \times mn$ defined as follows:

$$\Omega = (W(u_1, v_1), W(u_1, v_2), \ldots, W(u_1, v_n), \ldots, W(u_m, v_1), W(u_m, v_2), \ldots, W(u_m, v_n))$$

where W is defined in (22). Then since the energy defined in (20) is actually a function of (P^x, P^y, P^z), the task to be solved here can be put in the following form:

$$\underbrace{\text{Minimize}}_{(P^x,P^y,P^z)\in R^{3mn}} \; E(P^x, P^y, P^z) \text{ subject to } L \leqslant (P^x, P^y, P^z) \begin{pmatrix} \Omega & 0 & 0 \\ 0 & \Omega & 0 \\ 0 & 0 & \Omega \end{pmatrix} \leqslant U \quad (23)$$

This is the minimization of a non-linear function subject to a set of linear constraints on the variables. It can be solved using the sequential quadratic programming (SQP) method[7,9,23]. The E04UCF subroutine of the *NAG Fortran Library*[23] is used here for the minimization process.

Hagen and Schulze[10] use a simplified form of the exact energy in the generation of smooth surfaces from a grid of 3D data points.

$$E(S) = \int\int_D \left[\frac{L^2G^2}{(EG)^{3/2}} + \frac{E^2N^2}{(EG)^{3/2}} + \frac{2M^2}{(EG)^{1/2}}\right] du\,dv \quad (24)$$

This is the result of assuming that S_u and S_v are orthogonal ($F = S_u \cdot S_v = 0$). The minimization process is based on a variational approach. This approach can be solved as a special case of (23).

In the past, since computing (20) directly was

Table 2 Computation time (in seconds) of the interproximating curves

Case	Energy form (2)	Energy form (3)	Energy form (5)	Energy form (6)
1(a)	6.410000	0.020000	0.100000	1.150000
1(b)	5.570000	0.050000	0.100000	1.160000
1(c)	6.320000	0.070000	0.110000	1.180000
2(a)	30.310000	0.070000	0.070000	9.700000
2(b)	26.780000	0.070000	0.080000	9.710000
2(c)	35.190000	0.080000	0.090000	9.730000
3(a)	41.770000	0.050000	0.070000	9.700000
3(b)	39.130000	0.050000	0.070000	9.740000
3(c)	38.280000	0.060000	0.080000	9.710000
4(a)	6.300000	0.070000	0.020000	1.140000
4(b)	6.220000	0.060000	0.040000	1.160000
4(c)	6.270000	0.060000	0.030000	1.140000

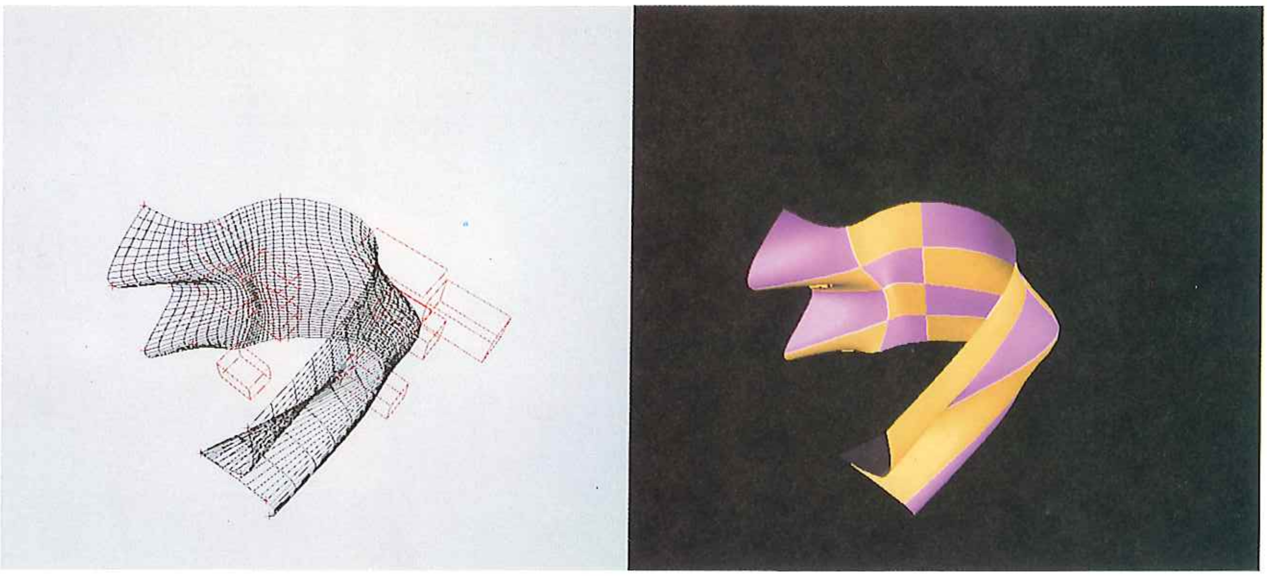

Figure 13 Case (a): combination of energy form (19) and average uniform model

quadratic programming technique (with constrained variable) to find the minimum energy surface. The subroutine E04NFF of the NAG Fortran Library is used for the minimization of (29).

5. IMPLEMENTATION

The implementation of the bicubic B-spline surface interproximation process is done on the same platform as the curve interproximation process. Both energy forms, (19) and (29), have been used in the energy minimization process. The matrix Q in (29) is set to an identity matrix which yields $|S_{uu}|^2 + |S_{uv}|^2 + |S_{vv}|^2$ as integrand for (29). The three parametrization techniques: the uniform model, the average centripetal model, and the average relative chord length model, have all been used in the construction of parameter knots (t_i) for each energy form. A number of test cases have been carried out. One test case is shown in *Figures 13–18* with *Case (a)* and *Case (b)* showing the combinations of the average uniform model with energy forms (19) and (29), respectively; *Case (c)* and *Case (d)* showing the combinations of the average centripetal model with energy forms (19) and (29), respectively; and *Case (e)* and *Case (f)* showing the combinations of the average relative chord length model with energy forms (19) and (29), respectively. The resulting surface for each combination is shown in both wire framed and shaded forms. The (exact) strain energies and computation times of the resulting surfaces are shown in *Table 3*.

It can be seen from *Figures 13–18* that the phenomena which held for interproximating curves hold for interproximating surfaces as well; the energy form has a much bigger impact on the shape and smoothness of the resulting surfaces than the parametrization technique, and the surfaces generated by the approximated energy form (29) tend to have sharper bumps. Therefore, for surface interproximation, one should also avoid using approximated energy forms unless sharper bumps and

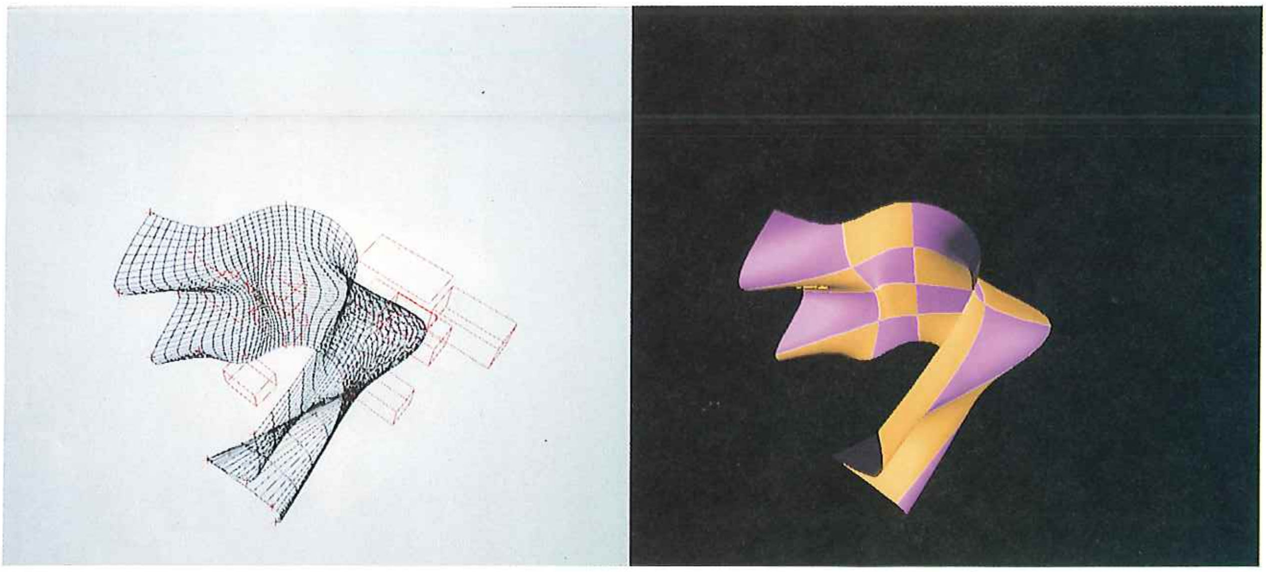

Figure 14 Case (b): combination of energy form (29) and average uniform model

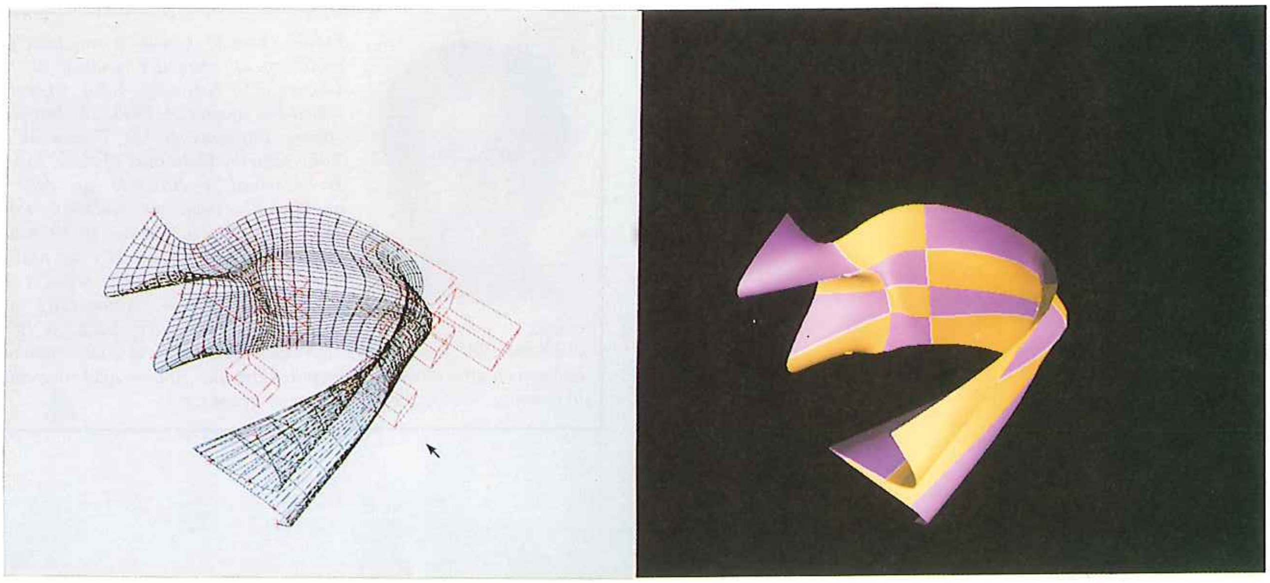

Figure 17 Case (e): combination of energy form (19) and average relative chord length model

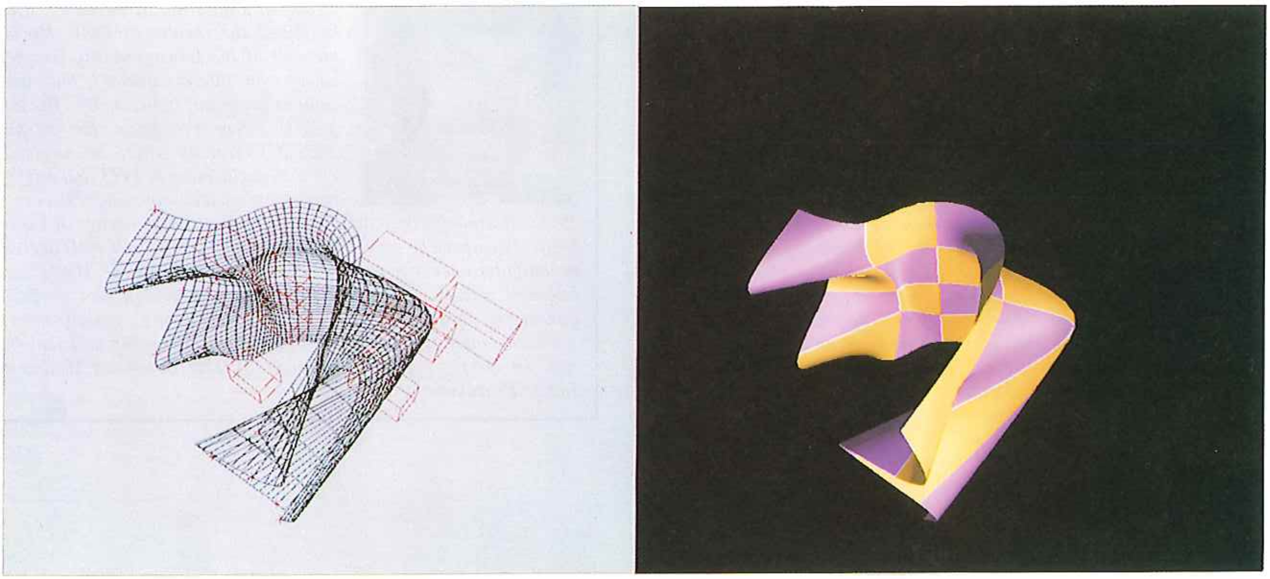

Figure 18 Case (f): combination of energy form (29) and average relative chord length model

ACKNOWLEDGEMENT

This work is supported by US NSF grant DMI-9400823.

REFERENCES

1. Ahlberg, J. H., Nilson, E. N. and Walsh, J. L., *The Theory of Splines and Their Applications*. Academic Press, New York, 1967.
2. Blanc, C. and Schlick, C., X Splines: a spline model designed for the end-user. *SIGGRAPH '95 Conference Proceedings*, Los Angeles, CA, 6–11 August 1995, pp. 377–386.
3. Cheng, C. and Zheng, Y. F., Thin plate spline surface approximation using Coons' patches. *Computer-Aided Geometric Design*, 1994, **11**, 269–287.
4. Cheng, F. and Barsky, B. A., Interproximation: interpolation and approximation using cubic spline curves. *Computer-Aided Design*, 1991, **10**, 700–706.
5. Cheng, F. and Barsky, B. A., Interproximation using cubic B-spline curves. In *Modeling in Computer Graphics*, eds. B. Falcidieno and T. L. Kunii. Springer-Verlag, Berlin, 1993, pp. 359–374.
6. de Boor, C., *A Practical Guide to Splines*. Springer-Verlag, New York, 1978.
7. Gill, P. E., Murray, W. and Wright, M. H., *Practical Optimization*. Academic Press, New York, 1981.
8. Faux, I. D. and Pratt, M. J., *Computational Geometry for Design and Manufacture*, 2nd Edition. Ellis Horwood, 1987.
9. Fletcher, R., *Practical Methods of Optimization*, 2nd Edition. Wiley, Chichester, 1987.
10. Hagen, H. and Schulze, G., Automatic smoothing with geometric surface patches. *Computer-Aided Geometric Design*, 1987, **4**, 231–236.
11. Halstead, M., Kass, M. and DeRose, T., Efficient, fair interpolation using Catmull–Clark surfaces. *SIGGRAPH '93*, August 1993, pp. 35–44.
12. Hosaka, M. and Kimura, K., Synthesis methods of curves and surfaces in CAD. In *IEEE Conference Proceedings: Interactive Techniques in Computer Aided Design*, Bologna, Italy, 1978, pp. 151–155.
13. Kallay, M., A method to approximate the space curve of minimal energy and prescribed length. *Computer-Aided Design*, 1987, **19**(2), 73–76.
14. Kallay, M. and Ravani, B., Optimal twist vectors as a tool for interpolating a network of curves with a minimum energy surface. *Computer-Aided Geometric Design*, 1990, **7**, 465–473.