

Fitting Parametric Curves to Dense and Noisy Points

A. Ardeshir Goshtasby

Abstract. Given a large set of irregularly spaced points in the plane, an algorithm for partitioning the points into subsets and fitting a parametric curve to each subset is described. The points could be measurements from a physical phenomenon, and the objective in this process could be to find patterns among the points and describe the phenomenon analytically. The points could be measurements from a geometric model, and the objective could be to reconstruct the model by a combination of parametric curves. The algorithm proposed here can be used in various applications, especially where given points are dense and noisy.

§1. Introduction

In many science and engineering problems there is a need to fit a curve or curves to an irregularly spaced set of points. Curve fitting has been studied extensively in Approximation Theory and Geometric Modeling, and there are numerous books on the subject [1,5,6,12,23]. Existing techniques typically find a single curve segment that approximates or interpolates the given points. Many techniques assume that the points are ordered and fit a curve to them by minimizing an error criterion [3,7,8,14,16,22,27,29,31,34]. If the points are ordered, piecewise polynomial curves can also be fitted to them [19,30]. Difficulties arise when the points are not ordered.

To fit curves to an irregularly spaced set of points, 1) the set should be partitioned into subsets, 2) the points in each subset should be ordered, and 3) a curve should be fitted to points in each subset. This paper will provide solutions to the first two problems; that is, partitioning a point set into subsets and ordering the points in each subset. Once the points in each subset are ordered, existing techniques can be used to find the curves.

Given a large set of irregularly spaced points in the plane, $\{\mathbf{p}_i = (x_i, y_i) : i = 1, \dots, N\}$, we would like to fit one or more parametric curves to the points, with the number of the curves to depend on the organization of the

points and the resolution of the representation. When fitting a parametric curve to an irregularly spaced set of points, the main problem is to find the nodes of the curve. The nodes of a parametric curve determine the adjacency relation between the points and order them. The curve will then approximate the points in the order specified. Methods to order sparse points [11,17,24] as well as dense points [25,26,32] have been developed. Existing methods, however, fit a single curve segment to an entire data set. Sometimes it is not desirable to fit a single curve segment to a large and complex point set, and it is necessary to represent the geometric structure present in the point set by many curve segments. In this paper it will be shown how to partition a point set into subsets and how to fit a parametric curve to each subset. A new method to order a set of dense and noisy points for curve fitting will also be presented.

In the proposed model, a radial field is centered at each point such that the strength of the field monotonically decreases as one moves away from the point. The sum of the fields has the averaging effect and reduces the effect of noise, and local maxima of the sum of the fields has the effect of tracing the spine of the points. Therefore, we will use the local maxima of the sum of the fields (the ridges of the obtained field surface) as an approximation to the curves to be determined. Based on the organization of the points, disjoint ridges may be obtained, each suggesting a curve. The ridges will be used to partition the points into subsets and fit a curve to each subset. In the following, the steps of this process are described in detail.

§2. Approach

A desirable property of an approximating curve is for it to pass as close as possible to the given points while providing a certain smoothness appearance. For a dense point set, the curve cannot pass close to all the points, so it is desired that the curve trace the spine of the points. In the model proposed here, an initial estimation to a curve is obtained by taking points in the xy plane whose sum of inverse distances to the given points is locally maximum. That is, if the sum of inverse distances of point (x, y) to given points $\{(x_i, y_i) : i = 1, \dots, N\}$ is larger than the sum of inverse distances of points in the neighborhood of (x, y) to the given points, then point (x, y) is considered an initial estimation to a point on the curve. Therefore, by tracing points in the xy plane that locally maximize

$$f(x, y) = \sum_{i=1}^N [(x - x_i)^2 + (y - y_i)^2 + 1]^{-\frac{1}{2}}, \quad (1)$$

we find an approximation to the curves we want to find.

The function f can also be interpreted as follows: Suppose a radial field of strength 1 is centered at point (x_i, y_i) , $i = 1, \dots, N$, such that the strength of the field decreases with inverse distance as one moves away from the point. Then, the strength of the field at point (x, y) will be $[(x - x_i)^2 + (y - y_i)^2]^{-\frac{1}{2}}$,

and the curves to be found can be considered points in the xy plane whose sum of field values are locally maximum.

Once a set of points is given, the function f becomes fixed, and the obtained ridges will have a fixed shape. In order to have control over the shape or smoothness of obtained ridges, we revise formula (1) as follows. If instead of inverse distances defined by $[(x - x_i)^2 + (y - y_i)^2 + 1]^{-\frac{1}{2}}$, we use

$$[(x - x_i)^2 + (y - y_i)^2 + r^2]^{-\frac{1}{2}} \quad (2)$$

in equation (1), we obtain

$$g(x, y) = \sum_{i=1}^N [(x - x_i)^2 + (y - y_i)^2 + r^2]^{-\frac{1}{2}}. \quad (3)$$

The basis functions defined by (2) are known as inverse multiquadrics [13]. The parameter r of the basis functions can be varied to generate different surfaces [21]. Figure 1b shows the field surface obtained when using the points of Fig. 1a and inverse multiquadric basis functions with $r = 5$.

Instead of inverse multiquadric basis functions, other radial basis functions [2,4,10,28,33,35] also can be used to define function g . The choice of the basis functions influences the shape of the obtained field surface, the shape of the obtained ridges, and, consequently, the shape of the obtained curves.

By tracing the local maxima of the field surface g in the xy plane, we will obtain an approximation to the curves. Parameter r changes the shape of the basis functions and affects the shape of the field surface.

Local maxima of surface g can result in structures that contain branches and loops. The proposed model, therefore, can recover very complex patterns in dense and noisy point sets. Note also that the proposed method does not require any knowledge about the adjacency relation between the points. This method, in fact, provides the means to determine the adjacency relation between the points.

§3. Implementation

Derivation of an analytic formula that represents the local maxima of the surface g may not be possible. Digital approximation to the local maxima, however, is possible. This approximation is found in the form of digital contours and is used to partition the points into subsets. To digitally trace surface ridges, the surface is digitized into a digital image. The digitization process involves starting from $x = x_{min}$ and $y = y_{min}$ and incrementing x and y by some small increment δ until reaching $x = x_{max}$ and $y = y_{max}$. For each discrete (x, y) , the value for $g(x, y)$ is then found from formula (3). x_{min} and x_{max} could be the smallest and largest x coordinates, and y_{min} and y_{max} could be the smallest and largest y coordinates of the given points. The parameter δ is used as the increment for both x and y because radially symmetric basis

functions are used to define g . This parameter determines the resolution of the obtained image. For a finer resolution, this parameter should be reduced, while for a coarser resolution this parameter should be increased. If this parameter is to be chosen automatically, it should be selected such that most given points map to unique pixels in the obtained image.

Digitizing the surface g in this manner will result in a digital image whose pixel values show uniform samples from surface g . Figure 1b shows digitization of a field surface into an image of 256×256 pixels. To find the image ridges, pixels with locally maximum intensities are located. To find locally maximum image intensities, the gradient magnitude and the gradient direction [20] of the image at each pixel are determined. The gradient direction at a pixel is the direction at which change in intensity at the pixel is maximum, and gradient magnitude is the magnitude of the intensity change in the gradient direction at the pixel.

To find the ridges, we find each pixel A in the image where two pixels B and C that are adjacent to it and are at its opposite sides have intensities that are smaller than that at A . Assuming that the image obtained after digitizing surface g is represented by I , we mark the pixel at (i, j) as A if one of the following is true:

$$I(i-1, j) < I(i, j) \quad \& \quad I(i+1, j) < I(i, j); \quad (4)$$

$$I(i, j-1) < I(i, j) \quad \& \quad I(i, j+1) < I(i, j); \quad (5)$$

$$I(i-1, j-1) < I(i, j) \quad \& \quad I(i+1, j+1) < I(i, j); \quad (6)$$

$$I(i-1, j+1) < I(i, j) \quad \& \quad I(i+1, j-1) < I(i, j). \quad (7)$$

Using the image of Fig. 1b, we find that pixels in the contours shown in Fig. 1c are marked as A . We will call the contours obtained in this manner the **minor ridges** of the image. Next, we find each pixel D whose value is not only larger than those of B and C adjacent to it and at its opposite sides, but which also has a gradient direction that is the same as the direction obtained by connecting pixels B and C . The gradient direction at a pixel is quantized with 45-degree steps to ensure that only directions that are possible to obtain when connecting pixels B and C in an image are obtained. The pixels marked as D are shown in Fig. 1d. We will call these contours the **major ridges** of the image. As can be observed, major ridges are a subset of minor ridges. We also see that major ridge points do not fall on small and noisy branches of the minor ridges but rather fall on contours that represent the spines of the points. If the minor ridges are cut at the branch points, and branches that do not contain a major ridge point are removed, and the remaining contours are thinned, we obtain Fig. 1e. The obtained contours will be called the **local-maxima contours**, or simply the **contours**. These contours will be taken as approximations to the curves to be found. We will use them not only to partition the points into subsets but also to order the points in the subsets.

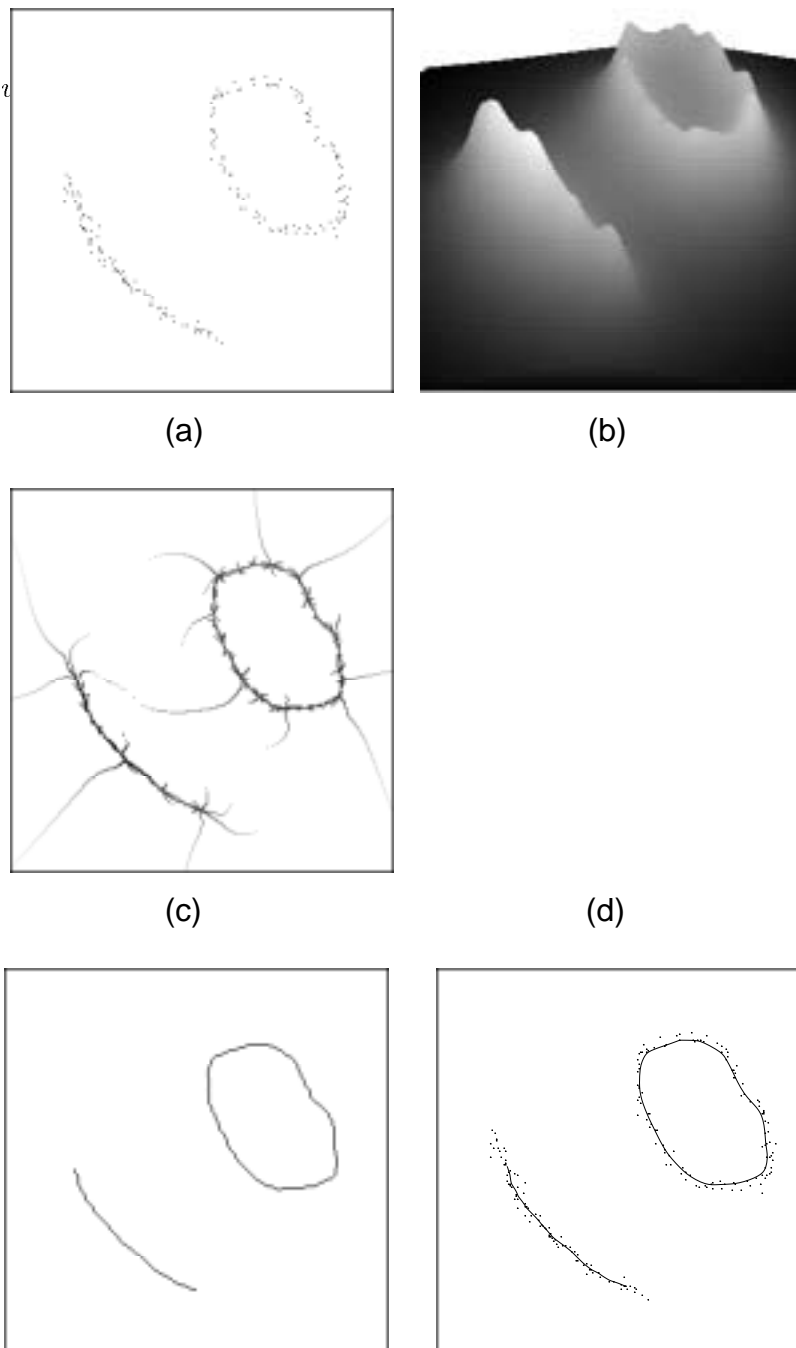


Fig. 1. (a) An irregularly spaced set of points. (b) A digitized field surface. (c) Contours representing the minor ridges. (d) Contours representing the major ridges. (e) Local-maxima contours. (f) RaG curves with $\sigma = 0.04$ approximating points shown in (a).

§4. Node Estimation

The method outlined in the preceding section determines contours that are approximations to the curves to be found. These contours will be used to partition a point set into subsets and order the points in each subset.

Suppose a point set has produced m contours; then, a point is assigned to contour j ($1 \leq j \leq m$) if it is closest to a pixel in contour j than to a pixel in any other contour. In this manner, a point is assigned to one of m contours. This process, when completed, will partition a point set into m subsets by

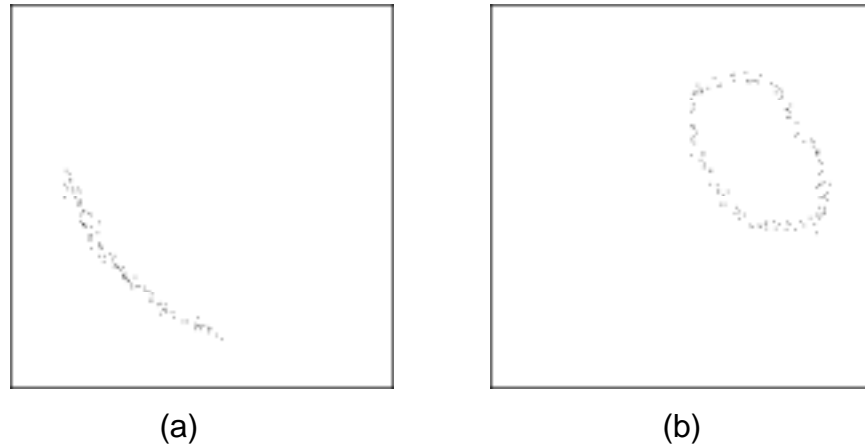


Fig. 2. (a), (b) Two point subsets obtained from the point set of Fig. 1a.

assigning the points into one of m contours. Figures 2a and 2b show the point subsets obtained in this manner from the point set of Fig. 1a.

To order points $\{\mathbf{q}_i : i = 1, \dots, n\}$ in subset j , for each point \mathbf{q}_i a point in contour j that is closest to it is determined. We call the obtained contour point the **projection** of point \mathbf{q}_i . After determining projections of all points in the subset to the contour, the contour is traced from one end to the other, and in the order the projections are visited, the associated points are ordered.

Since the contours are approximations to the curves to be found, the contour length from a projection to the start of the contour is divided by the length of the contour to obtain an arc-length estimation to the node of the point. If the contour is closed, an arbitrary point on the contour is taken as the start point. If the contour is open, one of the end points is taken as the start point.

The size of the image obtained by digitizing surface g determines the accuracy of the obtained nodes. If the surface g is very coarsely digitized, the obtained contours will be very short, and numerous points may produce the same node, especially when given points are dense. To provide a more accurate node estimation, the surface g should be digitized into an image large enough to produce unique nodes.

Once the coordinates of given points and the associated nodes are known, a parametric curve can be fitted to the points by one of the existing methods [9,11,16,18,30]. Fitting rational Gaussian (RaG) curves [9] to the points shown in Fig. 1a with nodes as determined above, we obtain the curves shown in Fig. 1f. The curves are overlaid with the original points to show the quality of the curve fitting. Note that these curves were obtained using the points in Fig. 1a and not the contour points in Fig. 1e. The contour points were used only to partition a point set into subsets and to determine the nodes of the points.

§5. Observations

To observe the behavior of the proposed curve-fitting method, results on three additional point sets are shown in Fig. 3. Figure 3a shows noisy points along

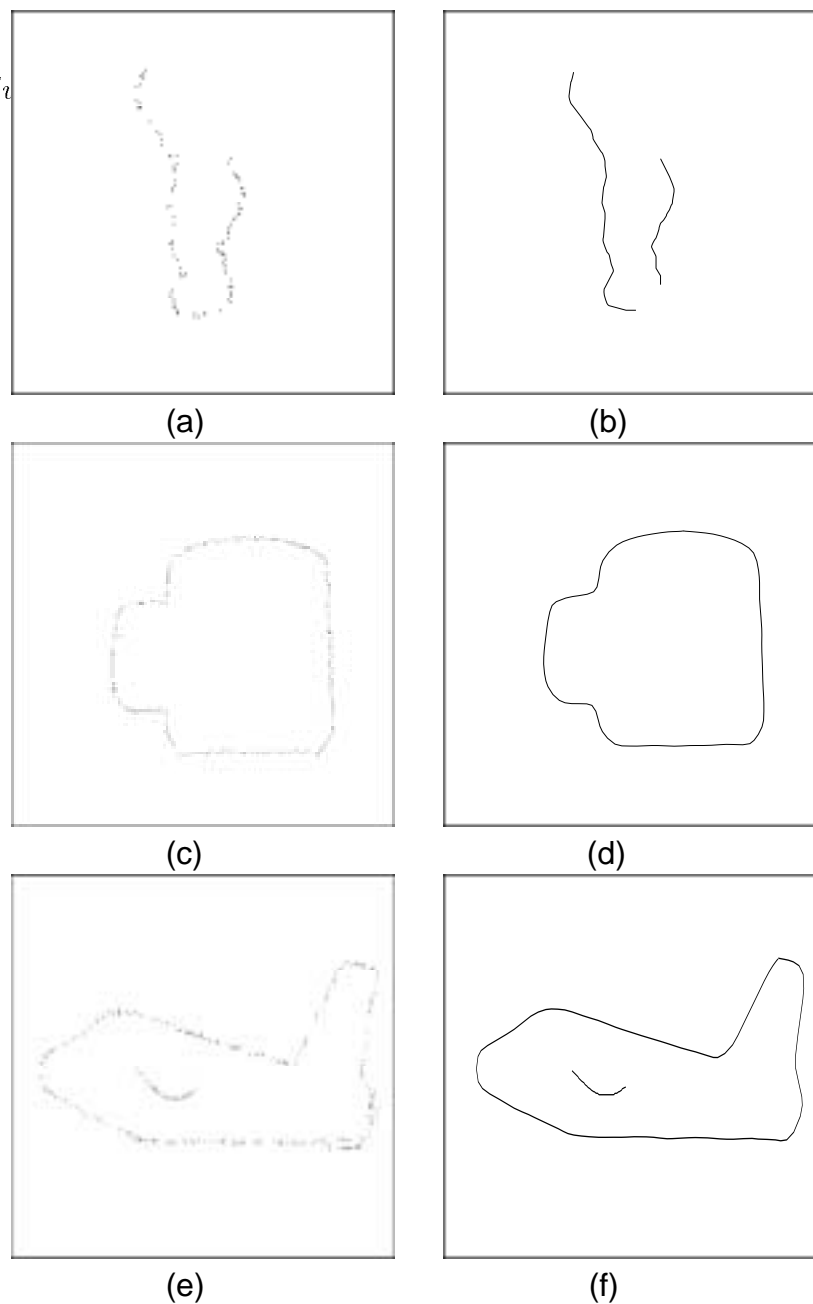


Fig. 3. A few curve-fitting examples.

an open contour, Fig. 3c shows a dense and noisy point set along the silhouette of a coffee mug, Fig. 3e shows irregularly spaced points along the silhouette of a model plane and one of its wings. We can see the geometric structures in these point sets and, if asked, can trace the structures manually without any difficulty. The algorithm proposed here is intended to do the same. The curves obtained are shown in Figs. 3b, 3d, and 3f.

The point sets shown in Fig. 3 did not contain geometric structures with branches and loops. If a point set contains branches and loops, the local-maxima contours will also contain branches and loops. A single curve segment, however, cannot represent branching structures. The solution we propose is to segment a complex contour into simple ones by cutting it at the branch points and fitting a curve to each branch.

§6. Summary and Conclusions

A large number of techniques for fitting parametric curves to irregularly spaced points have been developed. These techniques fit a single curve to the given points and often require that the points be ordered. In science and engineering problems that deal with measurement data, the given points may not be ordered and they may contain noise. Moreover, it may not be appropriate to fit a single curve segment to all the points. In this paper, a method to partition a point set into subsets and fit a parametric curve to each subset was described. The proposed method has the ability to take into consideration the noisiness and denseness of a point set when obtaining the curves.

Also introduced was a method to determine the nodes of a parametric curve that approximates a set of dense and noisy points. The proposed method provides the means to fit any parametric curve, including B-Splines and Non-Uniform Rational B-Splines, to irregularly spaced points. Although in this paper only inverse multiquadrics were used as basis functions to obtain a field surface, from which the curve segments were determined, other radial basis functions [33] can be used in the same manner. Depending on the parametric curve formulation and the radial basis functions used, the number and the shapes of the curves fitting to a set of points may vary.

Acknowledgments. This work was supported in part by a grant from the National Science Foundation: IRI-9529045.

References

1. Beach, R. C., *An Introduction to the Curves and Surfaces of Computer-Aided Design*, Van Nostrand, New York, 1991.
2. Buhmann, M. D., Multivariate cardinal interpolation with radial basis functions., *Constr. Approx.* **6** (1990), 225–255.
3. Cohen, E. and C. L. O'Dell, A data dependent parametrization for spline approximation, in *Mathematical Methods in Computer Aided Geometric Design*, T. Lyche and L. L. Schumaker (eds.), Vanderbilt Univ. Press, 1989, 155–166.
4. Dyn, N., Interpolation of scattered data by radial functions, in *Topics in Multivariate Approximation*, Academic Press, 1987, 47–61.
5. Farin, G., *Curves and Surfaces for Computer Aided Geometric Design*, Academic Press, New York, 1988.
6. Faux, I. D. and M. J. Pratt, *Computational Geometry for Design and Manufacture*, Ellis Horwood, Chichester, 1979.
7. Foley, T. A. and G. M. Nielson, Knot selection for parametric spline interpolation, in *Mathematical Methods in Computer Aided Geometric Design*, T. Lyche and L. L. Schumaker (eds.), Academic Press, New York, 1989, 261–271.

8. Fritsch, F. N. and R. E. Carlson, Monotone piecewise cubic interpolation, *SIAM J. Numer. Anal.* **17**, 2 (1980), 238–246.
9. Goshtasby, A., Geometric modeling using rational Gaussian curves and surfaces, *Computer-Aided Design* **27**, 5 (1995), 363–375.
10. Goshtasby, A. and W. D. O’Neill, Surface fitting to scattered data by a sum of Gaussians, *Comput. Aided Geom. Design* **10** (1993), 143–156.
11. Grossman, M., Parametric curve fitting, *Computer J.* **14**, 2 (1970), 169–172.
12. Hagen, H., *Curve and Surface Design*, SIAM, 1992.
13. Hardy, R. L., Theory and applications of the multiquadrics–biharmonic method, *Comput. Math. Appl.* **19**, 8/9 (1990), 163–208.
14. Hartley, P. J. and C. J. Judd, Parametrization and shape of B-spline curves for CAD, *Computer-Aided Design* **12**, 5 (1980), 235–238.
15. Hölzle, G. E., Knot placement for piecewise polynomial approximation of curves, *Computer-Aided Design* **15**, 5 (1983), 295–296.
16. Hoschek, J., Approximate conversion of spline curves, *Comput. Aided Geom. Design* **4** (1987), 59–66.
17. Hoschek, J., Intrinsic parametrization for approximation, *Comput. Aided Geom. Design* **5** (1988), 27–31.
18. Hoschek, J., Spline approximation of offset curves, *Comput. Aided Geom. Design* **5** (1988), 33–40.
19. Ichida, K. and T. Kiyono, Curve fitting by a one-pass method with a piecewise cubic polynomial, *ACM Trans. Math. Software* **3**, 2 (1977), 164–174.
20. Jain, A. K., *Fundamentals of Digital Image Processing*, Prentice Hall, 1989, 347–349.
21. Kansa, E. J., Multiquadrics—A scattered data approximation scheme with applications to computational fluid dynamics—I, *Comput. Math. Appl.* **19**, 8/9 (1990), 127–145.
22. Kosters, M., Curvature-dependent parametrization of curves and surfaces, *Computer-Aided Design* **23**, 8 (1991), 569–578.
23. Lancaster, P. and K. Salkauskas, *Curve and Surface Fitting*, Academic Press, New York, 1986.
24. Lee, E. T. Y., Choosing nodes in parametric curve interpolation, *Computer-Aided Design* **6** (1989), 363–370.
25. Lee, I.-K., Curve reconstruction from unorganized points, *Comput. Aided Geom. Design* **17** (2000), 161–177.
26. Levin, D., The approximation power of moving least-squares, *Math. Comp.* **67** (1998), 1517–1531.
27. Marin, S. P., An approach to data parametrization in parametric cubic spline interpolation problem, *J. Approx. Theory* **41** (1984), 64–86.

28. Meinguet, J., An intrinsic approach to multivariate spline interpolation at arbitrary points, *Polynomial and Spline Approximation*, B. N. Sahney (ed.), D. Reidel Publishing, 1979, 163–190.
29. Mullineux, M., Approximating shapes using parametrized curves, *IMA J. Applied Mathematics* **29** (1982), 203–220.
30. Piegl, L., A technique for smoothing scattered data with conic sections, *Computers in Industry* **9** (1987), 223–237.
31. Plass, M. and M. Stone, Curve-fitting with piecewise parametric cubics, *Computer Graphics* **17**, 3 (1983), 229–239.
32. Pottmann, H. and T. Randrup, Rotational and helical surface approximation for reverse engineering, *Computing* **60** (1998), 307–322.
33. Powell, M. J. D., Radial basis functions for multivariable interpolation: A review, *Algorithms for Approximation*, J. C. Mason and M. G. Cox (eds.), Clarendon Press, Oxford, 1987, 143–167.
34. Sarkar, B. and C-H Menq, Parameter optimization in approximating curves and surfaces in measurement data, *Comput. Aided Geom. Design* **8** (1991), 267–290.
35. Schagen, I. P., The use of stochastic processes in interpolation and approximation, *Int. J. Computer Math., Section B*, **8** (1980), 63–76.

A. Ardeshir Goshtasby
Department of Computer Science and Engineering
Wright State University
Dayton, OH 45435, USA
agoshtas@cs.wright.edu
<http://www.cs.wright.edu/~agoshtas/>