John L. Junkins*
James R. Jancaitis
University of Virginia
Charlottsville, VA. 22901
Gary W. Miller
McDonnell Douglas Astro. Co.
Huntington Beach, CA. 92647

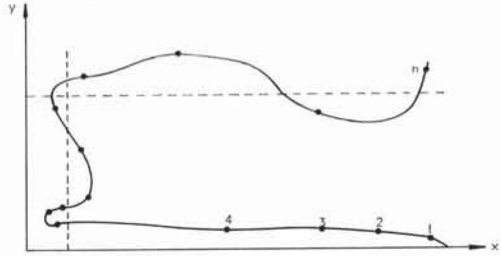

FIG. 1. A discretely sampled arbitrary planar curve.

Smooth Irregular Curves

A sequential approximation and interpolation procedure for computer analysis is applicable to contour lines.

Problem Statement and Discussion

GIVEN n arbitrary real data points,

$$\{x_1, y_1; x_2, y_2; \cdots; x_n, y_n\} \tag{1}$$

assumed to lie sequentially along some unknown planar curve. The objective is to obtain efficiently a reasonable approximation of the unknown curve. Prior to discussing the details of the method developed, it is useful to take a qualitative look

ABSTRACT: *An efficient procedure is presented for determining a smooth curve (with no value or slope discontinuities) through an arbitrary set of points in a plane. The method was developed specifically for, but is not restricted to, computer determination of smooth and accurate contour lines. A family of locally valid functions constrained to connect smoothly at the data points are sequentially determined. A new independent variable is introduced and employed in the interpolation equations. This independent variable is a "pseudo arc-length" which usually approximates true arc-length well and is increasing along the curve. This independent variable is shown to be well suited for accurate fitting of irregular contours and provides a direct control over the density of solved points, thus reliably yielding smooth computer plotted contours. The procedure is applicable to open or closed curves; further, the curves determined can even cross themselves as do contours for overhanging cliffs. Several examples are provided to demonstrate the practical use and efficiency of the method.*

at the problem. Consider the sketch in Figure 1. In the most general case, neither x nor y will be monotonically increasing. This results in y (in general) being a multivalued function of x and vice-versa; thus precluding the direct application of classical one dimensional interpolation procedures (e.g., Lagrangian interpolation,[1] Spline Function interpolation[2] or any other simple fit of y as a function of x or vice-versa).

* This study was conducted under U.S.A. TOPOCOM Contract DACA 71-71-0802; it was one of several ideas included in "Mathematical Terrain Analysis" presented at the Annual Convention of the American Society of Photogrammetry in Washington, D. C., March 1971.

After giving some thought to this problem, it was noted that an ideal choice of an independent variable for interpolation would be arc length along the curve because length is monotonically increasing along the curve and has a unique value at every point. Unfortunately, the objective curve is unknown; arc length along it is therefore difficult to measure! However, advantage can be taken of the knowledge that the n data points (Expression 1) lie on the unknown objective curve to define an approximation to arc length. A *pseudo-arc-length* is defined explicitly below and has been found to be an excellent independent variable for interpolation formulas.

Development of the Method

The method considers a local subset of six points to define sequentially a local polynomial approximation to the curve between Points 3 and 4 of the local subset. After each stage of the procedure, a new point is added and the first one deleted to define a new subset of six points. The family of locally valid polynomials are determined in such a manner that each connects at its bounding data points with no slope discontinuity with the polynomials approximating either the preceding or succeeding segment. The decision to consider sequentially six points is subjective and is not central to the method developed. Analogous processes can be developed using an arbitrary choice for the number of points in the local subset.

Prior to discussing the details of the fitting procedure, the independent variable to be employed will be defined. Consider the first six of n data points as depicted in Figure 2. Define a *cumulative polygon approximation to arc length* as

$$\left.\begin{aligned} s_1 &= 0 \\ s_k &= s_{k-1} + [(x_k - x_{k-1})^2 + (y_k - y_{k-1})^2]^{1/2}; \; k = 2, 3, \cdots, 6. \end{aligned}\right\} \qquad (2)$$

for each point in the data subset; at any x, y along the unknown curve between two of the data points (say points k and $k+1$) s is measured along the straight line between (x_k, y_k) and (x_{k+1}, y_{k+1}). Clearly, s so defined is single-valued and smoothly varying from s_1 to s_n. By choosing a power series of the form

$$\begin{aligned} x(s) &= \sum_{i=1}^{n} A_i S^{i-1} \\ y(w) &= \sum_{i=1}^{n} B_i S^{i-1} \end{aligned} \qquad (3)$$

as the parametric polynomials to approximate the unknown curve, we insure that every value of s from s_1 to s_n yields a unique pair of (x, y) values. A *left, middle* and *right* averaging procedure is developed next.

The fundamental ideas underlying the approximation method are illustrated in

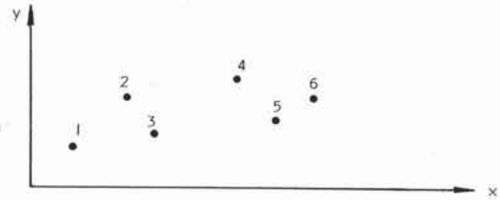

FIG. 2. The first six data points.

Figure 3. The procedure makes use of three overlapping cubic fits of x and y as functions of pseudo-arc-length (i.e., the *left*, *middle*, and *right* fits shown in Figures 3a, b, c). The left L, middle M, and right R cubic approximations are obtained by fitting

$$\begin{aligned} x &= A_1 + A_2 s + A_3 s^2 + A_4 s^3 \\ y &= B_1 + B_2 s + B_3 s^2 + B_4 s^3 \end{aligned} \tag{4}$$

to the data

$$(s, x, y)_L = \{0, x_1, y_1; s_2, x_2, y_2; s_3, x_3, y_3; s_4, x_4, y_4\} \tag{5}$$

$$(s, x, y)_M = \{s_2, x_2, y_2; s_3, x_3, y_3; s_4, x_4, y_4; s_5, x_5, y_5\} \tag{6}$$

$$(s, x, y)_R = \{s_3, x_3, y_3; s_4, x_4, y_4; s_5, x_5, y_5; s_6, x_6, y_6\} \tag{7}$$

respectively. The left L, middle, and right R coefficients are computed as

$$\boldsymbol{A}_L = \boldsymbol{S}_L^{-1}\boldsymbol{X}_L, \qquad \boldsymbol{B}_L = \boldsymbol{S}_L^{-1}\boldsymbol{Y}_L \tag{8)*$$

$$\boldsymbol{A}_M = \boldsymbol{S}_M^{-1}\boldsymbol{X}_M, \qquad \boldsymbol{B}_M = \boldsymbol{S}_M^{-1}\boldsymbol{Y}_M \tag{9)*$$

$$\boldsymbol{A}_R = \boldsymbol{S}_R^{-1}\boldsymbol{X}_R, \qquad \boldsymbol{B}_R = \boldsymbol{S}_R^{-1}\boldsymbol{Y}_R \tag{10)*$$

where the L, M, and R coefficient matrices are defined as

$$\boldsymbol{A}_L = \begin{bmatrix} A_{1L} \\ A_{2L} \\ A_{3L} \\ A_{4L} \end{bmatrix}, \qquad \boldsymbol{B}_L = \begin{bmatrix} B_{1L} \\ B_{2L} \\ B_{3L} \\ B_{4L} \end{bmatrix}, \boldsymbol{L} \to \boldsymbol{M}, \boldsymbol{R}. \tag{11}$$

The L, M, and R data matrices are defined as*

$$\boldsymbol{X}_L^T = [x_1 x_2 x_3 x_4]; \; x \to y \tag{12}$$

$$\boldsymbol{X}_M^T = [x_2 x_3 x_4 x_5]; \; x \to y \tag{13}$$

$$\boldsymbol{X}_R^T = [x_3 x_4 x_5 x_6]; \; x \to y \tag{14}$$

and the resulting coefficient matrices to be inverted in Equations 8, 9 and 10 are

$$\boldsymbol{S}_L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & s_2 & s_2^2 & s_2^3 \\ 1 & s_3 & s_3^2 & s_3^3 \\ 1 & s_4 & s_4^2 & s_4^3 \end{bmatrix} \tag{15}$$

$$\boldsymbol{S}_M = \begin{bmatrix} 1 & s_2 & s_2^2 & s_2^3 \\ 1 & s_3 & s_3^2 & s_3^3 \\ 1 & s_4 & s_4^2 & s_4^3 \\ 1 & s_5 & s_5^2 & s_5^3 \end{bmatrix} \tag{16}$$

* The inverses $\boldsymbol{S}_L^{-1}$, $\boldsymbol{S}_M^{-1}$, $\boldsymbol{S}_R^{-1}$ can be obtained analytically and are special cases of the classical Lagrangian interpolation formula[1]. Note that the same matrix inverted for the x-coefficient is re-used to compute the y-coefficients. This is a key factor contributing to the efficiency of the method.

$$S_R = \begin{bmatrix} 1 & s_3 & s_3^2 & s_3^3 \\ 1 & s_4 & s_4^2 & s_4^3 \\ 1 & s_5 & s_5^2 & s_5^3 \\ 1 & s_6 & s_6^2 & s_6^3 \end{bmatrix} \tag{17}$$

where the s_i are evaluated according to Equation 2.

The final approximation for the unknown curve between Points 3 and 4 of the subset makes use of the preliminary cubic approximations to obtain *reasonable* final slopes at Points 3 and 4. To obtain these objective slopes at Point 3, the left and middle slopes are evaluated at Point 3 and averaged as

$$\left.\overline{\frac{dx}{ds}}\right|_{s_3} = \frac{1}{2}\left[\left.\frac{dx_L}{ds}\right|_{s_3} + \left.\frac{dx_M}{ds}\right|_{s_3}\right] \tag{18}$$

$$\left.\overline{\frac{dy}{ds}}\right|_{s_3} = \frac{1}{2}\left[\left.\frac{dy_L}{ds}\right|_{s_3} + \left.\frac{dy_M}{ds}\right|_{s_3}\right]. \tag{19}$$

Similarly, the middle and right slopes are averaged at Point 4 to obtain

$$\left.\overline{\frac{dx}{ds}}\right|_{s_4} = \frac{1}{2}\left[\left.\frac{dx_M}{ds}\right|_{s_4} + \left.\frac{dx_R}{ds}\right|_{s_4}\right] \tag{20}$$

$$\left.\overline{\frac{dy}{ds}}\right|_{s_4} = \frac{1}{2}\left[\left.\frac{dy_M}{ds}\right|_{s_4} + \left.\frac{dy_R}{ds}\right|_{s_4}\right]. \tag{21}$$

It is desired that the final approximation to the $3 \rightarrow 4$ segment pass through points 2, 3, 4, and 5 and satisfy average slope requirements (Equations 18 to 21) at Points 3 and 4. The lowest degree polynomials in s that can satisfy these conditions is a quintic. Thus, the final approximation to the $3 \rightarrow 4$ segment is obtained by fitting

$$x = A_1 + A_2 s + A_3 s^2 + A_4 s^3 + A_5 s^4 + A_6 s^5 \tag{22}$$

$$y = B_1 + B_2 s + B_2 s^2 + B_4 s^3 + B_5 s^4 + B_6 s^5. \tag{23}$$

The coefficients are obtained as

$$\boldsymbol{A} = \boldsymbol{S}^{-1}\boldsymbol{X}, \qquad \boldsymbol{B} = \boldsymbol{S}^{-1}\boldsymbol{Y} \tag{24}$$

where

$$\boldsymbol{A}^T = [A_1 A_2 A_3 A_4 A_5 A_6] \tag{25}$$

$$\boldsymbol{B}^T = [B_1 B_2 B_3 B_4 B_5 B_6] \tag{26}$$

$$\boldsymbol{X}^T = \left[x_2 x_3 x_4 x_5 \left.\overline{\frac{dx}{ds}}\right|_3 \left.\overline{\frac{dx}{ds}}\right|_4\right] \tag{27}$$

$$\boldsymbol{Y}^T = \left[y_2 y_3 y_4 y_5 \left.\overline{\frac{dy}{ds}}\right|_3 \left.\overline{\frac{dy}{ds}}\right|_4\right] \tag{28}$$

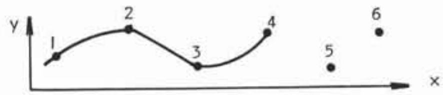

FIG. 3a

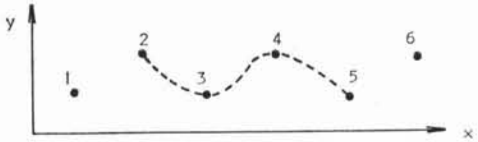

FIG. 3b

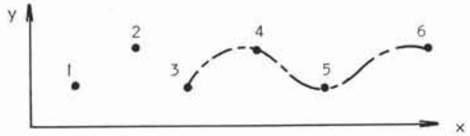

FIG. 3c

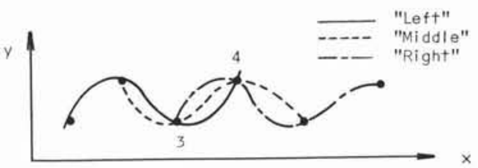

FIG. 3d

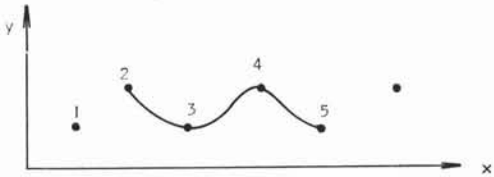

FIG. 3e

FIG. 3. The preliminary and final approximations. *3a*—The preliminary *left* approximation for the segment from Point 2 to 3. *3b*—The preliminary *middle* approximation for the segment from Point 3 to 4. *3c*—The preliminary *right* approximation for segment from Point 4 to 5. *3d*—The slope averaging concept at Points 3 and 4. The final slope at Point 3 is equal to (*left* slope at 3 plus *middle* slope at 3); the final slope at Point 4 is equal to $\frac{1}{2}$(*middle* slope at 4 plus *right* slope at 4). *3e*—The final approximation for the segment from Point 3 to 4.

and

$$S \equiv \begin{bmatrix} 1 & s_2 & s_2^2 & s_2^3 & s_2^4 & s_2^5 \\ 1 & s_3 & s_3^2 & s_3^3 & s_3^4 & s_3^5 \\ 1 & s_4 & s_4^2 & s_4^3 & s_4^4 & s_4^5 \\ 1 & s_5 & s_5^2 & s_5^3 & s_5^4 & s_5^5 \\ 0 & 1 & 2s_3 & 3s_3^2 & 4s_3^3 & 5s_3^4 \\ 0 & 1 & 2s_4 & 3s_4^2 & 4s_4^3 & 5s_4^4 \end{bmatrix}. \tag{29}$$

The final approximation Equations 22 and 23 can be used for interpolation at any desired interval to estimate points between Points 3 and 4.

We can now add the 7th x, y data point to the original six and delete the original first point. The same procedure can be applied to determine an approximation between Points 4 and 5 of the original data points (Points 3 and 4 of the new subset); but now we can proceed directly to the *right* approximation step (beginning with Equations 10), because the old *middle* approximation becomes the new *left* and the old *right* approximation becomes the new *middle*. The resulting final approximation between Points 4 and 5 will join smoothly with the previously determined final approximation to the curve between Points 3 and 4. This is true because both final approximations were constrained to the same slope at Point 4. All subsequent approximations follow by analogy, each new data point incorporated will require only the computation of *right* and *final* coefficients (Equations 10 and 24).

Applications

A Fortran IV subroutine (*SMOOTH*) has been developed based on this approach and has been used for interpolation between discrete points along a variety of irregular curves, including altitude contours. In order to maximize the efficiency and generality of this algorithm, four useful bits of logic were incorporated into SMOOTH, as explained below.

1. SMOOTH automatically transfers input points into the output array if these points are separated by less than the desired interpolation (plotting) interval.

2. As the algorithm (as explained above) begins producing points for plotting (for example) at the 3rd original data point and the last point for plotting would be the n-3 point, logic was introduced to make the representation *complete:*

(a) For *closed* curves—If the n original points are *closed* (as evidenced by the n-th point being equal to the first) then the 2nd through the 5th points are added to the end of the set. This has the effect of *closing the curve* so that SMOOTH output points will begin and end at the 3rd data point with no slope discontinuity.

(b) For *open curves*—Two new points a, b are introduced (Figure 4) between the first two data points in the set. These points a, b are selected to lie along the straight line connecting the first two points and to be separated by 1/10 the desired interpolation interval. This has the effect of causing the output points to begin at Point b (which is indistinguishable from Point 1) instead of Point 3. Similarly, two points c, d are artificially introduced adjacent to the final point along the straight line connecting the final two points.

3. As the arc length increases around the curve, the elements of Equation 24 tend to have an increasingly wide numerical range. As this tendency leads to poor accuracy (and eventually, singularity), it was necessary to circumvent this difficulty. The solution was relatively simple, the origin (the point where $s=0$) is simply redefined to be the first point of each subset (for that subset's *right* and *final* fits).

A simplified logic flow diagram for SMOOTH is given in Figure 5. This logic flow diagram should be correlated with the foregoing development in order to fully understand SMOOTH. To prevent excessive clutter of Figure 5, the logic for detecting dense input points (closer than the objective interpolation interval), directly outputing them, and subsequent restarting of the SMOOTH procedure is deleted.

To provide a basis for practical appreciation of the SMOOTH interpolation procedure, three applications are shown in Figures 6, 7, and 8. The first example uses as input points the four corners and mid-points of each side of a square (connected by straight lines in Figure 6-*left*). As is illustrated by the SMOOTHed points plotted in Figure 6-*right*, symmetric input data does yield a smooth, symmetric set of interpolated points.

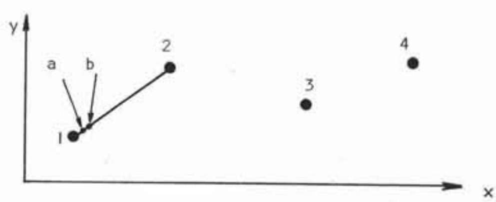

FIG. 4. Initializing points for open curves.

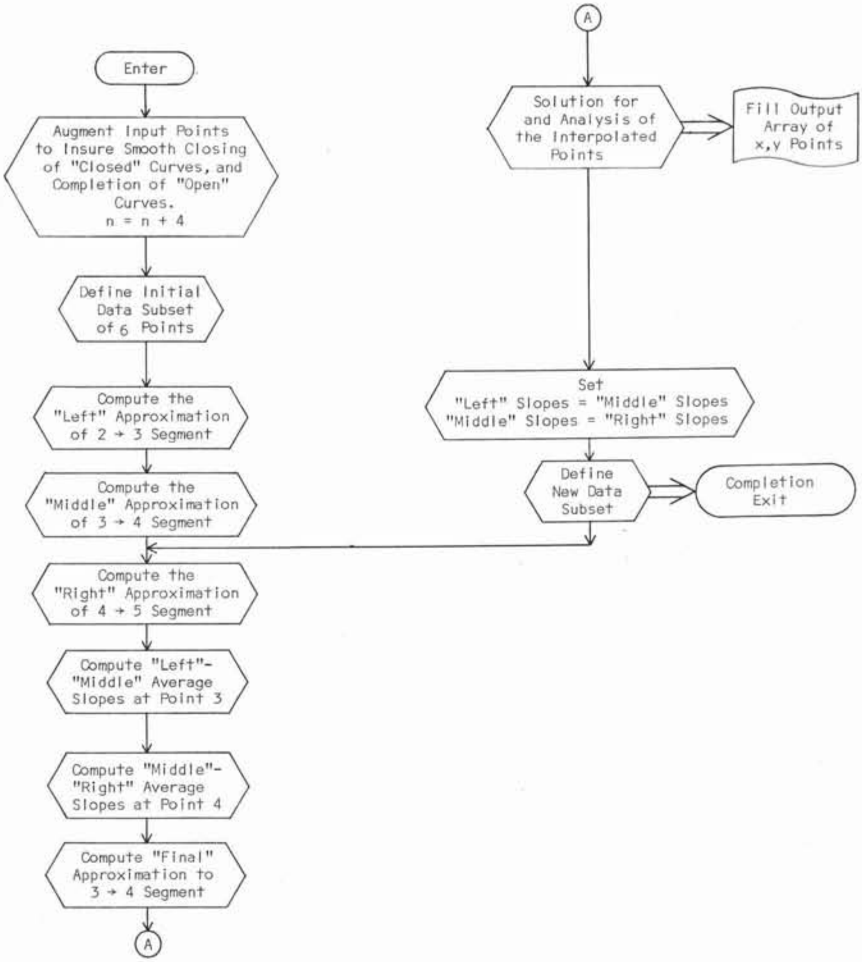

FIG. 5. The SMOOTH algorithm.

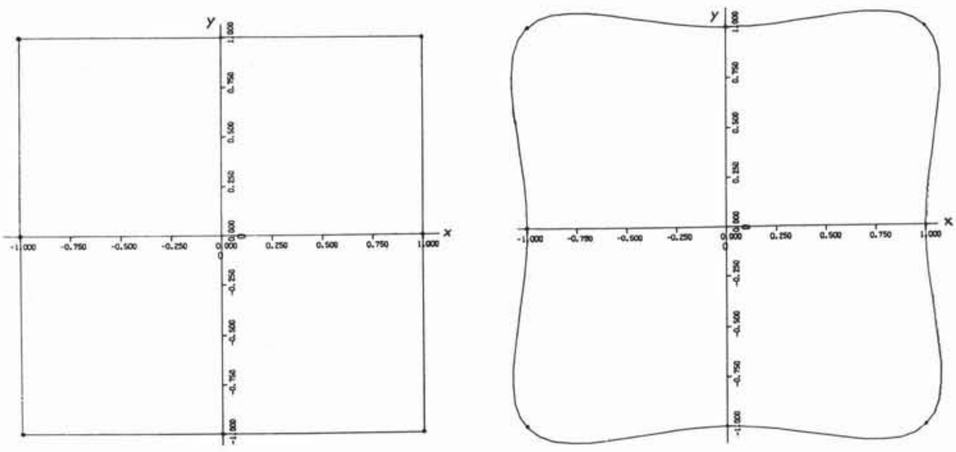

FIG. 6. Demonstration of the SMOOTH interpretation algorithm with symmetric data. *Left*—Original data connected by straight lines. *Right*—Plot of SMOOTHed points.

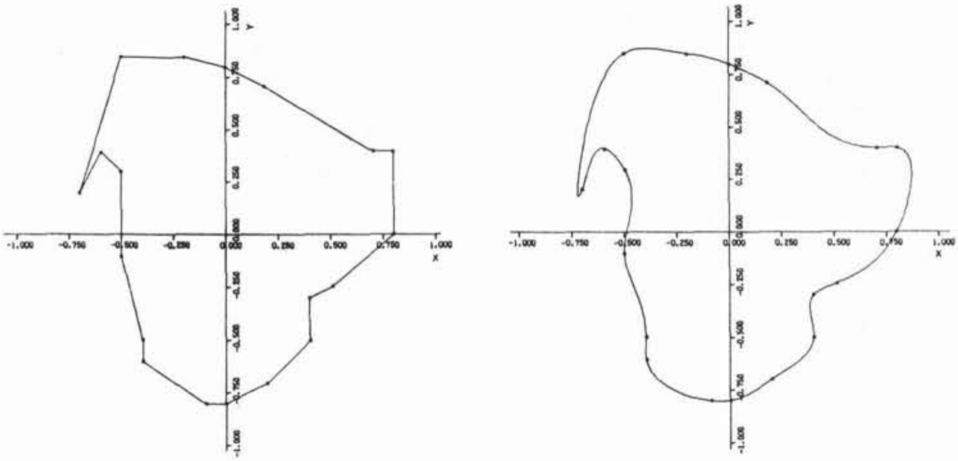

FIG. 7. Demonstration of the SMOOTH algorithm with typical unevenly spaced and irregular data. *Left*—Original data connected by straight line. *Right*—SMOOTH output.

Figure 7-*left* displays a bit more realistic set of points, typical of points along a contour line. The SMOOTHed curve in Figure 9-*right* is an *esthetically pleasing* smooth curve passing through these data.

In Figure 8-*left*, exact but irregularly spaced points along contours of the analytic surface function

$$Z = \frac{1000}{(y - x^2)^2 + (1 - x)^2 + 1} + \frac{500}{(x - 5)^2 + (y - 8)^2 + 1} + \frac{500}{(x - 8)^2 + (y - 8)^2 + 1} \tag{30}$$

are displayed, connected by straight lines. In Figure 8-*right*, the points resulting from application of the SMOOTH interpolation procedure are presented. The points

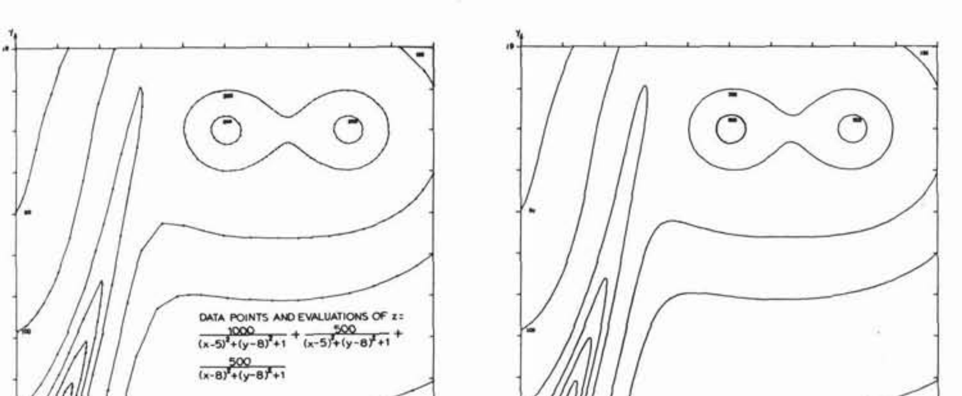

FIG. 8. Demonstration of the SMOOTH algorithm for contouring non-linear surfaces. *Left*—Original data connected by straight lines. *Right*—SMOOTH output.

in Figure 8-*right* resulting from SMOOTH differ typically in the fourth decimal place from the exact coordinates determined from precision root-solving of Equation 30, and are graphically identical to the actual contours upon plotting as in 8-*right*.

Concluding Remarks

The smoothing interpolation procedure is a reliable and efficient means for the computer approximation and display of discretely sampled unknown curves.

The comments and criticisms of Messers D. L. Light and M. J. Biggin of USA TOPOCOM are gratefully acknowledged.

References

1. Korn, G. A. and Korn, T. M., *Mathematical Handbook for Engineers and Scientists*. McGraw-Hill, New York, 1968, p. 20-5-2.
2. Greville, T. N. E., "Spline Functions, Interpolation and Numerical Quadrature," *Mathematical Methods for Digital Computers*, Vol. 2, A. Ralston and H. S. Wolf (Editors), Wiley, New York, 1967, ch. 8.
3. Junkins, J. L. and Jancaitis, J. R., "Mathematical Terrain Modeling," JLJ Rept. 7101, Final Rept., U.S. Army Contract DACA 71-71-0802, Jan. 1971, Charlottsville, Va.
4. Baker, R. M. L. and Makemson, M., *Astrodynamics*, 2nd ed., Academic Press, N. Y., N. Y., 1967, Appendix I.