Least-squares Fitting of Polygons[1]

J. Sinnreich

Civil Engineering Department, University of Florida, Gainesville, Florida 32611

e-mail: jsinn@ufl.edu

Abstract—Fitting a polygon to a set of given points in the plane is a problem which may arise in certain engineering, computer graphics or scientific applications. This paper presents an algorithm which computes a continuous function closely approximating various polygons, for which the sum of the squares of the distance to the given set of points is minimized.

Keywords: least squares, curve fitting.

DOI: 10.1134/S1054661816020218

INTRODUCTION (DEFINITION OF PROBLEM)

Data regression or curve-fitting to continuous closed-form two-dimensional shapes has typically been limited to very smooth curves such as circles and ellipses [3–5]. A solution for curve-fitting an arbitrary rectangle (that is, not centered on the origin and not aligned with the coordinate system) was required for analyzing certain engineering data. In order to implement a geometric least-squares solution with Gauss-Newton optimization, a continuous function which adequately modeled a polygon was needed. The parametric function chosen is the hypotrochoid, which is the trace of a point attached to a small circle of radius b rolling around the inside of a larger circle of radius a, where the trace point is a distance d from the center of the interior circle. Its most familiar incarnation is the "Spirograph" toy which allows the user to create interesting geometric patterns using geared circles of various diameters in a planetary gear configuration.

The analyses below closely follow the methods described by Gander [4], and for clarity, similar notation is used wherever possible. Note that procedures for producing the pseudoinverse of the Jacobian matrix, J^+, are not discussed since they are adequately addressed in [1] and elsewhere.

The parametric equations of the hypotrochoid function $h(\varphi)$ are defined as:

$$x = h_1(\varphi) = (a-b)\cos\varphi + d\cdot\cos\left(\frac{a-b}{b}\varphi\right),$$

$$y = h_2(\varphi) = (a-b)\sin\varphi - d\cdot\sin\left(\frac{a-b}{b}\varphi\right).$$

[1] The article is published in the original.

Received November 20, 2014

Table 1 below lists some specific combinations of a, b and d, and their corresponding geometric results.

Note that the ratios of a and b are fixed in order to create closed curves which approximate the desired geometric shapes. The values of d listed were arrived at by trial-and-error, and may be adjusted slightly to "sharpen" the vertices or straighten the line segments of the resulting curve as desired. Polygons with more vertices than an octagon may be defined, but begin to approximate a circle to such an extent that attempting to uniquely identify them in data sets with any amount of scatter may be impractical.

PROBLEM SOLUTION IN PARAMETRIC FORM

To generalize the solution to a shape of arbitrary size, orientation and location in the plane, the hypotrochoid function scale multiplier g, coordinate transform Q and centerpoint z are introduced:

$$\begin{Bmatrix} x(\varphi) \\ y(\varphi) \end{Bmatrix} = \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} + g\cdot Q(\alpha)\begin{Bmatrix} h_1(\varphi) \\ h_2(\varphi) \end{Bmatrix},$$

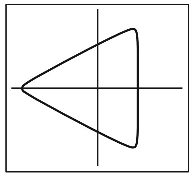
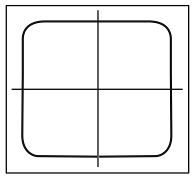
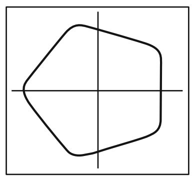
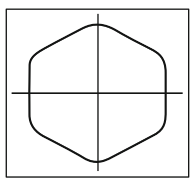
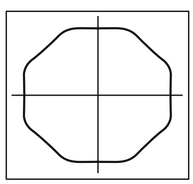
$$Q = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}.$$

Typically for symmetric polygons, g is a single value. Thus, any symmetric polygon is defined by four parameter values $u = (z_1, z_2, g, \alpha)$: the x- and y- coordinate of its centerpoint z, its hypotrochoid size g and its angular orientation relative to the coordinate axes α. Note that g is not a simple distance scalar, but is a function of a, b and d. If the parameter r is defined as the scalar distance from the centerpoint of the polygon to the midpoint of any of its faces, then the scale of the polygon is computed as:

$$r = g(a-b+d).$$

The original motivator for this calculation, the identification of a rectangle, may be defined by making g a two-valued parameter $g = \{g_1, g_2\}^T$. Thus, each

Table 1. Definition of hypotrochoid shapes

a	b	d	Shape	
3	1	−0.6	Triangle	
4	1	−0.4	Square	
5	1	−0.3	Pentagon	
6	1	−0.25	Hexagon	
8	1	−0.2	Octagon	

of the aligned axes of the rectangle has its own scaling factor, "stretching" a square into a rectangular shape defined by a total of five parameters $u = (z_1, z_2, g_1, g_2, \alpha)$.

The fitting procedure follows the parametric curve-fitting solution described by Gander [4]. Given the measured data points

$$X = \begin{bmatrix} x_1 & y_1 \\ \vdots & \vdots \\ x_m & y_m \end{bmatrix}$$

we need to determine the parameters u of the curve which minimize the geometric distance residual

$$f_i^2 = \min_{\varphi_i}[(x_i - x(\varphi_i))^2 + (y_i - y(\varphi_i))^2].$$

In order to find

$$\sum_{i=1}^{m} f_i^2 = \min$$

we simultaneously minimize for $v = \{\varphi_1, \ldots, \varphi_m, z_1, z_2, g, \alpha\}$; that is, find the minimum of the quadratic function

$$P(\varphi_1, \ldots, \varphi_m, z_1, z_2, g, \alpha) = \sum_{i=1}^{m}[(x_i - x(\varphi_i))^2 + (y_i - y(\varphi_i))^2],$$

which is equivalent to solving the nonlinear least-squares problem:

$$q_i = \begin{Bmatrix} x_i \\ y_i \end{Bmatrix} - \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} + g \cdot Q(\alpha) \begin{Bmatrix} h_1(\varphi_i) \\ h_2(\varphi_i) \end{Bmatrix} \approx 0, \quad i = 1, \ldots, m.$$

Thus we have $2m$ nonlinear equations for $m + 4$ unknowns (or $m + 5$ if $g_1 \neq g_2$). To compute the Jacobian, we need the partial derivatives of q_i:

$$\frac{\partial q_i}{\partial \phi_i} = \delta_{ij} Q(\alpha) \begin{Bmatrix} g_1 \cdot \dot{h}_1 \\ g_2 \cdot \dot{h}_2 \end{Bmatrix},$$

$$\frac{\partial q_i}{\partial z_1} = \begin{Bmatrix} -1 \\ 0 \end{Bmatrix}, \quad \frac{\partial q_i}{\partial z_2} = \begin{Bmatrix} 0 \\ -1 \end{Bmatrix}, \quad \frac{\partial q_i}{\partial \alpha} = -\dot{Q}(\alpha) \begin{Bmatrix} h_1 \cdot H_1(\varphi_i) \\ h_2 \cdot H_2(\varphi_i) \end{Bmatrix},$$

$$\frac{\partial q_i}{\partial g} = -Q(\alpha) \begin{Bmatrix} h_1(\varphi_i) \\ h_2(\varphi_i) \end{Bmatrix} \quad (g_1 = g_2),$$

$$\frac{\partial q_i}{\partial g_1} = -Q(\alpha) \begin{Bmatrix} h_1(\varphi_i) \\ 0 \end{Bmatrix}$$

$$\frac{\partial q_i}{\partial g_2} = -Q(\alpha) \begin{Bmatrix} 0 \\ h_2(\varphi_i) \end{Bmatrix} \quad (g_1 \neq g_2),$$

where

$$\dot{Q}(\alpha) = \begin{bmatrix} -\sin\alpha & -\cos\alpha \\ \cos\alpha & -\sin\alpha \end{bmatrix},$$

$$\begin{Bmatrix} \dot{h}_1(\varphi_i) \\ \dot{h}_2(\varphi_i) \end{Bmatrix} = \begin{Bmatrix} -((a-b)\sin(\phi_i) + d(b^{-1}(a-b))\sin(b^{-1}(a-b)\varphi_i)) \\ ((a-b)\cos(\phi_i) - d(b^{-1}(a-b))\cos(b^{-1}(a-b)\varphi_i)) \end{Bmatrix}$$

and the Kronecker delta notation is employed:

$$\delta_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases}.$$

The Jacobian is constructed as:

$$J = \begin{bmatrix} H_1 & A \\ H_2 & B \end{bmatrix},$$

where

$$H_1 = \mathrm{diag}(g_1 \cdot \dot{h}_1(\varphi_i) \cdot \cos(\alpha) - g_2 \cdot \dot{h}_2(\varphi_i) \cdot \sin(\alpha))$$

and

$$H_2 = \mathrm{diag}(g_1 \cdot \dot{h}_1(\varphi_i) \cdot \sin(\alpha) + g_2 \cdot \dot{h}_2(\varphi_i) \cdot \cos(\alpha))$$

are $m \times m$ diagonal matrices.

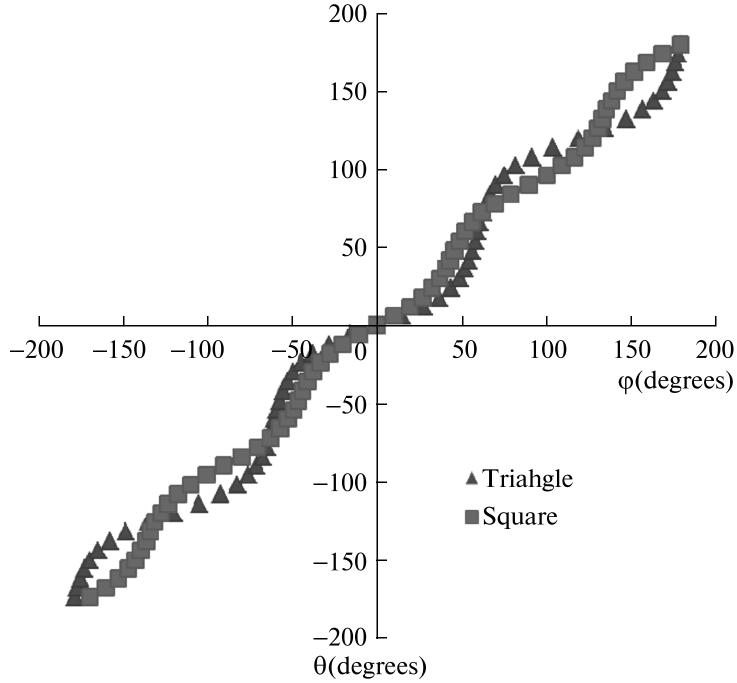


Fig. 1 Angle φ as a function of angle θ for triangles and squares.

If $g_1 = g_2$, then A and B are $m \times 4$ matrices defined as:

$$A(i,1:4) = [-1\ 0\ -(h_1(\varphi_i)\cos(\alpha) - h_2(\varphi_i)\sin(\alpha))\ g(h_1(\varphi_i)\sin(\alpha) + h_2(\varphi_i)\cos(\alpha))],$$

$$B(i,1:4) = [0\ -1\ -(h_1(\varphi_i)\sin(\alpha) + h_2(\varphi_i)\cos(\alpha))\ -g(h_1(\varphi_i)\cos(\alpha) - h_2(\varphi_i)\sin(\alpha))],$$

If $g_1 \neq g_2$, then A and B are $m \times 5$ matrices defined as:

$$A(i,1:5) = [-1\ 0\ -h_1(\varphi_i)\cos(\alpha)\ h_2(\varphi_i)\sin(\alpha)\ g_1h_1(\varphi_i)\sin(\alpha) + g_2h_2(\varphi_i)\cos(\alpha)],$$

$$B(i,1:5) = [0\ -1\ -h_1(\varphi_i)\sin(\alpha)\ -h_2(\varphi_i)\cos(\alpha)\ -g_1h_1(\varphi_i)\cos(\alpha) + g_2h_2(\varphi_i)\sin(\alpha)].$$

The standard two-step Gauss-Newton iterative solution regime is followed, where

1.$\{t\} = J^+\{q\}$

2.$\{v\} := \{v\} + \{t\}$.

As a starting point for the parameters u for initiating the iterative solution, a circle with parameters $\{z_1, z_2, r\}$ may be fit to the data as described by Gander [4]. This will give a reasonable starting approximation of z. The initial hypotrochoid size parameter g may be back-calculated from the circle radius r as $g = r(a - b + d)^{-1}$, while the orientation parameter α is either simply assumed to equal zero, or estimated by assuming some data point furthest from the initial estimate of center represents a vertex of the polygon. For a rectangle, an ellipse of parameters $\{z_1, z_2, r_1, r_2, \alpha\}$ should be used for the initial estimate, which provides the additional benefit of a reasonable initial α.

Note that the hypotrochoid function angles in general are not equal to the angles of the data points if expressed in polar form: $\varphi_i \neq \theta_i = \tan^{-1}(y_i - z_2/x_i - z_1)$. Figure 1 illustrates the relationship between φ and θ for a (3, 1, −0.6) hypotrochoid (triangle) and a (4, 1, −0.4) hypotrochoid (square):

However, in lieu of deriving a complicated mapping function for each polygon (see below), as an initial estimate we may assume that $\varphi_i \approx \theta_i$.

SAMPLE RESULTS (VALIDATION)

The following tables present some sample results of the analysis presented above. Data points are generated by picking some input values for u. To test the robustness of the solution, noise in the form of $x_i := x_i + c(Rand - 0.5)$ is added to the data points, where Rand produces random numbers between 0 and 1 inclusive and c is a constant. In the figures, the dashed line indicates the starting geometric figure from which data points were derived, while the solid line is the hypotrochoid function fitting result to the noisy data points. Starting estimates for u were generated by fitting a circle as described in Gander [4].

Table 2. Triangle Fit

Parameter	Input	Output	x	y	Plot
z_1	−10.000	−9.5804	−5.26	32.70	
			−12.32	24.12	
z_2	4.000	3.3715	−19.10	10.33	
			−26.87	3.48	
g	10.000	10.2982	−36.74	−7.28	
			−21.67	−9.88	
α	25.00°	25.83°	−9.27	−8.33	
			0.48	−13.34	
c	5.0	–	13.88	−11.42	
			7.77	−0.92	
convergence in 23 iterations			0.62	10.04	
			−1.46	22.68	

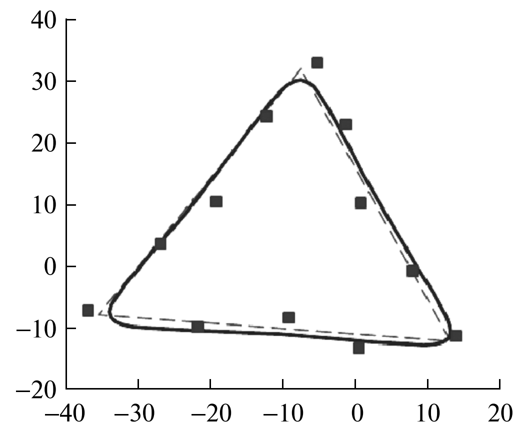
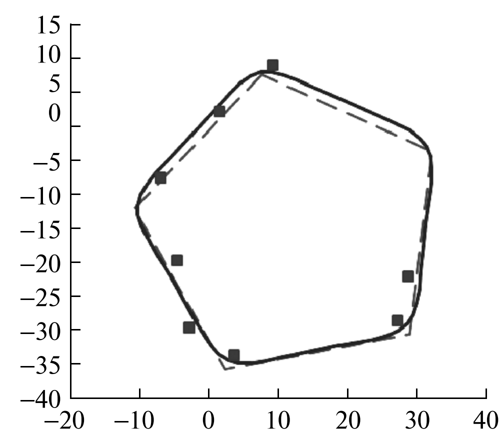


Table 3. Pentagon Fit

Parameter	Input	Output	x	y	Plot
z_1	12.000	11.9373	9.26	8.73	
			1.67	1.93	
z_2	−15.000	−14.0902	−6.85	−7.67	
			−4.50	−19.93	
g	5.000	5.1810	−2.71	−29.77	
			3.83	−33.80	
α	65.00°	65.74°	27.01	−28.72	
			28.66	−22.18	
c	5.0	–			
convergence in 7 iterations					



ANALYTICAL PROBLEM SOLUTION FOR RECTANGLES AND SQUARES

One potential disadvantage of the parametric solution described above is the size of the Jacobian matrix, which must be inverted, as the number of data points m becomes arbitrarily large. For example, the original problem motivating this analysis typically included 50 to 100 data points. If the hypotrochoid function input angles can be approximated separately, rather than adjusted directly with the Gauss-Newton iteration, then only the five parameters actually defining the geometry of the rectangle need to be solved for. The details of estimating φ from g_1, g_1 and θ for a rectangle are described in the next section below. To implement the solution, the input angles are adjusted by the alignment angle α:

$$\{x_i'\} = [Q]\{x_i\}$$

from which aligned angles θ_i' and φ_i' are also computed. Next, the residual vector of distances between each data point and the hypotrochoid rectangle is computed:

$$\{f_i(u)\}_{i=1,\dots,m} = \{\sqrt{(h_1(\varphi_i') - z_1)^2 + (h_2(\varphi_i') - z_2)^2} \\ -\sqrt{(x_i' - z_1)^2 + (y_i' - z_2)^2}\}.$$

Next, construct the $(m \times 5)$ Jacobian matrix

$$J = \begin{bmatrix} \frac{\partial f_1(u)}{\partial z_1} & \frac{\partial f_1(u)}{\partial z_2} & \frac{\partial f_1(u)}{\partial g_1} & \frac{\partial f_1(u)}{\partial g_2} & \frac{\partial f_1(u)}{\partial \alpha} \\ \cdot & \cdot & & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & \cdot \\ \frac{\partial f_m(u)}{\partial z_1} & \frac{\partial f_m(u)}{\partial z_2} & \frac{\partial f_m(u)}{\partial g_1} & \frac{\partial f_m(u)}{\partial g_2} & \frac{\partial f_m(u)}{\partial \alpha} \end{bmatrix}$$

using the partial derivatives of the hypotrochoid function with respect to each of the five rectangle parameters $\{z_1, z_2, g_1, g_2, \alpha\}$. These may be derived from the equations given above by a tedious but trivial process.

Finally, the standard two-step Gauss-Newton iterative solution regime is again followed, where

1. $\{t\} = J^{+}\{f\}$,
2. $\{u\} := \{u\} + \{t\}$.

The square can in this instance be considered a special case of the rectangle, where $g_2 \equiv g_1$.

RECTANGULAR HYPOTROCHOID ANGLE MAPPING FUNCTION

The m input data points to be fit are given in polar form $\{R_i, \theta_i\}_{i=1,\dots,m}$. Note the hypotrochoid function angle φ denotes the rotation of the smaller circle of radius b within the larger circle of radius a, and must not be confused with the polar coordinate θ of the data set to be fit, since the two are generally not equal. The following algorithm converts the data input angle θ into the hypotrochoid angle φ for a rectangle:

Assuming for the moment that the rectangle is aligned along the x-axis ($\alpha = 0$), and the center offset is negligible $(z_1, z_2) \approx (0, 0)$, the relationship between the polar data and hypotrochoid parametric function is defined as:

$$x_i = r_i \cos(\theta_i) \approx g_1 h_1(\varphi_i) = s_i \cos(\theta_i),$$

$$y_i = r_i \sin(\theta_i) \approx g_2 h_2(\varphi_i) = s_i \sin(\theta_i).$$

Therefore,

$$s_i = \frac{g_1 h_1(\varphi_i)}{\cos(\theta_i)} = \frac{g_2 h_2(\varphi_i)}{\sin(\theta_i)} \text{ and}$$

$$g_2 h_2(\varphi_i) = g_1 h_1(\varphi_i) \tan(\theta_i).$$

By trigonometric identities,

$$g_1 h_1(\varphi_i) = g_1 [3\cos(\varphi_i) - 0.4\cos(3\varphi_i)]$$

$$= g_1[3\cos(\varphi_i) - 0.4(4\cos^3(\varphi_i) - 3\cos(\varphi_i))]$$

$$= g_1[4.2\cos(\varphi_i) - 1.6\cos^3(\varphi_i)]$$

and

$$g_2 h_2(\varphi_i) = g_2 [3\sin(\varphi_i) + 0.4\sin(3\varphi_i)]$$

$$= g_2[3\sin(\varphi_i) + 0.4(3\sin(\varphi_i) - 4\sin^3(\varphi_i))]$$

$$= g_2[4.2\sin(\varphi_i) - 1.6\sin^3(\varphi_i)]$$

$$= g_1 h_1(\varphi_i)\tan(\theta_i) = g_1 [3\cos(\varphi_i) - 0.4\cos(3\varphi_i)]\tan(\theta_i).$$

Square:

$$(g_1 h_1(\varphi_i))^2$$

$$= g_1^2[17.64\cos^2(\varphi_i) - 13.44\cos^4(\varphi_i) + 2.56\cos^6(\varphi_i)],$$

$$(g_2 h_2(\varphi_i))^2$$

$$= g_2^2[17.64\sin^2(\varphi_i) - 13.44\sin^4(\varphi_i) + 2.56\sin^6(\varphi_i)].$$

Rearrange terms:

$$(g_1 h_1(\varphi_i)\tan(\theta_i))^2$$

$$= (g_1 h_1(\varphi_i))^2\left(\tan^2(\theta_i) + \left(\frac{g_2}{g_1}\right)^2\right) - (g_1 h_1(\varphi_i))^2\left(\frac{g_2}{g_1}\right)^2$$

$$= (g_2 h_2(\varphi_i))^2,$$

$$(g_1 h_1(\varphi_i))^2\left(\tan^2(\theta_i) + \left(\frac{g_2}{g_1}\right)^2\right)$$

$$= (g_1 h_1(\varphi_i))^2\left(\frac{g_2}{g_1}\right)^2 + (g_2 h_2(\varphi_i))^2.$$

Combine the equations from the steps above:

$$\left(\tan^2(\theta_i) + \left(\frac{g_2}{g_1}\right)^2\right) g_1^2[17.64\cos^2(\varphi_i)$$

$$-13.44\cos^4(\varphi_i) + 2.56\cos^6(\varphi_i)]$$

$$= g_2^2[17.64(\sin^2(\varphi_i) + \cos^2(\varphi_i))$$

$$-13.44(\sin^4(\varphi_i) + \cos^4(\varphi_i))$$

$$+2.56(\sin^6(\varphi_i) + \cos^6(\varphi_i))]$$

$$= g_2^2\left[17.64(1) - 13.44\left(\frac{3}{4} + \frac{1}{4}\cos(4\varphi_i)\right)\right.$$

$$\left.+2.56\left(\frac{5}{8} + \frac{3}{8}\cos(4\varphi_i)\right)\right] = g_2^2[9.16 - 2.40\cos(4\varphi_i)]$$

$$= g_2^2[9.16 - 2.40(8\cos^4(4\varphi_i) - 8\cos^2(4\varphi_i) + 1)]$$

$$= g_2^2[6.76 + 19.20\cos^2(\varphi_i) - 19.20\cos^4(\varphi_i)].$$

Rearrange:

$$-6.76g_2^2 - (19.20 - 17.64(g_1^2\tan^2(\theta_i) + g_2^2))\cos^2(\varphi_i)$$

$$+(19.20 - 13.44(g_1^2\tan^2(\theta_i) + g_2^2))\cos^4(\varphi_i)$$

$$+2.56(g_1^2\tan^2(\theta_i) + g_2^2)\cos^6(\varphi_i) = 0.$$

Normalizing the leading term:

$$\cos^6\varphi - \frac{(13.44(g_1^2\tan^2(\theta_i)) - 5.76g_2^2)}{2.56(g_1^2\tan^2(\theta_i) + g_2^2)}\cos^4(\varphi_i)$$

$$+\frac{(1.56g_2^2 - 17.64(g_1^2\tan^2(\theta_i)))}{2.56(g_1^2\tan^2(\theta_i) + g_2^2)}\cos^2(\varphi_i)$$

$$-\frac{6.76g_2^2}{2.56(g_1^2\tan^2(\theta_i) + g_2^2)} = 0.$$

Solve the cubic function of w_i:

$$w_i^3 + \beta_{1i}w_i^2 + \beta_{2i}w_i + \beta_{3i} = 0,$$

where

$$w_i = \cos^2(\phi_i),$$

$$\beta_{1i} = \frac{(5.76g_2^2 - 13.44(g_1^2\tan^2(\theta_i)))}{2.56(g_1^2\tan^2(\theta_i) + g_2^2)},$$

$$\beta_{2i} = \frac{(17.64(g_1^2\tan^2(\theta_i)) - 1.56g_2^2)}{2.56(g_1^2\tan^2(\theta_i) + g_2^2)},$$

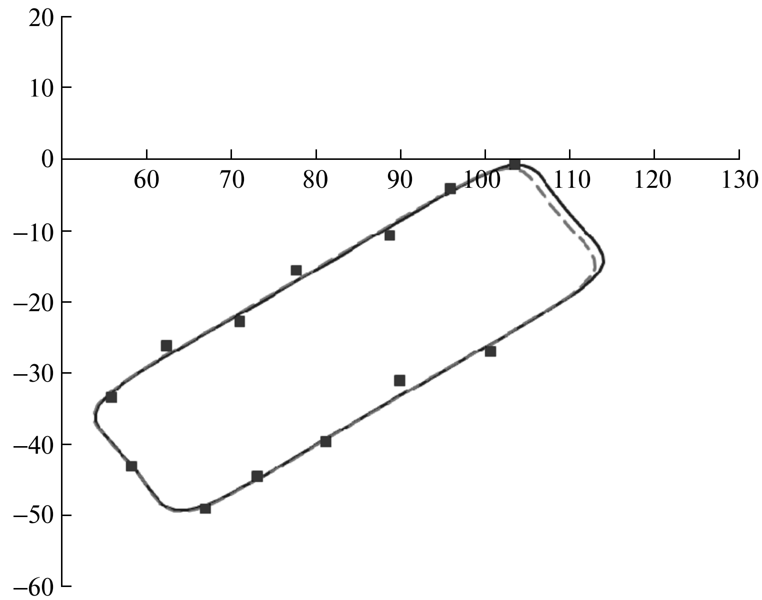


Fig. 2 Rectangle Fit Comparison Plot.

$$\beta_{3i} = \frac{-6.76g_2^2}{2.56(g_1^2\tan^2(\theta_i) + g_2^2)}$$

and

$$\varphi_i = \cos^{-1}(\sqrt{w}).$$

Thus, for an estimate of g_1, g_1 and data point angle θ_i, the corresponding hypotrochoid angle φ_i can be calculated.

RECTANGLE SOLUTION COMPARISON

For comparison purposes, both of the solutions described above are used to fit an arbitrarily-generated rectangle shape. Note that if no noise is introduced into the data, both the parametric and analytical solution yield essentially the exact values of the input parameters. For a noisy data set, each solution yields slightly different results, tabulated below.

In Fig. 2, the parametric solution result is plotted as a solid line, while the analytical solution result is plot-

Table 4. Rectangle Fit Comparison

Parameter	Input	Parametric Output	Analytical Output	x	y
z_1	84.00	84.0129	83.4874	103.6432	−0.8606
				96.0274	−4.2335
z_2	−25.00	−25.1138	−25.3913	88.8014	−10.7927
				77.847	−15.6558
g_1	12.00	3.8676	11.7878	71.1906	−22.8952
				62.6048	−26.2344
g_2	4.00	11.9909	3.8990	56.0376	−33.58
				58.4946	−43.1159
α	35.00°	124.65°	34.70°	67.1746	−49.0029
				73.2778	−44.6708
c	3.0	–	–	81.3374	−39.741
Convergence iterations:		4	11	90.1028	−31.0328
				100.7308	−27.0687

ted as a dashed line. Note that the parametric solution inverts the values of g_1 and g_2, while adding 90° to the value of α. This in effect produces a "short, wide" rather than a "long, thin" rectangle, but the resulting geometric shape is effectively the same. The analytical solution requires more iterations to converge, but requires the inversion of a smaller Jacobian matrix.

CONCLUDING REMARKS

The algorithms outlined above permit the use of the robust and well-known least-squares technique for what are traditionally not thought of as continuous functions. The analyses present a relatively robust solution to the problem of identifying a polygon from scattered data as opposed to alternative approaches, which may for example seek to identify each of the four sides of a rectangle independently and then impose constraints of a fixed $\pi/2$ angle between the four line segments (see [2]). Note the rectangle data set is effectively missing one entire side of the rectangle, and has relatively few data points in total, yet the fitting algorithms yield a reasonable approximation of the original shape, due to the fact that a single continuous function is being fit.

The analytical solution for rectangles has the advantage of a potentially greatly-reduced Jacobian to invert. However, it relies on a mapping function of θ to φ which is specific for a (4, 1, –0.4) hypotrochoid, and due to the nature of the trigonometric identities used in its derivation, may not be easily replicable for other polygons.

REFERENCES

1. A. Björck, *Numerical Methods for Least Squares Problems* (SIAM, Philadelphia, 1996).
2. C. Carlsson, "Vehicle size and orientation estimation using geometric fitting," Thesis (Dep. of Electrical Engineering, Linköpings univ., 2000), No. 840.
3. M. G. Cox and H. M. Jones, "An algorithm for least-squares circle fitting to data with specified uncertainty ellipses," IMA J. Num. Anal. **9** (3), 285–298 (1989).
4. W. Gander, G. H. Golub, and R. Strebel, "Least-squares fitting of circles and ellipses," BIT Num. Math. **34** (4), 558–578 (1994).
5. L. Al-Subaihi and G. A. Watson, "Fitting parametric curves and surfaces by l_∞ distance regression," BIT Num. Math. **45** (3), 443–461 (2005).

Jon Sinnreich obtained his M. Eng. in Offshore Structural Engineering at the University of Florida, and is a registered Professional Engineer in the State of Florida. Currently, he is employed as the Research Services Coordinator at the University of Florida's Powell Family Structures and Materials Laboratory. From 1998 to 2015, he was a staff engineer at Loadtest Inc. His areas of interest include advanced analysis and processing of data collected in structural and geotechnical load testing.