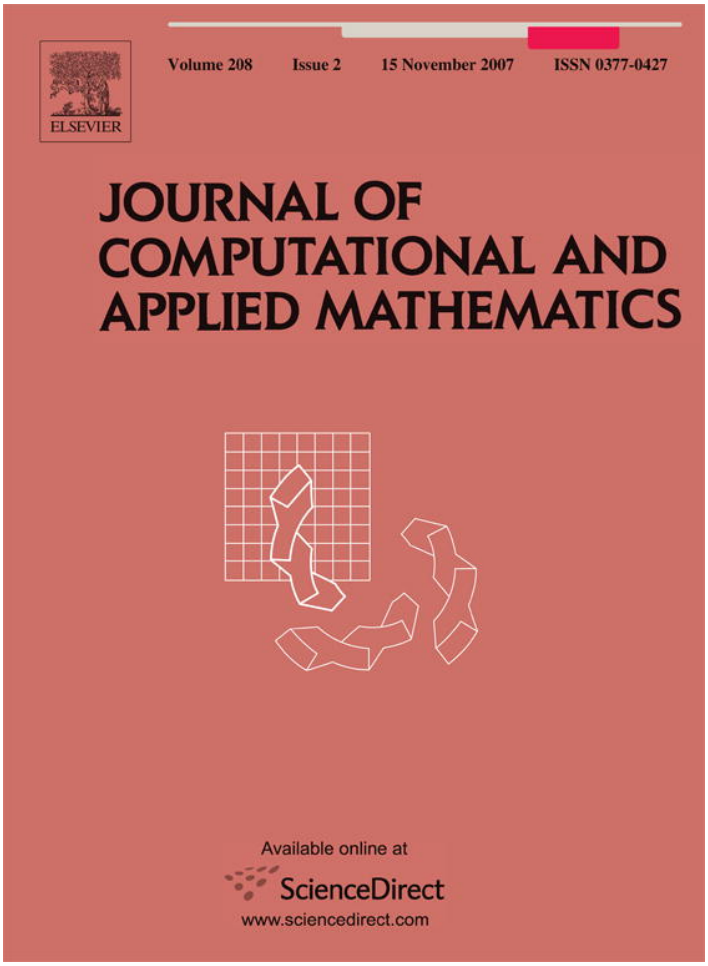

Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 208 (2007) 331–340
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

A Levenberg–Marquardt method for estimating polygonal regions

G.A. Watson

Department of Mathematics, University of Dundee, Dundee DD1 4HN, Scotland, UK

Received 20 August 2005; received in revised form 2 October 2006

Abstract

The problem is considered of the estimation of a polygonal region in two dimensions from data approximately marking the outline of the region. A solution is sought by formulating and solving a nonlinear least squares problem. A Levenberg–Marquardt method is developed for this problem, with an implementation which exploits the special structure so that the Levenberg–Marquardt step can be computed efficiently.

MSC: 65D10; 65K05

Keywords: Polygonal region estimation; Nonlinear least squares; Levenberg–Marquardt

1. Introduction

Of interest here is the estimation of a polygonal region in two dimensions from given data approximately marking the outline of the region. Problems of this kind arise in land surveying, agriculture and forestry (see, for example, [2,3,7,8]), when the data may be provided by noisy GPS (Global Positioning System) measurements. It is assumed that the given region has m corners or vertices and that these can be identified from the measurements. The remaining data correspond to points which should lie on the lines which join adjacent corners. Let the measured corner estimates be (x_i, y_i), $i = 1, \ldots, m$, and let the points which should lie along the lines between the corners defined by indices i and $i + 1$ be measured as (x_{ij}, y_{ij}), $i = 1, \ldots, m$, $j = 1, \ldots, n_i$. When the index $m + 1$ appears, it will be assumed that it defines the same point as index 1. Similarly the subscript $m + 1$ can be replaced by the subscript 1.

Because the data are not precise, of course there will not be an exact fit, and we consider least squares solutions. Let the corresponding modelled estimates be (X_i, Y_i), $i = 1, \ldots, m$, and (X_{ij}, Y_{ij}), $i = 1, \ldots, m$, $j = 1, \ldots, n_i$, so that the number of variables is $2m + 2p$, where

$$p = \sum_{i=1}^{m} n_i.$$

Then we can consider solving the least squares problem

$$\text{minimize} \quad \sum_{i=1}^{m} \sum_{j=1}^{n_i} \{(x_{ij} - X_{ij})^2 + (y_{ij} - Y_{ij})^2\} + \sum_{i=1}^{m} \{(x_i - X_i)^2 + (y_i - Y_i)^2\},$$

E-mail address: gawatson@maths.dundee.ac.uk.

0377-0427/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2006.10.001

subject to the condition that for $i = 1, \ldots, m$, $j = 1, \ldots, n_i$, the points (X_{ij}, Y_{ij}) are constrained to lie on the line joining (X_i, Y_i) to (X_{i+1}, Y_{i+1}).

In the next section a method of Levenberg–Marquardt type is developed, and it is shown how advantage can be taken of the special structure. This approach is similar to that used in [1] for a different nonlinear least squares problem. In Section 3, some numerical results are presented.

2. A Levenberg–Marquardt method

Assume that $n_i \geqslant 1$, $i = 1, \ldots, m$. The condition for points to lie on a line joining (X_i, Y_i) to (X_{i+1}, Y_{i+1}) corresponds to satisfying

$$\begin{aligned} &Y_i = aX_i + b, \\ &Y_{ij} = aX_{ij} + b, \quad j = 1, \ldots, n_i, \\ &Y_{i+1} = aX_{i+1} + b, \end{aligned}$$

for some a, b. The first and last can be solved for a and b so that we can express Y_{ij} in terms of $X_i, Y_i, X_{i+1}, Y_{i+1}, X_{ij}$ for every $i = 1, \ldots, m$, $j = 1, \ldots, n_i$. Let

$$q_i = \frac{Y_{i+1} - Y_i}{X_{i+1} - X_i}, \quad i = 1, \ldots, m,$$

which is defined unless the denominator is zero for any i. While a zero denominator is theoretically possible, it is unlikely in the context of genuinely error-contaminated data. However, this situation can be entirely avoided by changing the co-ordinate system to eliminate the potential problem, although some prior knowledge of the polygon is then needed. Other approaches make it possible to eliminate these restrictions. For example in [7,8], the co-ordinates of the corners are kept as parameters together with the relative position (a number between 0 and 1) for every other point, so that no singularities appear.

The problem may be restated as the unconstrained nonlinear least squares problem of minimizing

$$\mathbf{g}_1^{\mathrm{T}}\mathbf{g}_1 + \mathbf{g}_2^{\mathrm{T}}\mathbf{g}_2 + \mathbf{g}_3^{\mathrm{T}}\mathbf{g}_3,$$

where

$$\mathbf{g}_1 = [x_{11} - X_{11}, \ldots, x_{1n_1} - X_{1n_1}, \ldots, x_{m1} - X_{m1}, \ldots, x_{mn_m} - X_{mn_m}]^{\mathrm{T}} \in R^p,$$

$$\mathbf{g}_2 = [x_1 - X_1, y_1 - Y_1, \ldots, x_m - X_m, y_m - Y_m]^{\mathrm{T}} \in R^{2m},$$

$$\begin{aligned} \mathbf{g}_3 = [&y_{11} - Y_1 - q_1(X_{11} - X_1), \ldots, y_{1n_1} - Y_1 - q_1(X_{1n_1} - X_1), \ldots, \\ &y_{m1} - Y_m - q_m(X_{m1} - X_m), \ldots, y_{mn_m} - Y_m - q_m(X_{mn_m} - X_m)]^{\mathrm{T}} \in R^p. \end{aligned}$$

Here we have a nonlinear least squares problem in R^{2m+2p} with $2m + p$ unknowns. We can solve this problem by standard methods. Suppose we have an approximation to the solution, that is a particular set of values of the unknowns, and let $\mathbf{g}^{\mathrm{T}} = [\mathbf{g}_1^{\mathrm{T}}, \mathbf{g}_2^{\mathrm{T}}, \mathbf{g}_3^{\mathrm{T}}]$, and the Jacobian matrix $J = \nabla\mathbf{g}$ be evaluated there. Then systematic progress towards a solution can be obtained by solving a sequence of subproblems having the form

$$\begin{aligned} &\text{minimize} \quad \|\mathbf{g} + J\mathbf{d}\|^2 \\ &\text{subject to} \quad \|\mathbf{d}\|^2 \leqslant \tau, \end{aligned}$$

where $\|.\|$ denotes the l_2 norm. This is a standard trust region approach to the problem, where τ is adjusted adaptively, and the next approximation to the solution values is given by adding on $\gamma\mathbf{d}$, where $\gamma = 1$ or 0. This is repeated until convergence (to required accuracy) is obtained. The method has been shown to have good global convergence properties, see for example [4,5].

Alternatively, and in a sense equivalently, we can use a Levenberg–Marquardt approach, which is to solve

$$(J^{\mathrm{T}}J + \lambda I_{p+2m})\mathbf{d} = -J^{\mathrm{T}}\mathbf{g}, \tag{1}$$

for some positive λ, again adjusted adaptively [4]. Again this version of the method can have excellent global convergence properties: see for example [6], where convergence to a zero derivative is established provided that J and λ remain bounded.

Let the unknowns be partitioned into two groups so that we write them as

$$\mathbf{v}^{\mathrm{T}} = [\mathbf{v}_1^{\mathrm{T}}, \mathbf{v}_2^{\mathrm{T}}],$$

where

$$\mathbf{v}_1^{\mathrm{T}} = [X_{11}, \ldots, X_{1n_1}, X_{21}, \ldots X_{mn_m}] \in R^p,$$

$$\mathbf{v}_2^{\mathrm{T}} = [X_1, Y_1, \ldots, X_m, Y_m] \in R^{2m}.$$

Then the function to be minimized can be written as

$$F(\mathbf{v}) = \mathbf{g}^{\mathrm{T}}\mathbf{g},$$

and the Jacobian matrix J has the form

$$J = \begin{bmatrix} \nabla_{\mathbf{v}_1}\mathbf{g}_1 & \nabla_{\mathbf{v}_2}\mathbf{g}_1 \\ \nabla_{\mathbf{v}_1}\mathbf{g}_2 & \nabla_{\mathbf{v}_2}\mathbf{g}_2 \\ \nabla_{\mathbf{v}_1}\mathbf{g}_3 & \nabla_{\mathbf{v}_2}\mathbf{g}_3 \end{bmatrix} = \begin{bmatrix} -I_p & 0 \\ 0 & -I_{2m} \\ D & C \end{bmatrix},$$

where $D \in R^{p\times p}$ is a diagonal matrix, and $C \in R^{p\times 2m}$. In fact

$$D = \mathrm{diag}\{D_1, \ldots, D_m\},$$

where

$$D_i = -q_i I_{n_i}, \quad i = 1, \ldots, m.$$

Further

$$C = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{bmatrix},$$

where C_i is an $n_i \times 2m$ matrix with only four non-zero columns, those in positions $2i-1$ to $2i+2$ if $i = 1, \ldots, m-1$ and in positions $1, 2, 2m-1, 2m$ when $i = m$. Letting

$$p_{ij} = \frac{X_{ij} - X_i}{X_{i+1} - X_i}, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n_i,$$

specifically we have the jth row of C_i as

$$\mathbf{e}_j^{\mathrm{T}}C_i = [0 : \ldots : 0 : q_i(1 - p_{ij}) : p_{ij} - 1 : q_i p_{ij} : -p_{ij} : 0 : \ldots : 0], \quad j = 1, \ldots, n_i,$$

for $i = 1, \ldots, m-1$ and for $i = m$,

$$\mathbf{e}_j^{\mathrm{T}}C_m = [q_m p_{mj} : -p_{mj} : 0 \ldots 0 : q_m(1 - p_{mj}) : p_{mj} - 1], \quad j = 1, \ldots, n_m.$$

It follows that for any $\mathbf{x} \in R^{2m}$,

$$\mathbf{e}_j^{\mathrm{T}}C_i\mathbf{x} = q_i(1 - p_{ij})x_{2i-1} + (p_{ij} - 1)x_{2i} + q_i p_{ij}x_{2i+1} - p_{ij}x_{2i+2},$$

$$i = 1, \ldots, m, \quad j = 1, \ldots, n_i, \tag{2}$$

where the subscript $2m+1$ is replaced by 1 and $2m+2$ is replaced by 2.

The system (1) can be written as

$$\begin{bmatrix} D^2 + (1+\lambda)I_p & DC \\ C^{\mathrm{T}}D & C^{\mathrm{T}}C + (1+\lambda)I_{2m} \end{bmatrix} \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{g}_1 - D\mathbf{g}_3 \\ \mathbf{g}_2 - C^{\mathrm{T}}\mathbf{g}_3 \end{bmatrix},$$

where $\mathbf{d}^{\mathrm{T}} = [\mathbf{d}_1^{\mathrm{T}}, \mathbf{d}_2^{\mathrm{T}}]$ with $\mathbf{d}_1 \in R^p$, $\mathbf{d}_2 \in R^{2m}$. Define $P = D^2 + (1+\lambda)I_p$. Then we have

$$\mathbf{d}_1 = P^{-1}(\mathbf{g}_1 - D\mathbf{g}_3 - DC\mathbf{d}_2), \tag{3}$$

and so

$$[C^{\mathrm{T}}(I - DP^{-1}D)C + (1+\lambda)I]\mathbf{d}_2 = C^{\mathrm{T}}(DP^{-1}D - I)\mathbf{g}_3 + \mathbf{g}_2 - C^{\mathrm{T}}DP^{-1}\mathbf{g}_1. \tag{4}$$

Because P is diagonal, its inversion (and the inverse clearly always exists) is trivial. Further we will show that all quantities can in this case easily be computed. In particular, we will show that the matrix on the left-hand side of (4) is a 7-diagonal matrix, and so this can also be exploited in the solution process.

We have

$$DC = \begin{bmatrix} -q_1 C_1 \\ -q_2 C_2 \\ \vdots \\ -q_m C_m \end{bmatrix}.$$

Also since

$$P = \mathrm{diag}\{(q_i^2 + 1 + \lambda)I_{n_i}, i = 1, \ldots, m\},$$

then

$$I - DP^{-1}D = \mathrm{diag}\left\{ \frac{1+\lambda}{q_i^2 + 1 + \lambda} I_{n_i}, i = 1, \ldots, m \right\},$$

and

$$C^{\mathrm{T}}(I - DP^{-1}D)C = \sum_{i=1}^{m} \frac{1+\lambda}{q_i^2 + 1 + \lambda} C_i^{\mathrm{T}} C_i.$$

Note that $C_i^{\mathrm{T}}C_i$ only has 16 non-zero entries. Specifically for $i = 1, \ldots, m-1$, $C_i^{\mathrm{T}}C_i$ only has non-zeros in the intersection of rows $2i-1$ to $2i+2$ and columns $2i-1$ to $2i+2$; $C_m^{\mathrm{T}}C_m$ has only non-zero entries in the intersection of rows $1, 2, 2m-1, 2m$ and columns $1, 2, 2m-1, 2m$. Therefore for $i = 1, \ldots, m-1$, $C_i^{\mathrm{T}}C_i$ can be defined by the (symmetric) 4×4 matrix with 10 (in fact just 9) distinct elements

$$\begin{bmatrix} q_i^2 r_i & -q_i r_i & q_i^2 s_i & -q_i s_i \\ -q_i r_i & r_i & -q_i s_i & s_i \\ q_i^2 s_i & -q_i s_i & q_i^2 t_i & -q_i t_i \\ -q_i s_i & s_i & -q_i t_i & t_i \end{bmatrix}, \quad i = 1, \ldots, m-1,$$

and $C_m^{\mathrm{T}}C_m$ can similarly be defined by

$$\begin{bmatrix} q_m^2 t_m & -q_m t_m & q_m^2 s_m & -q_m s_m \\ -q_m t_m & t_m & -q_m s_m & s_m \\ q_m^2 s_m & -q_m s_m & q_m^2 r_m & -q_m r_m \\ -q_m s_m & s_m & -q_m r_m & r_m \end{bmatrix},$$

where

$$r_i = \sum_{j=1}^{n_i} (1 - p_{ij})^2,$$

$$s_i = \sum_{j=1}^{n_i} p_{ij}(1 - p_{ij}),$$

$$t_i = \sum_{j=1}^{n_i} p_{ij}^2.$$

It follows that for any α_i, $i = 1, \ldots, m$, the matrix

$$G = \sum_{i=1}^{m} \alpha_i C_i^{\mathrm{T}} C_i \tag{5}$$

is a 7-diagonal symmetric matrix whose non-zero upper diagonal elements are given by

$$\begin{aligned}
&G_{2i-1,2i-1} = \alpha_{i-1} q_{i-1}^2 t_{i-1} + \alpha_i q_i^2 r_i, \quad i = 1, \ldots, m,\\
&G_{2i-1,2i} = -\alpha_{i-1} q_{i-1} t_{i-1} - \alpha_i q_i r_i, \quad i = 1, \ldots, m,\\
&G_{2i-1,2i+1} = \alpha_i q_i^2 s_i, \quad i = 1, \ldots, m-1,\\
&G_{2i-1,2i+2} = -\alpha_i q_i s_i, \quad i = 1, \ldots, m-1,\\
&G_{2i,2i} = \alpha_{i-1} t_{i-1} + \alpha_i r_i, \quad i = 1, \ldots, m,\\
&G_{2i,2i+1} = -\alpha_i q_i s_i, \quad i = 1, \ldots, m-1,\\
&G_{2i,2i+2} = \alpha_i s_i, \quad i = 1, \ldots, m-1,
\end{aligned}$$

where any zero subscript is replaced by m.

If $\mathbf{g}_3$ and $\mathbf{g}_1$ are partitioned naturally into m subvectors of dimension n_i, $i = 1, \ldots, m$, say

$$\mathbf{g}_3^{\mathrm{T}} = \left[\mathbf{g}_3^{(1)^{\mathrm{T}}}, \ldots, \mathbf{g}_3^{(m)^{\mathrm{T}}}\right],$$

and similarly for $\mathbf{g}_1$, then

$$C^{\mathrm{T}}(I - DP^{-1}D)\mathbf{g}_3 = \sum_{i=1}^{m} \frac{1+\lambda}{q_i^2 + 1 + \lambda} C_i^{\mathrm{T}} \mathbf{g}_3^{(i)},$$

and

$$C^{\mathrm{T}} DP^{-1} \mathbf{g}_1 = \sum_{i=1}^{m} \frac{-q_i}{q_i^2 + 1 + \lambda} C_i^{\mathrm{T}} \mathbf{g}_1^{(i)}.$$

Of course, $C_i^{\mathrm{T}} \mathbf{g}_3^{(i)}$ and $C_i^{\mathrm{T}} \mathbf{g}_1^{(i)}$ will just have four non-zero components (out of $2m$), corresponding to the non-zero columns of C_i. For example, if we define

$$u_i = \sum_{j=1}^{n_i} (x_{ij} - X_{ij}), \tag{6}$$

$$v_i = \sum_{j=1}^{n_i} p_{ij}(x_{ij} - X_{ij}), \tag{7}$$

then

$$C_i^{\mathrm{T}} \mathbf{g}_1^{(i)} = [0 : \ldots : 0 : q_i(u_i - v_i) : v_i - u_i : q_i v_i : -v_i : 0 : \ldots : 0]^{\mathrm{T}}, \quad i = 1, \ldots, m-1,$$

$$C_m^{\mathrm{T}} \mathbf{g}_1^{(m)} = [q_m v_m : -v_m : 0 : \ldots : 0 : q_m(u_m - v_m) : v_m - u_m]^{\mathrm{T}}.$$

For any α_i, $i = 1, \ldots, m$, let

$$\mathbf{c} = \sum_{i=1}^{m} \alpha_i C_i^{\mathrm{T}} \mathbf{g}_1^{(i)}.$$

Then for $k = 1, \ldots, m$,

$$c_{2k-1} = \alpha_{k-1} q_{k-1} v_{k-1} + \alpha_k q_k (u_k - v_k), \tag{8}$$

$$c_{2k} = -\alpha_{k-1} v_{k-1} - \alpha_k (u_k - v_k), \tag{9}$$

where 0 subscripts are replaced by m.

Now redefine

$$u_i = \sum_{j=1}^{n_i} (y_{ij} - Y_i - q_i (X_{ij} - X_i))$$

or

$$u_i = n_i \frac{Y_i X_{i+1} - Y_{i+1} X_i}{X_{i+1} - X_i} + \sum_{j=1}^{n_i} y_{ij} - q_i \sum_{j=1}^{n_i} X_{ij}, \tag{10}$$

and

$$v_i = \sum_{j=1}^{n_i} p_{ij} (y_{ij} - Y_i - q_i (X_{ij} - X_i)). \tag{11}$$

Then with these changes to $\mathbf{u}$ and $\mathbf{v}$,

$$\mathbf{c} = \sum_{i=1}^{m} \alpha_i C_i^{\mathrm{T}} \mathbf{g}_3^{(i)}$$

is again given by (8) and (9).

Thus all entries required for (4) can be easily and efficiently computed. A summary of the computation leading to the solution of (4) and (3) is as follows:

Summary of computation of $\mathbf{d}$:

1. For $i = 1, \ldots, m$ compute

   $$q_i,\ p_{ij},\ j = 1, \ldots, n_i,\ r_i,\ s_i,\ t_i.$$

2. Set

   $$\alpha_i = \frac{1 + \lambda}{q_i^2 + 1 + \lambda}, \quad i = 1, \ldots, m.$$

3. Compute $G = \sum_{i=1}^{m} \alpha_i C_i^{\mathrm{T}} C_i = C^{\mathrm{T}} (I - D P^{-1} D) C$, using the equations for the non-zero elements of G which follow (5).
4. Compute $\mathbf{u}$ and $\mathbf{v}$ from (10) and (11), and from (8) and (9), compute

   $$\mathbf{c} = \sum_{i=1}^{m} \alpha_i C_i^{\mathrm{T}} \mathbf{g}_3^{(i)} = C^{\mathrm{T}} (I - D P^{-1} D) \mathbf{g}_3.$$

5. Set

   $$\alpha_i = \frac{-q_i}{q_i^2 + 1 + \lambda}, \quad i = 1, \ldots, m.$$

6. Compute $\mathbf{u}$ and $\mathbf{v}$ from (6) and (7), and from (8) and (9), compute

$$\mathbf{c} = \sum_{i=1}^{m} \alpha_i C_i^{\mathrm{T}} \mathbf{g}_1^{(i)} = C^{\mathrm{T}} D P^{-1} \mathbf{g}_1.$$

7. Solve (4) for $\mathbf{d}_2$.
8. Compute $C_i \mathbf{d}_2$, $i = 1, \ldots, m$ using (2) and hence compute $DC\mathbf{d}_2$.
9. Compute $\mathbf{d}_1$ using (3).

We can now readily implement a standard Levenberg–Marquardt algorithm for the problem (see, for example, [4]). A natural set of starting approximations is

$$X_i = x_i, \quad Y_i = y_i, \quad i = 1, \ldots, m,$$

$$X_{ij} = x_{ij}, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n_i,$$

and $\lambda = 1$. Adjustment of λ is as usual according to the value of the ratio

$$\frac{F(\mathbf{v}) - F(\mathbf{v} + \mathbf{d})}{F(\mathbf{v}) - \|J\mathbf{d} + \mathbf{g}\|^2}$$

at each iteration, as in [4]. Of course

$$\|J\mathbf{d} + \mathbf{g}\|^2 = \|\mathbf{g}_1 - \mathbf{d}_1\|^2 + \|\mathbf{g}_2 - \mathbf{d}_2\|^2 + \|\mathbf{g}_3 + D\mathbf{d}_1 + C\mathbf{d}_2\|^2,$$

which again can readily be computed.

Remark. The method given here can be extended to deal with the case where the straight line sides of the region are replaced by quadratic pieces, or other polynomial pieces, or indeed combinations of these.

3. Examples

The algorithm has been implemented according to the summary given at the end of the previous section and used to solve some model problems. To illustrate, we consider three examples. First suppose we have data for a triangular region, with 25 points on each side, the data generated by making random perturbations of points on the sides. The progress of the method for a particular data set is given in Table 1, where k is the iteration number, where F is the current value of $\mathbf{g}^{\mathrm{T}}\mathbf{g}$, where λ denotes the current value of the parameter λ and where $\|\mathbf{d}\|_\infty$ is the l_∞ norm of the current vector $\mathbf{d}$ satisfying (3) and (4). The performance of the algorithm was monitored by the size of $\|\mathbf{d}\|_\infty$. Fig. 1 is a sketch of the given data and the calculated triangle. The points marked with a diamond are the corner data points.

Next we generate data by starting with a particular 6-sided polygon and taking 19 data points on each side, again obtained by making random perturbations to exact values. The progress of the method for a particular data set is given in Table 2. Fig. 2 is a sketch of the given data and the calculated polygon.

Table 1
Example: $m = 3$, $p = 75$

k	F	λ	$\|\mathbf{d}\|_\infty$
1	6.3547	1.0	0.5108
2	2.4370	0.5	0.0722
3	2.1945	0.25	0.0068
4	2.1738	0.125	0.0008
5	2.1707	0.0625	0.0002
6	2.1700	0.0312	3.4×10^{-5}
7	2.1699	0.0156	7.6×10^{-6}
8	2.1698		

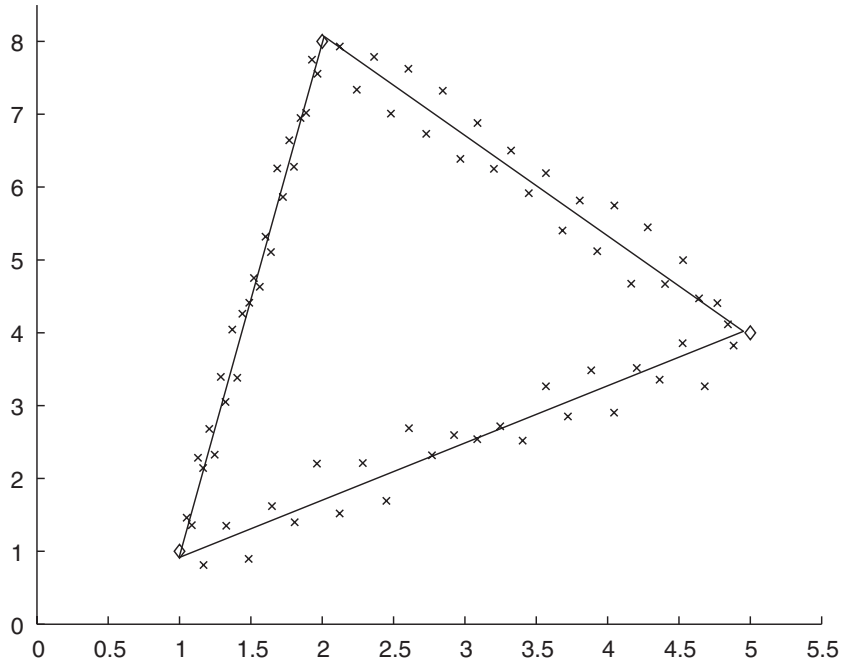

Fig. 1. Triangular region.

Table 2
Example: $m = 6$, $p = 114$

k	F	λ	$\|\mathbf{d}\|_\infty$
1	39.1131	1.0	2.1773
2	18.3373	0.5	0.4823
3	16.6276	0.25	0.1564
4	16.1255	0.125	0.0420
5	15.9939	0.0625	0.0099
6	15.9635	0.0312	0.0021
7	15.9569	0.0156	0.0004
8	15.9556	0.0078	8.7×10^{-5}
9	15.9553		

As a final example we generate data based on a square region with sides parallel to the co-ordinate axes. This example might be expected to cause problems for the method because of the denominators in the expressions for q_i. However, under the assumption that the data are not artificially fixed, but are genuinely contaminated by errors, zero denominators are unlikely even in this case, and the performance of the method is not necessarily affected, although convergence might be slowed. In particular, consider a square with corners at the points (1, 1), (1, 2), (2, 2), (2, 1), and with data obtained as for the other examples. Again 19 observations are assumed along each side. The performance of the method is illustrated in Table 3, and by Fig. 3, which shows the given data and calculated region.

4. Concluding remarks

The problem has been considered of fitting a polygon to data using a least squares criterion. The method proposed here is a Levenberg–Marquardt method for the nonlinear least squares problem, and the novelty lies in the way in which the structure of the problem is exploited to make the process efficient.

The approach taken here offers an alternative to others which have been suggested: see for example [8] There is evidence that it can work well in practice, and indeed it has a number of positive features. These include (1) solution

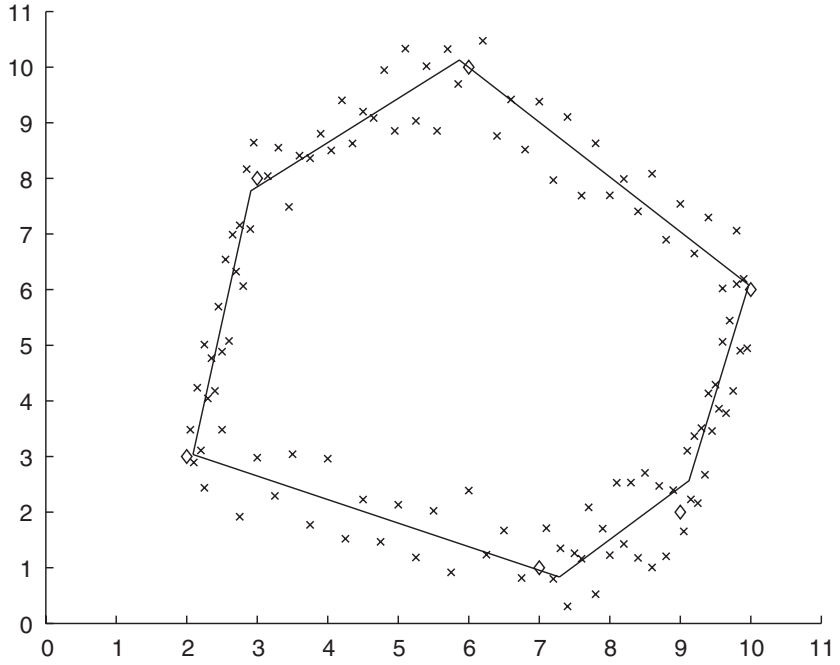

Fig. 2. 6-sided region.

Table 3
Example: $m = 4$, $p = 76$

k	F	λ	$\|\mathbf{d}\|_\infty$
1	80.8880	1.0	0.0028
2	18.8307	0.5	0.0049
3	3.7274	0.25	0.0097
4	0.2812	0.125	0.0100
5	0.1778	0.125	0.0064
6	0.1204	0.5	0.0012
7	0.0749	0.25	0.0015
8	0.0658	0.25	8.8^{-6}
9	0.0616	0.125	3.9^{-5}
10	0.0615		

of an unconstrained rather than a constrained problem, with reduced number of variables, (2) full exploitation of the underlying structure, (3) easy implementation with natural starting values, (4) excellent global convergence properties, and well established performance characteristics, (5) ready generalization, if required, to deal with the case when the straight line sides are replaced by higher degree polynomial pieces.

On the other hand, there are some negative features of the approach, which should also be mentioned. In particular, it is strongly co-ordinate dependent, although the problem is genuinely geometrical and invariant with respect to change of co-ordinates. For a co-ordinate free approach, see [7,8]. These methods also deal readily with the case when the straight lines are replaced by circular (or elliptical) arcs.

It is natural to ask how these different approaches compare in practice. Any valid comparison depends on many things, but especially on a proper basis for making such comparisons. We consider a comprehensive practical comparison of methods, which would enable meaningful conclusions to be drawn, as being beyond the scope of this paper.

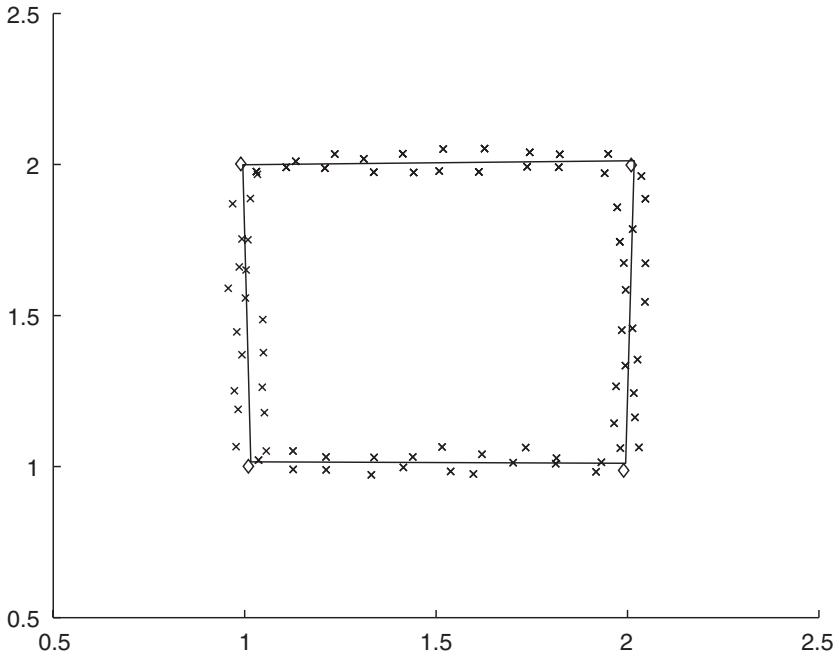

Fig. 3. 4-sided region.

References

[1] P.T. Boggs, R.H. Byrd, R.B. Schnabel, A stable and efficient algorithm for nonlinear orthogonal distance regression, SIAM J. Sci. Statist. Comput. 8 (1987) 1052–1078.
[2] L. Bondesson, G. Stahl, S. Holm, Standard errors of area estimates obtained by traversing and GPS, Forest Sci. 44 (1998) 405–413.
[3] R.E. Davis, F.S. Foote, J.M. Anderson, E.M. Mikhail, Surveying Theory and Practice, sixth ed., McGraw-Hill, New York, 1981.
[4] R. Fletcher, Practical Methods of Optimization, second ed., Wiley, New York, 1987.
[5] J.J. Moré, The Levenberg–Marquardt algorithm: implementation and theory, in: G.A. Watson (Ed.), Numerical Analysis, Dundee, 1977, Springer, Berlin, 1978, pp. 105–116.
[6] M.R. Osborne, Nonlinear least squares—the Levenberg algorithm revisited, J. Aust. Math. Soc. B 19 (1976) 348–357.
[7] A. Persson, Fitting polygons to coordinate data, Ph.D. Thesis, University of Umeå, Sweden, 2002.
[8] A. Persson, L. Bondesson, N. Börlin, Estimation of polygons and areas, Scand J. Stat. 33 (2006) 541–560.