A Variation of the Levenberg Marquardt Method.
An attempt to improve efficiency.

by

Evelyn Araneda

B.S, Geology (1999)

University of Chile

Submitted to the Department of Earth, Atmospheric
and Planetary Sciences
in Partial Fulfillment for
the Master of Science in Geosystems

at the

Massachusetts Institute of Technology

May, 2004
[June 2004]

Signature of Author .....
Department of Earth, Atmospheric and Planetary Sciences
May, 2004

Certified by .....
F. Dale Morgan
Professor of Geophysics
Thesis Advisor

Accepted by .....
Maria Zuber, E.A. Griswold
Professsor of Geophysics
Head of the Department

Content

Abstract

The Levenberg-Marquardt method is an efficient and popular damped least square technique. This method is a combination between the Gauss and the steepest gradient descent methods, where the amount of damping used in each iteration is central in establishing the behavior of the system. Further, the damping is determined by four parameters, whose optimum values vary from model to model. An inappropriate selection of the damping parameters could trigger a decrease in the rapidness of convergence, a convergence to a local minimum, or system instability. Therefore, finding proper values for these parameters is fundamental in the use of this method and implies a great deal of extra time. This lack of efficiency is considered a disadvantage in comparison to other techniques.

In an attempt to eliminate the use of arbitrary damping parameters as well as to improve the rapidness of the method, this work offers a new formulation for damping. Preliminary results show a positive behavior of the new method, which makes self-consistent automatic choices for the damping coefficients. An apparent improvement in efficiency is observed, despite the fact that a matrix determinant is included in the calculation of damping and more computational resources are involved. The savings in time due to the mechanization of the damping calculation seem to compensate for the extra resources. More study will be needed in order to validate or disqualify the proposed method.

Acknowledgments

I would like to express my gratitude to my advisor Dale Morgan for his support, guidance, and ideas throughout the development of this project. I really don't know what I would have done without his help. I would also like to especially thank Rama Rao for his time, patience, and kind advice in all aspects of this project.

I am deeply indebted to my friends Patricia and Marcos for their help and friendship during my time at MIT. I am also sincerely thankful to Eduardo, my boyfriend, for his endless love, patience and understanding. Finally, I would like to dedicate this work to my parents whose support and unconditional love have made me who I am today.

1. Introduction

1.1 Modeling methods

To predict the behavior of natural processes, modeling techniques try to fit a mathematical model to experimental data. For each model the best estimates of parameters are the ones that make the model the best match for the observed data. In order to find these best parameter approximations an error function, which is usually calculated as the median of the square residuals, is minimized.. The lowest point on this surface corresponds to the best approximation to the unknown parameters. Furthermore, in most natural processes, data are often nonlinear with respect to the parameters, so the search for a minimum error is usually iterative. These iterative procedures require the user to provide starting values for the unknown parameters from which the algorithm starts the minimization.

The least squares method is one of the most popular and powerful techniques for fitting mathematical models to experimental data. One advantage of this method is the broad range of functions that can be fit and the efficient use of data. However, the starting values must be reasonably close to the global minimum or the optimization procedure may not converge or may converge to local minima. In addition, a strong sensitivity to outliers can also cause the technique to converge to a local minimum.

There are two main approaches to the least squares estimation. On one hand, a Taylor expansion (Gauss or Gauss-Newton method) may be applied to the model, which corrects for parameters at each iteration assuming local linearity. On the other hand, the steepest gradient method iterates to the best estimate of the parameters going down the error function surface. Both methods are not optimal in the search of a global minimum, the first one because of divergence of the successive iterations, the second one because of slow convergence after a few iterations (Marquardt, D., 1963).

The popular Levenberg-Marquardt (LM) method (Levenberg, 1944; Marquardt, 1963) is a damped least square technique for nonlinear models where a positive constant (damping) is added to the diagonal of the Jacobian matrix in order to control the behavior of the system and prevent singularity. This method combines the advantages of the Gauss and the steepest gradient descent methods. If the damping used at one iteration reduces the error, the damping is divided by a reduction constant before the next iteration and the convergence to the solution is speeded. If the error increases then the damping is multiplied by an amplification constant, making the convergence slower but ensuring that a solution can be found. In this way, the method switches from one technique to the other smoothly. The LM method uses the method of the steepest descent when the results are far from the minimum. But as the solution approaches the minimum, the algorithm switches to the Gaussian method, which will tend to a zero step size when approaching the best fit.

Unfortunately, the LM technique presents some difficulties related to the calculation and rate of change of the damping, which is controlled by four main parameters: the initial damping, the amplification and reduction constants, and the minimum damping. An inappropriate choice of these parameters will cause the method to diverge or to converge too strongly or too slowly (Lampton, 1997). In addition, because the presence of damping increases the value of eigenvalues, preventing in this way the singularity of the system, the presence of the minimum damping is fundamental. This minimum value importantly affects the rapidness and stability of the method, and a proper amount is critical for the effectiveness of the LM technique.

Therefore, rapid convergence and the stability of the system are dependent upon an appropriate choice of the damping parameters, whose optimum value will vary from one model to another. Unfortunately, estimating these parameters often translates into trying the method several times before starting an experiment, which leads to a great deal of extra time spent in the modeling process.

1.2 Thesis Objectives

Taking advantage of the simplicity of the Levenberg-Marquardt algorithm, the present work will attempt to examine and improve the efficiency of the method. There are two primary goals:

(1) To minimize the number of parameters upon which the damping depends by obtaining automatically and self-consistently coefficients, thereby reducing the modeling time as well as the error associated with a large number of variables.

(2) To optimize the running time and the number of iterations the method takes to get to the solution.

2. Background

Most techniques trying to fit a mathematical model to experimental data focus on finding a global minimum in the error surface (Brent, 1973). When descending through a steep valley in the error surface it is best to use a small step size to avoid missing the minimum of the valley, even though this could also cause convergence to local minima. On the other hand when moving along a gently sloping area of the error surface it is more convenient to take large steps, otherwise it will take too long to converge.

2.1 Steepest Gradient Descent Method

The steepest gradient descent method works by making a step that is the negative gradient of the error times some constant. This means that in steep regions (where slow convergence is advisable) the algorithm moves quickly and in shallow regions (where fast convergence is more favorable) the method moves slowly. The iterative convergence to a solution from an initial guess of parameters (B_0) is represented by

$$\Delta B = - a \frac{\partial E(B)}{\partial B}$$

, where $E = T - f(z_k, B_j)$ is the estimation error function

T is the experimental data

$f(z_k, B_j)$ is the fitting function

z_k are the number of independent data, of dimension N

B_j are the parameters, of dimension M

a is a positive constant

The steepest gradient descent method works fine with simple models, but it fails when more complexity is added. In addition, convergence can take a long time because the method goes through most of the error surface missing the minima. Besides, the curvature of the error surface may not be the same in all directions, which implies more complexity in the error surface. For this method, including information about second order derivative (curvature of the error surface) would be useful. However, it is often too costly to compute second derivatives and methods incorporating this information are difficult.

2.2 Gauss Method

This method is based on the idea that nonlinear models can be approximated by linear functions through Taylor expansion when the system is close to a minimum in error space. Then the square error (E^2) will approximate a quadratic equation where the linear least square method can be used to find a minimum. If the approximation is valid the method will converge to a global minimum faster than the steepest gradient descent technique. The Gauss method may be expressed as follows (Dennis & Schnabel, 1983; Fletcher, 1987):

For $T = f(z_k, B_j)$

Assuming a local linearity of parameters and using a first order Taylor expansion we obtain,

$$T(z_k) = T_0(z_k) + \{\partial T(z_k) / \partial B_j\} \Delta B_j$$

then

$$\{\partial T(z_k) / \partial B_j\} \Delta B_j = T(z_k) - T_0(z_k)$$

, which in matrix notation is stated as

$$\mathbf{A}\,\Delta\mathbf{B} = \Delta\mathbf{T} \qquad (1)$$

, where $\mathbf{A\ (k,j) = \{\partial T(z_k) / \partial B_j\}}$ is the Jacobian matrix.

The error is defined as

$$E = \Delta T - A\ \Delta B$$

and the error squared as

$$E^2 = (\Delta T - A\ \Delta B)^T\ (\Delta T - A\ \Delta B)$$

Now, in order to minimize by least squares, the gradient is calculated and then equaled to zero:

$$\frac{\partial E^2}{\partial \Delta B} = -2\ A^T \Delta T + 2\ A^T A\ \Delta B = 0$$

then

$$A^T A\ \Delta B = A^T \Delta T$$

Finally, solving for the parameters vector ΔB, we get:

$$\mathbf{\Delta B = (A^T A)^{-1}\ A^T \Delta T} \qquad (2)$$

, which is solved iteratively to converge from some initial parameters (B_0) to the final solution.

The distinctive property of least squares problems is that we can consider A^TA as an estimation of the Hessian matrix ($Hij = \partial Hij/\partial Bi \partial Bj$). The Hessian approximation is exact only when f(z, B) is linear, which forces the nonlinear least squares method to rely on it only in regions where a linear approximation to f(z, B) is reasonable (close to

minima). In practice, this approximation is helpful only when the step size is small (Levenberg, 1944; Marquardt, 1963).

The main advantage of the Gauss technique is rapid convergence. However, the rate of convergence is highly sensitive to the starting location and the linearity around the starting location.

2.3 Levenberg Marquardt (LM) method

The Levenberg-Marquardt method performs an interpolation between the Gauss and the steepest gradient descent methods based upon the maximum neighborhood in which the truncated Taylor series gives an adequate representation of the nonlinear model (Marquardt, D., 1963).

In the algorithm a positive constant (damping) is added to the diagonal of A^TA in order to control the convergence of the method and provide an effective way to avoid the singularity of the system. In the former case damping will determine the rapidness of convergence, with large damping producing slow convergence and vice versa. In the latter case the presence of damping will artificially increase the eigenvalues improving the ill-conditioning of matrix A^TA. In the method the step to converge from an initial guess to a final solution is represented by:

$$\mathbf{\Delta B = (A^T A + \varepsilon^2 I)^{-1} A^T \Delta T} \qquad (3)$$

where B = Parameters to find

A = Jacobian matrix

ε^2 = Damping

T = Data

The general steps of the Levenberg-Marquardt algorithm are as follows:

1. Choose the initial parameters B_0
2. Choose the values for the positive constants α and β
3. Start with a large initial damping ε_0^2
4. Determine $A^T A$
5. Determine ΔB and calculate B_{i+1}
6. Check at each step
   - If $RMSE_i < RMSE_{i-1}$, then $\varepsilon^2_{i+1} = \varepsilon^2_i / \beta$
   - If $RMSE_i > RMSE_{i-1}$, then $\varepsilon^2_{i+1} = \alpha\ \varepsilon^2_i$
7. Maintain a minimum value for damping (ε^2_{min}) to ensure non-singularity of the matrix $Q = A^T A + \varepsilon^2 I$.

Nevertheless, there are some inconveniences in this method. The fact that damping is defined by four parameters, whose values must be decided before running the algorithm, strongly affects the rapidness and the way of convergence of the method. The parameters to be determined are the amplification (α) and reduction (β) constants, the initial damping and the minimum damping. If these parameters are not optimum for the model under study, the damping may become excessively large or small, affecting the way of convergence of the algorithm and thus the final results.

In this work it was found that the system is extremely sensitive to the value of minimum damping. A large minimum damping may cause slowness, a small minimum damping may cause singularity and in both cases it was likely to get trapped in a local minimum.

Therefore, the selection of adequate values for the damping parameters is a fundamental part of the standard LM method. Unfortunately, the only way to find these optimum values, besides previous knowledge based on past cases, is through experimentation, which implies extra processing time and loss in efficiency.

3. Motivation

From the Levenberg-Marquardt algorithm we have that

$$\mathbf{\Delta B = (A^T A + \varepsilon^2 I)^{-1} A^T \Delta T}$$

Let us define $\mathbf{H = (A^T A)}$; $\mathbf{G = \varepsilon^2 I}$; and $\mathbf{Q = H + G}$; where

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \ldots \\ h_{21} & h_{22} & h_{23} \ldots \\ h_{31} & h_{32} & h_{33} \ldots \\ \ldots & \ldots & \ldots \end{pmatrix}$$

In this work we will concentrate on maintaining the main property of damping, which is avoiding zero or small eigenvalues and thus the singularity of $A^T A$ by increasing the values of the diagonal of H. In order to do that, a minimum amount of damping is needed for those vectors with small diagonal values. We will also assume that as the damping increases the rapidness of convergence decreases.

Then, let us suppose there is a positive constant δ such as

$\varepsilon^2_1 = \delta_1 * (1/ h_{11})$

$\varepsilon^2_2 = \delta_2 * (1/ h_{22})$

$\varepsilon^2_3 = \delta_3 * (1/ h_{33})$

....

and the values of ε^2_j are such that they keep Q from being singular and make the algorithm converge to a solution. This concept was first proposed by Levenberg (1944)

where the constant δ_i are a system of positive constants or weighting factors expressing the relative importance of damping the different increments. In this work we will use one unique value for $\delta_i = \delta$. Then, we have that

$$G = \begin{pmatrix} \delta(1/h_{11}) & 0 & 0 & \dots \\ 0 & \delta(1/h_{22}) & 0 & \dots \\ 0 & 0 & \delta(1/h_{33}) & \dots \\ \dots & \dots & \dots & \end{pmatrix}$$

and

$$Q = \begin{pmatrix} h_{11} & h_{12} & h_{13} & \dots \\ h_{21} & h_{22} & h_{23} & \dots \\ h_{31} & h_{32} & h_{33} & \dots \\ \dots & \dots & \dots & \end{pmatrix} + \begin{pmatrix} \delta(1/h_{11}) & 0 & 0 & \dots \\ 0 & \delta(1/h_{22}) & 0 & \dots \\ 0 & 0 & \delta(1/h_{33}) & \dots \\ \dots & \dots & \dots & \end{pmatrix}$$

For example, for

$$\mathbf{H} = \begin{pmatrix} \mathbf{10^{-5}} & h_{12} & h_{13} & h_{14} \\ h_{21} & \mathbf{10^{-3}} & h_{23} & h_{24} \\ h_{31} & h_{32} & \mathbf{10^{1}} & h_{34} \\ h_{41} & h_{42} & h_{43} & \mathbf{10^{2}} \end{pmatrix}$$

we have that

$$G = \begin{pmatrix} \delta 10^{5} & 0 & 0 & 0 \\ 0 & \delta 10^{3} & 0 & 0 \\ 0 & 0 & \delta 10^{-1} & 0 \\ 0 & 0 & 0 & \delta 10^{-2} \end{pmatrix}$$

and

$$Q = \begin{pmatrix} (10^{-5}+\delta 10^{5}) & h_{12} & h_{13} & h_{14} \\ h_{21} & (10^{-3}+\delta 10^{3}) & h_{23} & h_{24} \\ h_{31} & h_{32} & (10^{1}+\delta 10^{-1}) & h_{34} \\ h_{41} & h_{42} & h_{43} & (10^{2}+\delta 10^{-2}) \end{pmatrix}$$

then

$$Q \simeq \begin{pmatrix} \delta 10^{5} & h_{12} & h_{13} & h_{14} \\ h_{21} & \delta 10^{3} & h_{23} & h_{24} \\ h_{31} & h_{32} & 10^{1} & h_{34} \\ h_{41} & h_{42} & h_{43} & 10^{2} \end{pmatrix}$$

We notice that in the proposed technique, the lowest values in the diagonal of H get the highest values of damping, ensuring in this way the non-singularity of the matrix. On the other hand, for the high values of the diagonal, the amounts of damping are low, which avoids the slowness of the method caused by unnecessary damping.

In this way this approach assigns variable damping to each term of the diagonal of H as a function of the diagonal values themselves, assigning significant damping only to the terms that need it and improving the rapidness of the algorithm.

However, because high values of damping ensure the stability of the matrix H but also make the convergence slower, the role of constant δ will be central for the method to behave efficiently. δ has to be chosen optimally so it maintains stability (keeping the significant value of damping where needed) and makes the convergence as fast as possible (reducing the value of damping as much as possible). The following sections will investigate possible values for this constant.

The main advantage of this method, if convergence is reached, would be the fact that there are not external parameters affecting the behavior of damping. The system would adjust the amount of damping internally based on the behavior of the Jacobian matrix, allowing the mechanization of the algorithm and an improvement in efficiency.

4. Testing

Following the concepts stated in the previous section, the damping will be defined as

$$\varepsilon^2_j = \delta^* (1/ h_{jj})$$

,where ε^2_j is the value of damping added to the j^{th} parameter,

h_{jj} is the value of j^{th} diagonal element of matrix H

δ is a positive constant

We know that large values of damping would produce a slower convergence and vice versa. Therefore a small value of damping will be preferable when singularity is not compromised.

When h_{jj} is small then the damping ε^2_j will tend to become large. In this way small eigenvalues will be avoided and stability maintained. On the other hand, if h_{jj} is large the value of damping will tend to become small, which will not decrease the rapidness of convergence. Then the constant δ needs to have the following property:

To reduce the value of ε^2_j when $1/ h_{jj}$ is large, so the algorithm does not become slow, while keeping the comparative large damping to avoid singularity.

Alternatively, we know that if one or more eigenvalues of the matrix $H = (A^TA)$ are equal to zero, then the **determinant of H** will also be zero (Gere & Weaver, 1983). Therefore, the determinant of a matrix is another indicator for the condition of a matrix and can be used to determine an adequate value of constant δ.

In the following sections, three different variations for the constant δ calculated based on the determinant of (A^TA) will be tried.

In general we consider a function $\mathbf{T = f\ (z_k\ ,\ B_j)}$

, where T = Data

z_k = Number of data, of dimension N

B_j = Number of parameters, of dimension M

and the estimation error is defined as $E = T - f\ (z_k\ ,\ B_j)$

The model selected as a characteristic representation of non-linear models is

$$T(z) = (B_1)*z + (B_2)*\exp(-B_3*z) \qquad (4)$$

where $B_1=3$; $B_2=2$; $B_3=1$.

This will be the model used for all experiments. Furthermore, as a way to understand the comparison between methods, no noise will be added. The data will be synthetically produced based on this model, where z is a set of 100 values ranging from 0 to 5.

In addition, as a way to compare the effectiveness of the experiments, several runs with the standard Levenberg-Marquardt algorithm were tried. The main parameters to compare in between methods will be whether there is convergence and the number of iterations and time taken to get to a solution.

The stopping and evaluation criteria applied to all experiments will be the following:

- The maximum number of iterations will be 3,000.
- The stopping criterion will be when

$$\frac{[RMSE_i - RMSE_{i-1}]}{RMSE_{i-1}} * 100 < 10^{-4}$$

or

$$RMSE_{i-1} < 10^{-8}$$

, whichever comes first.

- The final solution will be evaluated in terms of its closeness to the global solution (indicated by equation (4), where $B_1=3$; $B_2=2$; $B_3=1$).

4.1 Experiment 1:

The results for the experiments with the standard Levenberg-Marquardt algorithm are shown in Table 1. The first column shows the initial guesses of the model parameters, where $(B_1)_0$, $(B_2)_0$, and $(B_3)_0$ where assigned as equal for all the experiments and ranging from 5 to 100. The three following columns contain the quantities for the four damping parameters. In this case α and β were assigned as equal with values of 5 and 10, the initial damping (which in LM is a unique amount for all parameters) has values of 1 and 2, and the minimum damping was designated as 10^{-4} or 10^{-3}. All the quantities of the damping parameters were selected arbitrarily. The fifth column presents the solution to which the algorithm converged. Sol1 represents the global solution ($B_1=3$, $B_2=2$, $B_3=1$) and sol2 represents a local minimum solution. The last four columns are the number of iterations and time taken to converge to a solution and the average for each initial guess case.

Standard Levenberg-Marquardt Algorithm								
Initial Parameters	**Damping Constants**	**Initial Damping**	**Min Damp**	**Solution**	Iterations	**Average Iterations**	**Time**	**Average Time**
B1=B2=B3	**$\alpha = \beta$**	**g1=g2=g3**	**gm**		it	**<it>**	**t**	**<t>**
5	5	1	1.00E-04	glob sol	6	**7.0**	0.062	**0.07**
			1.00E-03	glob sol	6		0.063	
		2	1.00E-04	glob sol	8		0.078	
			1.00E-03	glob sol	8		0.078	
	10	1	1.00E-04	glob sol	7		0.078	
			1.00E-03	glob sol	7		0.062	
		2	1.00E-04	glob sol	7		0.079	
			1.00E-03	glob sol	7		0.078	
20	5	1	1.00E-04	glob sol	121	**49.4**	2.5	**0.93**
			1.00E-03	**loc sol**	13		0.17	
		2	1.00E-04	glob sol	66		1.8	
			1.00E-03	**loc sol**	58		0.75	
	10	1	1.00E-04		singular		singular	
			1.00E-03	glob sol	27		0.23	
		2	1.00E-04	**loc sol**	25		0.61	
			1.00E-03	glob sol	36		0.43	
40	5	1	1.00E-04	glob sol	65	**30.0**	1.4	**0.63**
			1.00E-03	**loc sol**	28		0.33	
		2	1.00E-04	glob sol	24		1	
			1.00E-03	**loc sol**	33		0.83	
	10	1	1.00E-04		singular		singular	
			1.00E-03	glob sol	14		0.34	
		2	1.00E-04	glob sol	22		0.16	
			1.00E-03	glob sol	24		0.38	
60	5	1	1.00E-04	glob sol	38	**32.7**	0.48	**0.61**
			1.00E-03	glob sol	13		0.36	
		2	1.00E-04		singular		singular	
			1.00E-03	**loc sol**	40		0.73	
	10	1	1.00E-04		singular		singular	
			1.00E-03	glob sol	25		0.52	
		2	1.00E-04	glob sol	38		1.016	
			1.00E-03	**loc sol**	42		0.562	
80	5	1	1.00E-04	glob sol	20	**51.4**	0.36	**1.42**
			1.00E-03	**loc sol**	21		0.36	
		2	1.00E-04	**loc sol**	194		6.6	
			1.00E-03	glob sol	30		0.48	
	10	1	1.00E-04		singular		singular	
			1.00E-03	**loc sol**	31		0.48	
		2	1.00E-04	glob sol	40		1.3	
			1.00E-03	glob sol	24		0.33	
100	5	1	1.00E-04	glob sol	52	**38.4**	1.4	**0.77**
			1.00E-03	glob sol	38		0.44	
		2	1.00E-04	glob sol	45		0.42	
			1.00E-03	glob sol	28		1.3	
	10	1	1.00E-04	glob sol	30		0.36	
			1.00E-03	**loc sol**	28		0.19	
		2	1.00E-04		singular		singular	
			1.00E-03	glob sol	48		1.3	

Table 1. Standard Levenberg-Marquardt Algorithm results.

Table 1 shows many combinations of damping parameters resulting in a local minimum solution (loc sol). A deeper analysis indicates that both, a slow convergence and a fast convergence (given by a large or small minimum damping) can lead to a local minimum solution.

Apparently, one of the most important factors affecting this behavior is the amount of minimum damping. For all the cases tried where the method converged to a local minimum solution, it was possible to find at least one choice of minimum damping for which the method converged to the global solution (glob sol). However, because finding that optimum minimum damping took considerable extra time, we will not consider those results in this work, so the comparison with other methods can be more equivalent.

In addition, the singularity of the system seems also to be related to the amount of minimum damping. All the cases of singularity found in the experiment were improved when the minimum damping was increased.

Therefore, the rapidness of the convergence seems to be an important factor to consider in terms of local minimum solutions and singularity. In the first case, both a large damping (slow convergence) and a small damping (fast convergence) may lead to a local minimum, whereas a very small damping may cause singularity. This fact should be considered at the time of selecting the appropriate value for the constant δ.

4.2 Experiment 2:

The first modification of the Levenberg-Marquardt algorithm will try to get a process independent from constants α and β and from the initial damping ε^2_0. We will keep the minimum damping as an arbitrary parameter, but we will try to eliminate it later.

For this experiment we will try the value

$$\delta = 1/\det Q$$

Because matrix Q includes the damping as a factor in its calculation, we will consider the determinant of Q an indicator of how the damping behaves. When the damping gets large, we will expect a higher value for the determinant of Q. Consequently, the constant δ will be comparatively small, decreasing the value of the damping in the following calculation. In this way excessive damping, due to a small value of h_{jj}, will be controlled. We expect that the reduction of the damping will not be excessive such that Q becomes singular. The case with a small damping and a comparable smaller determinant of Q will provide a way to control small values of damping.

In order to apply this concept, the values as well as the determinant of Q have to be obtained from the previous iteration to the one for which we are deriving. Then the constant δ will be expressed as,

$$\delta_i = 1/\det Q_{i-1}$$

and the damping will become

$$(\varepsilon^2_j)_i = (1/ h_{jj})_i * 1/\det Q_{i-1}$$

, where i is the iteration number and j is the column position in H and Q.

Therefore, δ_i will reflect the behavior of the damping in iteration (i-1) and control extreme values. If the damping was too large in iteration (i-1) then δ_i will reduce the damping in iteration i and vice versa.

The value of δ for the first iteration will be fixed at 1, which will make the damping for the first iteration $(\varepsilon^2_j)_0 = (1/ h_{jj})_0$. This initial value is dependant on the first matrix H and assigns large damping to small h_{jj}'s and vice versa.

In order to make the comparison between methods easier, the values for minimum damping (ε^2_{min}) will be the same ones used in the runs of standard Levenberg-Marquardt (Table 1).

The results of this experiment are shown in Table 2 .

Initial Parameters	1/det $Q_{(l-1)}$ * 1/hii (with minimum damping)					
B1= B2 = B3	Min Damp	Solution	Iterations		Time	
Bi	gm		it	Average	t	Average
5	1,00E-04	loc sol	9	9	0,08	0,09
	1,00E-03	loc sol	9		0,11	
20	1,00E-04	glob sol	36	33	0,28	0,38
	1,00E-03	glob sol	29		0,48	
40	1,00E-04	singular		19		0,16
	1,00E-03	glob sol	19		0,16	
60	1,00E-04	glob sol	25	25	0,19	0,20
	1,00E-03	glob sol	24		0,20	
80	1,00E-04	glob sol	27	30	0,33	0,34
	1,00E-03	loc sol	33		0,34	
100	1,00E-04	loc sol	171	171	2,20	2,20
	1,00E-03	singular				

Table 2. Experiment 1 results.

As observed on Table 2, the first column presents the initial model parameters, where as mentioned before $(B_1)_0=(B_2)_0=(B_3)_0$. The second column shows the values of minimum damping (10^{-4} or 10^{-3}), the third column shows the solution to which the algorithm converged to and the last four columns show the number of iterations and time (including the average for each initial guess case) taken for the method to get to a solution.

This first modification to the standard LM shows apparent good results. It seems to be faster than the standard Levenberg-Marquardt algorithm for most of the cases and it does not depend on an initial damping, or on constants α and β.

However, if now, in an effort to reduce the number of parameters in the algorithm, we eliminate the minimum damping, the results are much more discouraging. For most of the cases the Jacobian matrix becomes singular. The results of this case compared to the previous case and to the standard Levenberg-Marquardt method are showed in Table 3.

	Levenberg-Marquardt Algorithm		Experiment 1			
			1/det Q(i-1) * 1/hjj (with min damping)		1/det Q(i-1) * 1/hjj (no min damping)	
Initial Parameter	Average Iterations	Average Time	Iterations	Time	Iterations	Time
	it	t	it	t	it	t
5	7	0,07	9	0,09	6	0,14
20	49	0,93	33	0,38	12	0,16
40	30	0,63	19	0,16	26	0,20
60	33	0,61	25	0,20	singular	
80	51	1,42	30	0,34	singular	
100	38	0,77	171	2,20	singular	

Table 3. Comparison among LM method, Experiment 1 with min damping and Experiment 1 with no min damping.

In Table 3 the first column shows the different initial parameters cases, and the following columns present the number of iterations (or average) and the time (or average) taken for the standard LM, Experiment 2 with minimum damping, and Experiment 2 with no minimum damping respectively to get to a solution. It is indicated when the method does

not converge to a solution because it becomes singular (indicated as singular), as well as when the method does not converge to a solution before the maximum number of iterations is reached (indicated as > 3000).

The most likely reason for the results of Experiment 1 with no minimum damping is that constant δ is too small. More specifically, in the cases where the damping in iteration (i-1) was large, $\delta = 1/\det Q_{i-1}$ becomes small, decreasing the value of 1/hjj and making $(\varepsilon^2_j)_i$ excessively small.

This behavior contradicts the main property previously stated for the constant δ reducing the value of damping to improve rapidness while keeping a comparative large value to avoid singularity. However, because our main goal is to get an algorithm not dependent upon arbitrary parameters, we will test some other possibilities for δ.

4.3 Experiment 3:

After the results obtained in the previous attempt, we conclude that even though the first choice for damping did not work as well as expected when the constraint of the minimum damping was omitted, the general behavior of the experiment showed some interesting aspects.

The apparent good behavior of the method for the constrained case (minimum damping included) is a good indication that the inverse of the Jacobian determinant could still be a good candidate for constant δ. However, we need to find a way to increase the value of 1/detQ or alternative the value of $1/ h_{jj}$, so the damping increases and keeps stability.

In the next experiment, a set of runs intending to accomplish a more balanced damping will be analyzed. Because we cannot measure the exact amount of increase or decrease we need for the components of the damping, we will try all the possibilities related to the case already tried.

The choices of damping to be evaluated will be the following ones:

a) $(\varepsilon^2_j)_i = 1/ (h_{jj})_i * 1/\det H_i$

b) $(\varepsilon^2_j)_i = 1/ (q_{jj})_{i-1} * 1/\det Q_{i-1}$

c) $(\varepsilon^2_j)_i = 1/ (q_{jj})_{i-1} * 1/\det H_i$

We expect that at least one of these alternatives will balance the value of damping so the minimum damping will not be necessary. The results of these modifications comparing with the results of Experiment 1 with no minimum damping are shown in Table 4.

	Experiment 1			Experiment 2								
Initial Parameters	1/(hjj)i * 1/detQ			1/(hjj)i * 1/detH			1/(qjj)i-1 * 1/detQi-1			1/(qjj)i-1 * /detH		
	Iterations	time	Solution	Iterations	time	Solution	Iterations	time	Solution	Iterations	time	Solution
B1=B2=B3	it			it			it	t		it		
5	6	0,14	gdbsd	6	0,13	gdbsd	6	0,13	gdbsd	6	0,13	gdbsd
20	12	0,16	gdbsd	67	0,63	gdbsd			singular	12	0,16	gdbsd
40	26	0,20	gdbsd	18	0,17	locsol	14	0,17	locsol	40	0,38	gdbsd
60			singular	135	1,00	locsol			singular			singular
80			singular	744	6,4	locsol			singular	96	0,72	gdbsd
100			singular	>3,000					singular			singular

Table 4. Experiment 1 versus Experiment 2 (no minimum damping).

In Table 4, the first column represents the choices for initial parameters and then grouped in sets of three columns are the number of iterations and time taken to get to a solution as well as the final solution of Experiment 2 (with no damping) and Experiment 3 (three cases) respectively.

From the results we observe that the cases with the most encouraging results are the ones including $\delta=1/\det H$ as the factor for the diagonal inverses, where the fewest cases of

singular were obtained. The experiment where $(\varepsilon^2_{jj})_i = 1/ (q_{jj})_{i-1} * 1/\det H_i$ works well in most cases. The case where $(\varepsilon^2_{jj})_i = 1/ (h_{jj})_i * 1/\det H_i$ has slow convergence and not all solutions converge to the global minimum (glob sol).

In general, we observe two extremes, one where the convergence is too fast and singularity is produced (when $1/(q_{jj})_{i-1}$ or $1/\det Q_{i-1}$ are used for the damping) and the other one where the convergence is too slow (when $1/(h_{jj})_{i-1}$ or $1/\det H_i$ are included in the damping).

As a next step, we will try to improve the previous results by applying the square root to constant δ. For the cases where $\delta > 1$, this will produce a smaller δ, a decrease in the damping and the speeding of the process, whereas when $\delta < 1$, the square root will create a larger constant and the slowness of the method. We expect that at least one of the previous experiments will improve with this modification.

4.4 Experiment 4:

So, let us define

d) $(\varepsilon^2_j)_i = 1/ (h_{jj})_i * \text{sqrt}(1/\det Q_{i-1})$

e) $(\varepsilon^2_j)_i = 1/ (h_{jj})_{i-1} * \text{sqrt}(1/\det H_i)$

f) $(\varepsilon^2_j)_i = 1/ (q_{jj})_{i-1} * \text{sqrt}(1/\det Q_{i-1})$

g) $(\varepsilon^2_j)_i = 1/ (q_{jj})_{i-1} * \text{sqrt}(1/\det H_i)$

The results for this experiment are shown in Table 5, where the first column represents the choices for initial parameters and then grouped in sets of three columns are the number of iterations and time taken to get to a solution as well as the final solution of the four cases of this experiment.

Initial Parameter	$1/(h_{jj})$ * sqrt(1/detH)			$1/(h_{jj})$ * sqrt(1/detQ)			$1/(q_{jj})^{-1}$ * sqrt(1/detH)			$1/(q_{jj})^{-1}$ * sqrt(1/detQ)		
	Iterations	time	Solution	Iterations	time	Solution	Iterations	time	Solution	Iterations	time	Solution
B1=B2=B3	it			it			it	t		it	t	
5	7	0,13	glob sol	7	0,13	glob sol	7	0,16	glob sol	7	0,14	glob sol
20	877	6,7	loc sol	35	0,34	loc sol	17	0,27	glob sol	12	0,35	glob sol
40	7	0,16	loc sol	2922	32	loc sol	73	0,78	loc sol	17	0,19	loc sol
60	38	0,3	loc sol	6	0,11	loc sol	110	0,81	glob sol	21	0,22	loc sol
80	88	0,64	loc sol	5	0,13	loc sol	singular					singular
100	209	2,1	loc sol	5	0,09	loc sol	26	0,27	loc sol			singular

Table 5. Experiment 3 results.

In the case of the latest results, we observe that when $1/h_{jj}$ is used the method becomes extremely slow, with many of the solutions standing far from the global minimum. On the other hand, the method where $1/\det H_i$ is a factor of $1/q_{jj}$ seems much more promising. For this latter method, the number of iterations and running times are comparable to those of the Levenberg-Marquardt method.

We also observe that some of the results converge to a local solution (loc sol) and in one case becomes singular. We will consider these results as positive because they are very close to those obtained with the standard LM method and have the advantage of making self-consistent automatic choices for the damping coefficient. For this reason, we will select this algorithm for a more detailed analysis.

4.5 Modified Levenberg-Marquardt (LM) Algorithm:

As previously indicated, in the method selected to become the Modified LM Algorithm the damping is giving by

$$(\varepsilon^2_j)_i = 1/ (q_{jj})_{i-1} * \text{sqrt}(1/\text{det}H_i)$$

or alternatively

$$(\varepsilon^2_j)_i = 1/ ([A^TA]_{jj} + \varepsilon^2_j)_{i-1} * \text{sqrt}(1/\text{det}(A^TA)_i)$$

In order to study this method, we will run more trials with different initial guesses, so the behavior of the new algorithm becomes clearer. The results are showed in Table 6.

In Table 6 the first column represent the choices for initial parameters, the second and third columns show the number of iterations and time taken by the method to converge to a solution (it is indicated as singular when an experiment did not converge because singularity of the system), and the fourth column shows the solution to which the method converged to. *Glob sol* is the global solution whereas all the rest (*loc sol*) are local minima solutions.

1/qii * sqrt(1/detH)			
Initial Parameters	Iterations	time	solution
B1=B2=B3	it	t	
5	7	0,16	glob sol
10	9	0,16	**glob sol**
20	17	0,27	glob sol
30	40	0,39	glob sol
40	73	0,78	**loc sol**
50	95	0,83	glob sol
60	110	0,81	glob sol
70	113	0,86	glob sol
79	111	0,84	glob sol
80	singular		
81	110	0,84	glob sol
90	22	0,2	**loc sol**
100	26	0,27	**loc sol**
110	23	0,26	**loc sol**
120	81	0,69	glob sol
130	78	0,78	glob sol
140	102	0,84	glob sol
150	82	0,72	glob sol
160	40	0,34	**loc sol**
170	103	0,72	glob sol
180	56	0,53	**loc sol**
190	singular		
200	singular		
-5	14	0,16	glob sol
-10	singular		

Table 6. Modified Levenberg-Marquardt Algorithm results.

5. Results

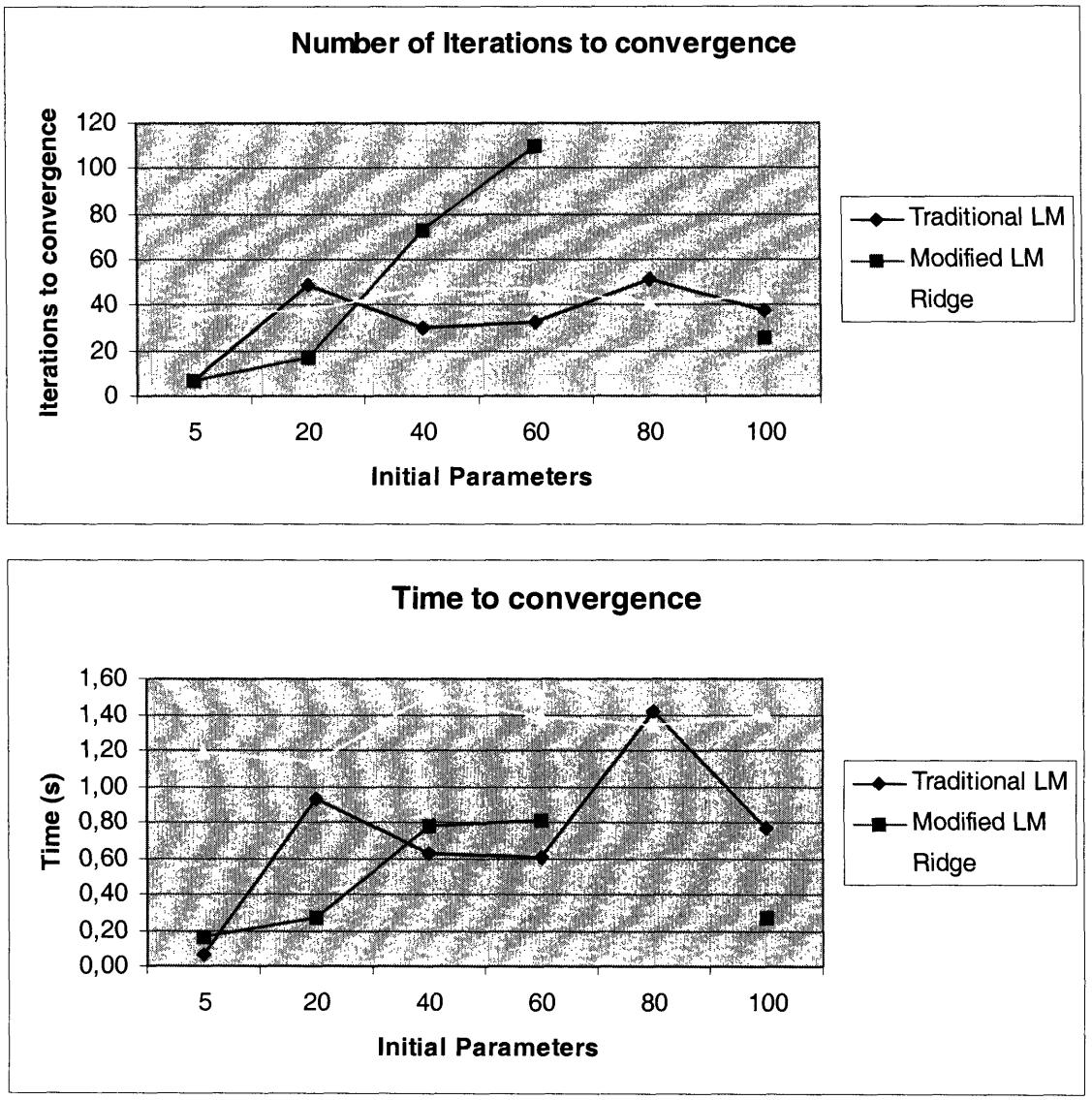
The results of the Modified LM Algorithm showed in Table 6 represent a considerable improvement from the standard Levenberg-Marquardt Algorithm. One main attribute making this new method preferable is the fact that the number of damping parameters that the user needs to determine before using this technique was reduced from four (damping constants, initial damping, and minimum damping) to one (initial damping, which was defined as $(\varepsilon^2_j)_0 = \delta_0 * (1/ h_{jj})_0$, where $\delta_0=1$.

The comparison between both methods can be appreciated in Table 7. In order to get a better perspective of the capacity of the new method, the number of iterations and time taken by the well-known, automatic, and self-consistent Ridge Regression method is also presented. For this latter case, the automatic algorithm used in the Earth Resources Laboratory at MIT was used, where the stopping criteria were the same used in the previous experiments (10^{-8} %) and the threshold for parameter change was 70%. Two graphs showing the results in Table 7 are also offered (Graph 1a and 1b).

Initial Parameter	Traditional LM			Modified LM			Ridge		
	Average Iterations	Average Time	solutions	Iterations	Time	solutions	Iterations	Time	solutions
5	7	0,07	glob sd	7	0,16	glob sd	36	1,19	glob sd
20	49	0,93	glob sd/ loc sd/ singular	17	0,27	glob sd	38	1,13	glob sd
40	30	0,63	glob sd/ loc sd/ singular	73	0,78	loc sd	47	1,50	glob sd
60	33	0,61	glob sd/ loc sd/ singular	110	0,81	glob sd	47	1,38	glob sd
80	51	1,42	glob sd/ loc sd/ singular	singular			41	1,34	glob sd
100	38	0,77	glob sd/ loc sd/ singular	26	0,27	loc sd	44	1,39	glob sd

Table 7. Comparison between standard LM and Modified LM methods.

In Table 7, the first column represent the different choices for initial guesses, the three following columns indicate the average number of iterations and average time to get to a solution as well as all the different solutions reached by the standard LM method. The three following columns show the number of iterations and the time taken by the modified LM to converge to a solution as well as the solutions obtained. Finally, the same information is also indicated for the case of the Ridge Regression in columns eight, nine, and ten.



Graph 1. (a) Number of iterations and (b) Time in seconds taken by each method to converge starting from different initial parameters.

From the results in Table 7 we notice that for initial parameters close to the global solution, the Modified LM method takes fewer steps to converge to the solution than the standard LM. However, this situation is reversed as the initial guesses get farther from the global solution. We also notice that the standard LM includes local minima as solutions in several experiments and became singular in one occasion. This was not improved with the Modified LM, which also includes local minima solutions and a singular case.

In addition, we observe that whereas the Modified LM develops a singular matrix when the initial guesses for the parameters are 80, the standard LM gets singular results when initial guesses equal 40. This could also be interpreted as an advantage of the Modified LM since this kind of behavior is more acceptable when we are far from the global solution than when we are close.

Finally, it is important to consider that, even though in general the standard LM takes less time to converge to a solution than the Modifies LM, the great amount of time invested in the selection of the damping coefficients is a major disadvantage. This problem is overcome in the Modified LM, which produced a decrease in the running time of the algorithm.

6. Conclusion

The main goal of this work was to adapt the standard Levenberg-Marquardt method to a more automatic system that does not depend on the selection of damping variables. The results of this work are promising. The Modified LM method eliminates most arbitrary damping coefficients, making self-consistent automatic choices of these parameters. In addition, the Modified LM method is comparable to the standard LM method in terms of rapidness, even though at starting points far from the global minimum, the standard LM works better.

Unfortunately, the calculations involved in this new method seem to have a high computational cost due to the calculation of a determinant. However, the savings in time and errors associated to the great amount of parameters involved in the standard LM method may compensate for the extra resources employed.

Finally, the method generated in the present work, even though it is preliminary, shows good potential as a starting point for further investigation in the improvement of the standard LM method.

7. Further Work

As indicated before, the work presented here has to be considered only as preliminary. More testing is necessary in order to validate, discard, or suggest an improved variant of the proposed Modified Levenberg-Marquardt Algorithm.

Three central courses of action are recommended in order to continue this study:

1. Run the proposed method considering noise in the model and compare the results to those of the standard Levenberg-Marquardt method, also including noise.

2. Run the proposed method with other models, with and without noise.

3. If the Modified LM algorithm seems to encounter problems in the above testing, new alternatives for the value of constant δ may be searched.

References

Brent, R. P., 1973. Algorithms for Minimization without Derivatives. Prentice-Hall, Englewood Cliffs, NJ, 1973.

Dennis J. E.; Schnabel R. B., 1983. Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Prentice-Hall, Englewood Cliffs, NJ, 1983

Fletcher, R., 1987. Practical methods of Optimization. New York, Willey.

Gere, J.M., Weaver. W.J., 1983. Matrix Algebra for Engineers Second Edition. Brooks/Cole Engineering Division.

Lampton, M., 1997. Damping-undamping Strategies for the Levenberg-Marquardt Nonlinear Least Square Method. Computers in Physics, Vol. 11, No 1, Jan/Feb 1997. pp 110-115.

Levenberg, K., 1944. A Method for the Solution of certain Nonlinear Problems in Least Squares. Quart Applied Mathematics, 2, 1944. pp 164-168.

Marquardt, D. W., 1963. An Algorithm for Least Squares Estimation of Nonlinear Parameters. Journal Society Industrial Applied Mathematics. Vol. 11, No. 2, June 1963. pp 431-441.

.

Abstract

The Levenberg-Marquardt method is an efficient and popular damped least square technique. This method is a combination between the Gauss and the steepest gradient descent methods, where the amount of damping used in each iteration is central in establishing the behavior of the system. Further, the damping is determined by four parameters, whose optimum values vary from model to model. An inappropriate selection of the damping parameters could trigger a decrease in the rapidness of convergence, a convergence to a local minimum, or system instability. Therefore, finding proper values for these parameters is fundamental in the use of this method and implies a great deal of extra time. This lack of efficiency is considered a disadvantage in comparison to other techniques.

In an attempt to eliminate the use of arbitrary damping parameters as well as to improve the rapidness of the method, this work offers a new formulation for damping. Preliminary results show a positive behavior of the new method, which makes self-consistent automatic choices for the damping coefficients. An apparent improvement in efficiency is observed, despite the fact that a matrix determinant is included in the calculation of damping and more computational resources are involved. The savings in time due to the mechanization of the damping calculation seem to compensate for the extra resources. More study will be needed in order to validate or disqualify the proposed method.