Technote: MERIS Advanced Water Algorithm Description of the parameters controlling the Levenberg-Marquardt algorithm

GKSS, H. Schiller

November 6, 2006

The MERIS Advanced Water Algorithm offers the possibility to improve the water constituent retrieval by iteratively minimizing the deviation of the modelled water reflectance spectrum from the measured one. The algorithm used for this minimization is the Levenberg-Marquardt algorithm (LMA).

The main idea of the LMA is to modify the minimum search strategy according to the recent success of the search. The search strategy switches between the steepest descent and the Gauss-Newton method.

- Far from the minimum the steepest descent (with a sufficiently small step) goes downhill but the convergence might be rather slow.
- Near enough the minimum the Gauss-Newton method leads to a fast convergence but away from the minimum it can go uphill.

The switching between the two methods is achieved by a damping factor μ:

$$(\mathbf{J}^T\mathbf{J} + \mu\mathbf{I})\mathbf{h} = -\mathbf{g} \tag{1}$$

Here $\mathbf{J}$ is the Jacobian of the model and $\mathbf{g}$ the gradient of the sum of squared deviations. The damping factor μ controls the calculation of the parameter step $\mathbf{h}$:

- For large values of μ we get $\mathbf{h} = -\frac{1}{\mu}\mathbf{g}$ which is a short step in the steepest descent direction. This is good if the current iterate is far from the minimum.
- For small values of μ we get a step according a quadratic approximation[1] which is good in the final stage of the iterations.

[1] Near the minimum $\mathbf{J}^T\mathbf{J}$ is a good approximation of the Hessian matrix.

In many textbooks (*e.g.* [1]) the recommented recipe for the choice of the damping factor is rather simple:

- if the step from (1) goes uphill then increase μ by a factor of 10 and try again (1),
- otherwise accept the step, decrease μ by a factor of 10 amd use (1) again.

The LMA implemented in 'MERIS Advanced Water Algorithm' follows a more elaborate scheme described in [2]. The ratio ρ between the actual and the predicted decrease in function value ('gain ratio', negative for uphill step) is used to update the damping factor.

if $\rho > 0$
$\mu := \mu * max\{\frac{1}{3}, 1-(2\rho-1)^3\}; \quad \nu := 2$
else
$\mu := \mu * \nu; \quad \nu := 2 * \nu$

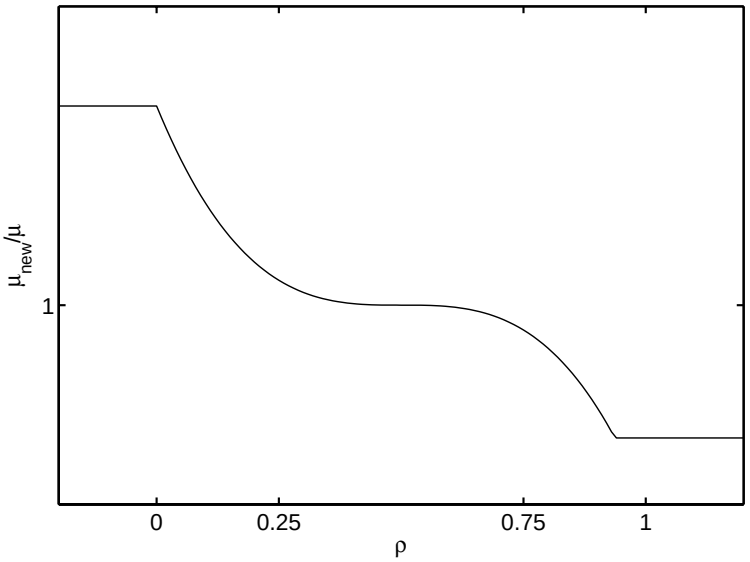

Figure 1: Updating of μ with $\nu = 2$.

The choice of the initial μ-value is related to the size of the diagonal elements of $\mathbf{J}^T\mathbf{J}$ by $\mu_0 = \tau * max(diag(\mathbf{J}^T\mathbf{J}))$. The algorithm is not very sensitive to the choice of τ. One should use a small value ($\tau = 10^{-6}$) if one has a very good start approximation. Otherwise use $\tau = 10^{-3}$ or even $\tau = 1$. The factor ν is inititialized to $\nu = 2$. Thus a series of consecutive failures results in rapidly increasing μ-values.

The rest of the parameters concerns the stopping criteria. The algorithm stops if

- the gradient $\mathbf{g}$ is sufficiently small (ε_1) or
- the change of the parameters is small [2] (ε_2) or
- the maximum number ($nitermax$)of iteration steps is reached.

References

[1] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical Recipes*. Cambridge University Press, 1992.

[2] K. Madsen, H. B. Nielsen and O. Tingleff, *Methods for Non-Linear Least Squares Problems*, 2nd Edition, April 2004, Informatics and Mathematical Modelling, Technical University of Denmark.

[2]The criterion reads $\|\mathbf{x}_{new} - \mathbf{x}\| < \varepsilon_2(\|\mathbf{x}\| + \varepsilon_2)$ where $\mathbf{x}$ is the vector of parameters. The expression gives a gradual change from relative step size ε_2 when $\|\mathbf{x}\|$ is large to absolute step size ε_2^2 if $\mathbf{x}$ is close to 0.