Comm. Korean Math. Soc. **11** (1996), No. 4, pp. 1147–1157

NUMERICAL METHODS USING TRUST-REGION APPROACH FOR SOLVING NONLINEAR ILL-POSED PROBLEMS

SUNYOUNG KIM

ABSTRACT. Nonlinear ill-posed problems arise in many application including parameter estimation and inverse scattering. We introduce a least squares regularization method to solve nonlinear ill-posed problems with constraints robustly and efficiently. The regularization method uses Trust-Region approach to handle the constraints on variables. The Generalized Cross Validation is used to choose the regularization parameter in computational tests. Numerical results are given to exhibit faster convergence of the method over other methods.

1. Introduction

Differential equations describing many real-life models must be solved numerically. This involves discretization and transformation to a system of nonlinear algebraic equations,

$$F(\mathbf{x}, y) = 0, \quad 0 \le \mathbf{x} \le \mathbf{b} \tag{1}$$

where the dimensions of vectors $F, \mathbf{x}, y$ depend on the number of discretization. Let $\mathbf{x} \in R^m$, F and $\mathbf{y} \in R^n$, where $m \le n$. Then, (1) becomes

$$\min_{\mathbf{x}} \|F(\mathbf{x}, y)\|_2^2, \quad 0 \le \mathbf{x} \le \mathbf{b}. \tag{2}$$

Problem (2) is said to be well-posed if: (i) for any $\mathbf{y} \in R^n$, there exists a solution $\mathbf{x} \in R^m$; (ii) the solution $\mathbf{x}$ is unique; (iii) the solution

Received April 11, 1996. Revised September 4, 1996.

1991 AMS Subject Classification: 49M35, 65F10.

Key words and phrases: nonlinear problems, ill-posed problems, constrained optimization.

Research supported by Korean Ministry of Education BSRI-96-1430.

$\mathbf{x}$ depends continuously on the data $\mathbf{y}$. Otherwise, the problem is ill-posed. Examples of nonlinear ill-posed problem are inverse problems of differential equations (i.e., parameter estimation) [6], inverse scattering, nonlinear Fredholm first kind equation [8].

The purpose of this paper is to solve nonlinear ill-posed problems efficiently, which, in most cases, is impossible with usual numerical methods. In many modeling problems, obtaining accurate solutions is very important. One way to achieve this is by discretizing the differential equations with small intervals. This, in turn, results a large number of equations and parameters. As the number of parameters and equations of problem (2) increases, it tends progressively difficult to solve the problem. Therefore, it is essential to develop a stable numerical method that provides accurate solutions.

Before we present numerical methods to solve (2) for any value of $\mathbf{y}$ that corresponds to $0 \le \mathbf{x} \le \mathbf{b}$, we discuss Trust-Region approach briefly. Trust-Region approach has been well-known for unconstrained optimization problems

$$\min_x f(x)$$

where $f : R^n \to R$ is smooth [1]. It usually generates a new iterate $x_{k+1} = x_k + s$ from the current approximation x_k by solving the subproblem

$$\min M(x) \;\; \text{subject to} \;\; \|s\|^2 \le \beta^2 \tag{3}$$

where $M(x)$ is a quadratic approximation to $f(x_k + s)$ and $\beta > 0$. If the solution s for (3) decreases the value of $f(x_k)$, viz., $f(x_k + x) < f(x_k)$, then $x_{k+1} = x_k + s$. Otherwise, the trust region parameter β is decreased and solve (3) until either $f(x_k + s) < f(x_k)$ or $\beta \approx 0$, when the iteration is terminated. In order to solve (2), we develop a method utilizing the Trust-Region approach that handles restriction on $\mathbf{x}$ properly. We first discuss how (2) has been solved by outlining Algorithm 2 in [rpt] and then show how the Trust-Region approach can be used to circumvent the ill-conditioning and bounds on $\mathbf{x}$.

Algorithm 1[11] ($\mathbf{y}$ fixed, $\mathbf{x}$ variable)

1. Given $\mathbf{y}$, assume an initial value for $\mathbf{x}$.

2. Compute $F(x,y)$. If $\|F(x,y)\|_2 < \epsilon$, where ϵ is a small number, or too many iterations have been done, then stop; else compute

$$\frac{dF}{d\mathbf{x}}\mathbf{e}_j = \frac{F(\mathbf{x}+\epsilon\mathbf{e}_j,\mathbf{y}) - F(x,y)}{\epsilon}, \quad j = 1, ..., m,$$

where $\mathbf{e}_j$ is the jth column of the identity matrix of order m.
3. Solve

$$\frac{dF}{d\mathbf{x}}\delta\mathbf{x} = -F(x,y) \tag{4}$$

for $\delta\mathbf{x}$, and let $\mathbf{x} = \mathbf{x} + \delta\mathbf{x}$. Go to step (2).

Step (3) of the above algorithm involves the solutions of a system of linear equations. Let $J \equiv \frac{dF}{d\mathbf{x}}$ and $\tilde{\mathbf{y}} \equiv -F(x,y)$. Since $\mathbf{x} \in R^m$ and J is a $n \times m$ matrix, solving equation (4) is equivalent to the linear least-squares problem

$$\min_{\delta\mathbf{x}} \|J\delta\mathbf{x} - \tilde{\mathbf{y}}\|_2^2. \tag{5}$$

One of well-known methods for solving (5) is Gauss-Newton method [9]. Instead of solving $J\delta\mathbf{x} = \tilde{\mathbf{y}}$ in **Algorithm** 1,

$$J^T J\delta\mathbf{x} = J^T\tilde{\mathbf{y}}$$

is solved for $\delta\mathbf{x}$. However, Gauss-Newton method fails to provide a solution for many problems due to its computational aspects [2]. It often exhibits large oscillations. One of the features of nonlinear ill-pose problems is that J in (5) is often ill-conditioned and therefore its condition number is very large. For these reasons, Gauss-Newton method and other methods for (5) are unsuccessful. Let us look at (5) carefully. Problem (5) is called discrete ill-posed [3.5] if the following conditions are satisfied:

1. The singular values of J decay gradually to zero.
2. The ratio between the largest and the smallest nonzero singular values is large.

For this problem, standard methods such as LU, Cholesky, QR factorization may give inappropriate results with large oscillations. One of the popular methods for ill-posed problems is called Tikhonov regularization method [12], from which many numerical methods for ill-posed problem have originated. In the next section, we mention a regularization method using generalized singular value decomposition and present a method using Trust-Region approach. We also modify the problem so that a meaningful solution can be determined. Then, we will present numerical results showing better performance of our suggested method over other methods.

2. Numerical Methods

In Tikhonov Regularization method, we define the regularized solution $\mathbf{x}$ as the following minimizer of the weighted combination of the residual norm and the side constraint

$$\min_{\delta\mathbf{x}}\{\|J\delta\mathbf{x}-\tilde{\mathbf{y}}\|_2^2+\gamma^2\|L(\delta\mathbf{x}-\delta\mathbf{x}^*)\|_2^2\},$$

where $\delta\mathbf{x}^*$ is an initial estimate and L is typically either identity matrix or a $p\times n$ discrete approximation of the $(n-p)$th derivative operator.

The successful numerical tools for nonlinear ill-posed problems are the Singular Value Decomposition (SVD) of J [4] and the Generalized Singular Value Decomposition (GSVD) of the matrix pair (J,L). It is known that the SVD reveals the difficulties associated with the ill-conditioning of the matrix J while the GSVD of (J,L) yields important insight into the regularization problem involving J and L [5]. For this reason, we discuss the GSVD and use it to present a new algorithm. The numerical method using the GSVD is called modified regularization method.

Modified Regularization Method (R1) Minimization problem (5) is modified to

$$\min_{\delta\mathbf{x}}\{\|J\delta\mathbf{x}-\tilde{y}\|_2^2+\gamma^2\|L\delta\mathbf{x}\|_2^2\} \tag{6}$$

where L is a discrete approximation to some derivative operator and γ is a free parameter. Let L be an $(m-1)\times m$ matrix of rank of $(m-1)$. Problem (6) is equivalent to solving the least squares solution to the overdetermined linear system [13]

$$\begin{pmatrix} J \\ \gamma L \end{pmatrix} \delta\mathbf{x} = \begin{pmatrix} \tilde{\mathbf{y}} \\ 0 \end{pmatrix},$$

which leads to the normal equation

$$(J^T J + \gamma^2 L^T L)\delta\mathbf{x} = J^T\tilde{\mathbf{y}}. \tag{7}$$

Since γ is a free parameter, equation (7) must be solved for various values of γ. But, it is pointed out in [13], that solving (7) with different γ's is not as effective as the regularization method solving

$$\min_{\delta\mathbf{x}}\{\|J\delta\mathbf{x}-\tilde{y}\|_2^2+\gamma^2\|\delta\mathbf{x}\|_2^2\}.$$

This method is a special case of (6) when $L=I$. Since we are interested in solving (6), we use the Generalized Singular Value Decomposition technique. The Generalized Singular Value Decompositions of J and L are

$$J = UD_aP^{-1}, \quad L = VD_cP^{-1},$$

where U and V are, respectively, $n\times n$ and $(m-1)\times(m-1)$ orthogonal matrices, $D_a = diag(a_1,...,a_m)$ and

$$D_c = \left[\begin{array}{cccc|c} c_1 & & & & \\ & \cdot & & & \\ & & \cdot & & 0 \\ & & & & \\ & & & c_{m-1} & \end{array}\right].$$

In view of the above equations, from (7) it follows that

$$(D_a^2+\gamma^2 D_c^T D_c)P^{-1}\delta\mathbf{x} = D_aU^T\tilde{\mathbf{y}},$$

which leads to

$$\delta\mathbf{x} = \sum_{i=1}^{m}\left(\frac{(U^T\tilde{\mathbf{y}})_i}{a_i + \gamma^2 c_i^2/a_i}\right)p_i, \tag{8}$$

where p_i is the ith column of P. If the modified regularization method is used to solve ill-posed problem, the following algorithm can be used.

Algorithm 2

1. L is given. Get an initial guess $\mathbf{x}_0$.
2. for $k = 1, \cdots$
3. Compute the Jacobian $J(\mathbf{x}_k)$.
4. Find the GSVD with $J(\mathbf{x}_k)$ and L.
5. Choose γ from GCV (Generalized Cross Validation) fun-ctional.
6. Compute $\mathbf{x}_{k+1} = \mathbf{x}_k + \delta\mathbf{x}$ using (8)
7. for $i = 1, \cdots, m$
   if $\mathbf{x}_{k+1}(i) > \mathbf{b}(i)$, then $\mathbf{x}_{k+1}(i) = \mathbf{b}(i)$.
   if $\mathbf{x}_{k+1}(i) < \mathbf{0}(i)$, then $\mathbf{x}_{k+1}(i) = \mathbf{0}(i)$.

Algorithm 2 is simple conceptually and easy to implement with respect to handling the bounds on $\mathbf{x}_k$. However, setting $\mathbf{x}_k$ to the value of the bounds when $\mathbf{x}_k$ is too large or too small may cause divergence of the iterates from a solution. This may happen more than often for solving ill-posed problems because the changes to iterates must be made carefully. This leads us to explore other possibilities, e.g., Trust-Region method, for the bounds on $\mathbf{x}_k$.

A Regularization Method using Trust-Region Approach (R2) We would like to develop a regularization method with the Generalized Singular Value Decomposition and Trust-Region approach. We first transform (6) to a constrained minimization problem. Let $J\mathbf{x}_k + \tilde{\mathbf{y}} \equiv z$. Since

$$J\delta\mathbf{x} - \tilde{y} = J(\mathbf{x}_{k+1} - x_k) - \tilde{\mathbf{y}},$$

we have

$$\min_{\mathbf{x}_{k+1}}\{\|J\mathbf{x}_{k+1} - z\|_2^2 + \gamma^2\|L\mathbf{x}_{k+1}\|_2^2\} \tag{9}$$
$$\text{subject to } \mathbf{0} \le \mathbf{x}_{k+1} \le \mathbf{b}$$

Notice that it minimizes on $\mathbf{x}_{k+1}$. It is based on the fact that we only know the bounds on $\mathbf{x}_k$'s. One way to solve (9) is to include Trust-Region approach for constraints on $\mathbf{x}_{k+1}$. Let $\mathbf{x}_{k+1} \equiv \bar{x}$. Then, the problem (9) is equivalent to the constrained problem:

$$\min_{\bar{\mathbf{x}}}\{\|J\bar{\mathbf{x}} - z\|_2^2 + \gamma^2\|L\bar{\mathbf{x}}\|_2^2\} \text{ subject to } \|\bar{\mathbf{x}}\|_2^2 \leq \alpha^2, \tag{10}$$

where $\alpha = \|\mathbf{b}\|_2$. If we use the GSVD described in the modified regularization method, we get

$$(D_a^2 + \gamma^2 D_c^T D_c)P^{-1}\bar{\mathbf{x}} = D_a U^T \mathbf{z}, \quad \text{subject to } \|\bar{\mathbf{x}}\|^2 \leq \alpha^2.$$

Our approach for solving the above problem is to use the first-order necessary conditions [7],

$$\begin{aligned} &(D_a^2 + \gamma^2 D_c^T D_c)P^{-1}\bar{\mathbf{x}} - D_a U^T \mathbf{z} + \lambda\bar{\mathbf{x}} = 0, \\ &\text{subject to } \ \|\bar{\mathbf{x}}\|^2 \leq \alpha^2, \ \ \lambda[\|\bar{\mathbf{x}}\|^2 - \alpha^2] = 0, \ \ \lambda \geq 0. \end{aligned} \tag{11}$$

For computation of a solution $\mathbf{x}_k$ of (11), we utilize a routine in NETLIB for constrained minimizations. An algorithm for the regularization method using Trust-Region approach is given as follows.

Algorithm 3

1. L is given. Get an initial guess $\mathbf{x}_0$.
2. for $k = 1, \cdots$
3. Compute the Jacobian $J(\mathbf{x}_k)$.
4. Find the GSVD with $J(\mathbf{x}_k)$ and L.
5. Choose γ from GCV functional.
6. Compute $(D_a^2 + \gamma^2 D_c^T D_c)P^{-1}$ and $D_a U^T \mathbf{z}$.
7. Solve (11) for $\mathbf{x}_{k+1}$ using a NETLIB routine.

In **Algorithm 3**, we do not set the values of $\mathbf{x}_{k+1}$ to $\mathbf{0}$ or $\mathbf{b}$ as **Algorithm 2**. Rather, we solve (11) with the bounds on $\mathbf{x}_k$. This is possible by transforming the problem (6) to the constrained minimization problem (11).

3. Computational Results

We apply the numerical methods in the previous section to a severely ill-posed problem arising in a modeling problem of physiology [10]. We describe the problem briefly. The differential equations for the model are:

$$\frac{d\mathbf{F}_{iv}}{dw} + \mathbf{J}_{iv}(w) = \mathbf{0},$$

$$\frac{d\mathbf{F}_{iv}C_{ik}}{dw} + \mathbf{J}_{ik}(w) = \mathbf{0}, \ i \neq 4,$$

$$\frac{d\mathbf{F}_{4k}}{dw} + \mathbf{J}_{4k}(w) = \mathbf{0},$$

$$D_{4k}\frac{d\mathbf{C}_{4k}}{dw} - \mathbf{F}_{4k} + \mathbf{F}_{4v}C_{4k} = \mathbf{0},$$

$$\sum_{i=1}^{4}\mathbf{J}_{iv}(w) = \mathbf{0}, \quad \sum_{i=1}^{4}\mathbf{J}_{ik}(w) = \mathbf{0}$$

where $\mathbf{J}_{iv}(w)$ and $\mathbf{J}_{ik}(w)$ are, respectively, volume and solute fluxes, $\mathbf{D}_{4k}$ are the diffusion coefficients and $\mathbf{F}_{4k}(w)$ are the axial solute flows, $\mathbf{J}_{iv}(w)$ and $\mathbf{J}_{ik}(w)$ are functions of only $\mathbf{C}_{ik}(w), \mathbf{C}_{4k}(w)$ and $\mathbf{x}(w)$(a parameter vector, e.g. water and solute permeabilities). This problem is transformed to a system of nonlinear equations using the trapezoidal rule. We can denote the system of nonlinear eqution as

$$\phi(\mathbf{x}, y) = \mathbf{0},$$

where $\mathbf{x} \in R^{18}$ and $\mathbf{y} \in R^{63}$. The direct problem is: Given $\mathbf{x}$, determine $\mathbf{y}$ from

$$\min_{\mathbf{x}} ||\phi(\mathbf{x}, y)||.$$

And the inverse problem is: Given $\mathbf{y}$, determine $\mathbf{x}$ from

$$\min_{\mathbf{x}} ||\phi(\mathbf{x}, y)||, \ \mathbf{0} \leq \mathbf{x} \leq \mathbf{b}. \tag{12}$$

See [10,11] for additional detail.

In order to test methods R1 and R2 for (12), as in [10,11], we used the model with parameters $\mathbf{x}m$ and $\mathbf{x}l$ given in [14]. Let us denote the $\mathbf{y}$'s that correspond to $\mathbf{x}m$ and $\mathbf{x}l$, respectively, by $\mathbf{y}m$ and $\mathbf{y}l$ which can be obtained from solving the direct problem. We use these values to provide initial values of $\mathbf{y}$ for the inverse problem. The intermediate values obtained from $\mathbf{y}m$ and $\mathbf{y}l$ are taken as inputs in (2) by

$$\mathbf{y} = \mathbf{y}l + \theta(\mathbf{y}m - \mathbf{y}l). \tag{13}$$

We expect to have $\mathbf{x}m$ as a solution of the inverse problem for $\mathbf{y}m$ obtained from $\theta = 1$ in (13) and $\mathbf{x}l$ for $\mathbf{y}l$ from $\theta = 0$. Therefore, in the computional tests, we initially vary θ from 0 to 1 and then check whether we can find a solution for other values of θ. We could get a solution for θ 0 to 2 as given in Table 1.

We have computed solutions using methods R1 and R2. In R1 and R2, we took

$$L = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & & & \\ & & & 1 & -1 \end{pmatrix}.$$

The term $\gamma^2\|L\bar{\mathbf{x}}\|_2^2$ in (10) is included to control the smoothness [5] of the solution $\bar{\mathbf{x}}$. The regularization parameter γ controls the weight given to $\|L\bar{\mathbf{x}}\|_2^2$ relative to $\|J\bar{\mathbf{x}} - z\|_2^2$. The parameter γ in (10) was chosen by using Generalized Cross Validation technique. It is very popular and successfully incorporated in many regularization methods such as Tikhonov Regularization [12] and Truncated Singular Value Decomposition [4]. The GCV functional is

$$G(\gamma) = \frac{\|J\bar{\mathbf{x}} - \mathbf{z}\|_2^2 + \gamma^2\|L\bar{\mathbf{x}}\|_2^2}{\{Tr[I - J(J^TJ + \gamma^2L^TL)^{-1}J^T]\}^2}.$$

In **algorithm 2** and **3**, we compute the $G(\gamma)$ for various values of γ and choose γ that gives the smallest value of (10).

All computations were performed on Sun Sparc-20. The initial guess for $\mathbf{x}$ was given as $\mathbf{x}l$. Without using the GCV functional, the convergence was much slower and sometimes failed. We used R1 because R1 has been the most successful method so far solving the problem mentioned above. We used 500 as the maximum number of iterations for R1 and R2. In the experiments, both R1 and R2 do not provide better solutions after 500 iterations. We obtained smaller values of $\|F\|$ for the various values of $\mathbf{y}$ by using R2. The values of γ for each θ are given in Table 1. The time difference of **Algorithm 2** and **Algorithm 3** is from step 6. R1 takes less time in each iteration than R2. However, R2 always converges to the values near $\|F\|$ shown in Table 1 earlier in iterations, say, after 5 iterations than R1. That is, R2 converges faster than R1 in $\|F\|$. Moreover, R2 gives more accurate solutions as shown in Table 1. Hence, we can conclude that R2 shows better performance to find a solution than R1.

Table 1. **R1** vs. **R2**

θ	R1			R2		
	Iter.	$\|F\|$	γ	Iter.	$\|F\|$	γ
0.10	500	4.88851234-04	4.0363-08	500	9.93059090-05	4.0395-09
0.25	500	7.75893815-04	3.9897-08	500	1.19343469-04	3.9919-09
0.50	500	1.14674364-03	3.8983-08	500	1.60958269-04	3.9005-09
0.75	500	1.48632481-03	3.8658-08	500	2.07426879-04	3.8687-09
1.00	500	1.84542734-03	3.8787-08	500	2.56120387-04	3.8807-08
1.50	500	2.42974140-03	3.9110-08	500	3.36237531-04	3.9120-09
2.00	500	3.29750461-03	3.9430-08	500	4.58714961-04	3.9436-09

References

1. J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall Inc., Englewood Cliffs, New Jersey (1983).
2. C. Fraley, *Computational behavior of Gauss-Newton Methods*, SIAM J. Sci. Stat. Comput. **10** (1989), 515-532.
3. V. N. Joshi and R. P. Tewarson, *On Solving Ill-conditioned Systems of Linear Equations*, Transactions of New York Academy of Sciences **34** (1972), 565-571.
4. P. C. Hansen, *The Truncated SVD as a method for Regularization*, BIT **27** (987), 543-553.
5. Per Christian Hansen, *Regularization Tools: A Matlab Package for Analysis and Solution of Discrete Ill-posed Problem*, Numerical Algorithms **6** (1994), 1-35.
6. C. Kravaris and J. H. Seinfeld, *Identification of Parameters in Distributed Parameter Systems by Regularization*, SIAM J. Control Optim. **23**, 217-241 yr 1985.

7. H. W. Kuhn and A. W. Tucker, *Nonlinear Programming in proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, J. Neyman(editor), University of California Press, Berkeley and Los Angeles, Calif . (1961), 481-492.
8. F. O'Sullivan and G. Wahba, *A Cross Validated Bayesian Retrieval Algorithm for Nonlinear Remote Sensing Experiments*, J. Comput. Phys. **59** (1985), 551-555.
9. J. Stoer and R Bulirsch, *Numerical Analysis*, Springer-Verlag, New York, 1980.
10. R. P. Tewarson, *Inverse Problem for Kidney Concentrating Mechanism Analyses*, Appl. Math. Lett. **4** (1993), 63-66.
11. ———, *Models of Kidney Concentrating Mechanism: Relationship Between Core Concentrations and Tube Permeabilities*, Appl. Math. Lett. **6** (1993), 71-73.
12. A. N. Tikhonov, *Solution of Incorrectly Formulated Problems and the Regularization Method*, Dokl. Adak. Nauk. SSSR, **151** (1963), 501-504.
13. J. M. Varah, A Practical Examination of Some Numerical Methods for Linear Discrete Ill-Posed Problems, SIAM Rev. **21** (1979), 100-111.
14. A. S. Wexler, R. E. Kalaba and D. J. Marsh, *Three-dimensional Anatomy and Renal Concentrating Mechanism I. Modeling Results* Am. J. Physiol., 260, (Renal Fluid Electrolyte Physiol. 29), F368-F383, (1991)

Department of Mathematics
Ewha Womans University
Dahyun-Dong, Seoul 120-750, Korea
e-mail: skim@mm.ewha.ac.kr