BIT
2000, Vol. 40, No. 4, pp. 001–004

0006-3835/00/4004-0001 $15.00

MINIMIZATION OF LINEAR FUNCTIONALS DEFINED ON SOLUTIONS OF LARGE-SCALE DISCRETE ILL-POSED PROBLEMS*

L. ELDÉN[1], P. C. HANSEN[2] and M. ROJAS[3]†

[1] *Department of Mathematics, Linköping University, S-581 83 Linköping, Sweden. email: laeld@math.liu.se*

[2] *Informatics and Mathematical Modelling, Technical University of Denmark, Building 321, DK-2800 Kgs. Lyngby, Denmark. email: pch@imm.dtu.dk*

[3] *Department of Mathematics, Wake Forest University, P.O. Box 7388, Winston-Salem, NC 27109, USA. email: mrojas@mthcsc.wfu.edu*

Abstract.

The minimization of linear functionals defined on the solutions of discrete ill-posed problems arises, e.g., in the computation of confidence intervals for these solutions. In 1990, Eldén proposed an algorithm for this minimization problem based on a parametric-programming reformulation involving the solution of a sequence of trust-region problems, and using matrix factorizations. In this paper, we describe MLFIP, a large-scale version of this algorithm where a limited-memory trust-region solver is used on the subproblems. We illustrate the use of our algorithm in connection with an inverse heat conduction problem.

AMS subject classification: 65F22

Key words: Discrete ill-posed problems, confidence intervals, large-scale algorithms, trust regions.

1 Introduction

This work is concerned with the minimization of a linear functional defined on the solution of a discrete ill-posed problem. The mathematical formulation of this problem takes the following form: let A be an ill-conditioned coefficient matrix of dimensions $m \times n$, let b be the corresponding right-hand side, and let w be an arbitrary vector of unit length ($\|w\|_2 = 1$), then compute the solution x_w to the problem

$$\min w^T x \qquad \text{subject to} \qquad \|A\,x - b\|_2 \le \epsilon, \quad \|x - d\|_2 \le \delta. \tag{1.1}$$

*Received xxx. Revised xxx. Communicated by xxx.

†This work was partially supported by grant no. 9901581 from the Danish Natural Science Research Foundation, and by the Science Research Fund of Wake Forest University.

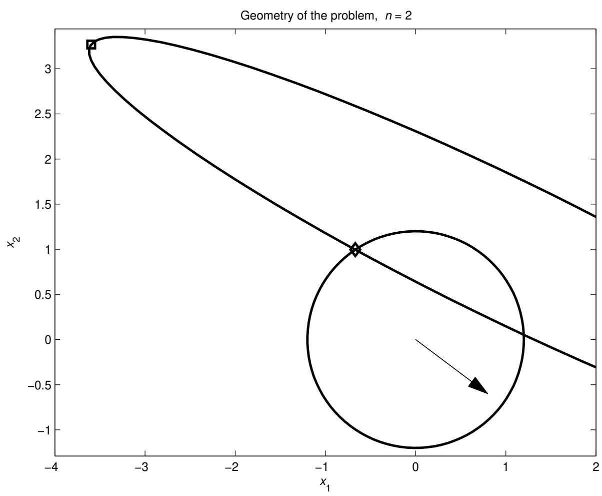

Figure 1.1: The circle and the ellipsoid are defined by $\|x - d\|_2 = \delta$ and $\|A\,x - b\|_2 = \epsilon$, respectively; here $d = 0$ and the ellipsoid is centered at $(1,1)^T$. The arrow illustrates the vector $w = (0.8, -0.6)^T$. The solution to the problem $\min\, w^T x$ s.t. $\|A\,x - b\|_2 \le \epsilon$ is located at the square – far from the origin – while the solution x_w to (1.1) is located at the diamond.

Here ϵ and δ are two positive constants that are used to stabilize the computation of the solution vector x. The vector d contains a priori knowledge of the solution, and may be set to the zero vector. Problems of the form (1.1) arise in the numerical treatment of inverse problems. One important application is the computation of confidence intervals for the solution of inverse problems [8]. Other applications include the computation of windowed or cumulative profiles of deconvolved data [9].

A related problem is that of computing stabilized/regularized solutions to discrete ill-posed problems, e.g., by solving the problem

$$\min \|A\,x - b\|_2 \qquad \text{subject to} \quad \|x - d\|_2 \le \delta. \tag{1.2}$$

In this problem, δ acts as a regularization parameter that controls the "size" of the solution. The solution to (1.2) is identical to the Tikhonov solution, formally given by

$$x_\lambda = \left(A^T A + \lambda^2 I\right)^{-1} A^T b, \tag{1.3}$$

for a particular value of the Lagrange parameter λ.

Looking at the two problems (1.1) and (1.2), it may appear at first that one constraint – namely $\|A\,x - b\|_2 \le \epsilon$ – would be enough to stabilize the first problem. However, this is not so; without the constraint $\|x - d\|_2 \le \delta$ the vector x can have a large norm, and this happens when w points in a direction for which the ellipsoid $\|A\,x - b\|_2 = \epsilon$ is very elongated. The purpose of the constraint $\|x - d\|_2 \le \delta$ is thus to ensure that this does not happen – at the expense of the possible non-existence of a solution (depending on the choice of δ and d). Figure 1.1 illustrates these points.

Several papers deal with algorithms for solving (1.1). Our work builds on an algorithm developed by Eldén [3], based on the reformulation of (1.1) as a parametric programming problem involving the solution of a sequence of trust-region problems. While the algorithm in [3] uses matrix factorizations, our algorithm is designed for large-scale computations where such factorizations are prohibitive. We achieve this by using the implementation [11] of the limited-memory algorithm LSTRS by Rojas, Santos and Sorensen [10] to solve the trust-region subproblems. This method was chosen for its large-scale features as well as for its ability to handle the singularities arising in discrete ill-posed problems [12].

Another important issue in our work is the design of a robust implementation of the algorithm, suited for general use (e.g., in connection with the Regularization Tools package [5]). We want to be able to check the existence of a solution to a given problem, and we want to guarantee that our implementation always computes a solution if it exists. The resulting method is the algorithm MLFIP described in this work.

This paper is organized as follows. In Section 2 we briefly study the properties of the solution of (1.1) from the regularization point of view. In Section 3 we summarize the basic algorithm from [3]. The core of this algorithm is the computation of the smallest root of a special nonlinear function, and in Section 4 we describe our algorithm for this problem. In Section 5 we discuss some implementation issues of the algorithm. Finally, in Section 6 we demonstrate the use of MLFIP in connection with the sideways heat equation problem.

2 Properties of the Solution

The purpose of this section is to discuss certain properties of x_w, regarded as a regularized solution of (1.1). Consider first the left plot in Fig. 2.1, which shows the geometry of the problem for $n = 2$. The gray region is the set $\mathcal{S}$ of "feasible points" given by

$$\mathcal{S} = \{x \mid \|A\,x - b\|_2 \le \epsilon \ \wedge \ \|x - d\|_2 \le \delta\} \tag{2.1}$$

while the two black circles represent the set

$$\mathcal{S}^* = \{x \mid \|A\,x - b\|_2 = \epsilon \ \wedge \ \|x - d\|_2 = \delta\}. \tag{2.2}$$

Moreover, the thick solid line represents all the Tikhonov solutions x_λ given by (1.3) for $0 \le \lambda < \infty$. Any solution x_w to (1.1) lies on the boundary of $\mathcal{S}$, and coincides with a Tikhonov solution only for two particular instances of the vector w. For all other w, the solution x_w differs from a Tikhonov solution – hence it is not "optimal" in the sense of balancing the norms $\|x - d\|_2$ and $\|A\,x - b\|_2$, and it is not necessarily smooth in the sense of being dominated by the principal singular vectors.

Next consider the right plot in Fig. 2.1 which shows a generic L-curve, i.e., a log-log plot of the norm of the Tikhonov solution $\|x_\lambda\|_2$ versus the corresponding

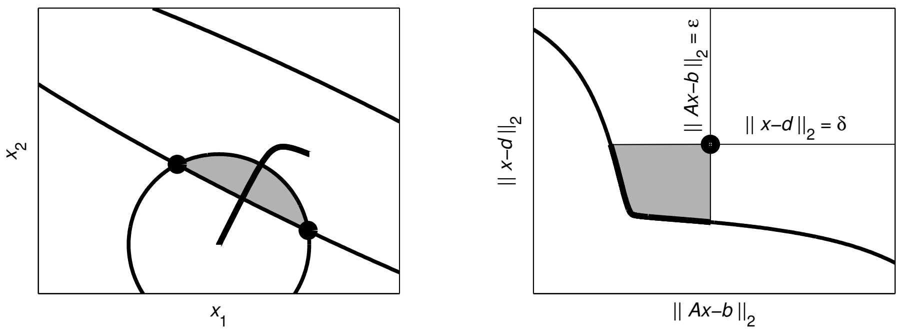

Figure 2.1: The gray region and the black circles represent the sets $\mathcal{S}$ and $\mathcal{S}^*$, respectively, while the thick line represent all the Tikhonov solutions.

residual norm $\|A\,x_\lambda - b\|_2$. It is well known (see, e.g., Section 4.6 in [6]) that for an arbitrary $x \in \mathbb{R}^n$, the point $(\|A\,x - b\|_2, \|x\|_2)$ must lie on or above the L-curve. The thin vertical and horizontal lines represent vectors x for which $\|A\,x - b\|_2 = \epsilon$ and $\|x - d\|_2 = \delta$, respectively.

Moreover, the gray region represents all the solutions in the set $\mathcal{S}$, while all the points that represent the solutions in the set $\mathcal{S}^*$ coincide in a single point (the black circle). We note that all solutions x_w to (1.1) lie on the upper or right boundary of the gray region that represents the set $\mathcal{S}$. Again, this illustrates the fact that the solutions x_w are different from the Tikhonov solutions x_λ.

To illustrate this point further, we consider the test problem shaw from [5]. Let x^{exact} denote the exact solution, and let $b = A\,x^{\mathrm{exact}} + e$ where the elements of the

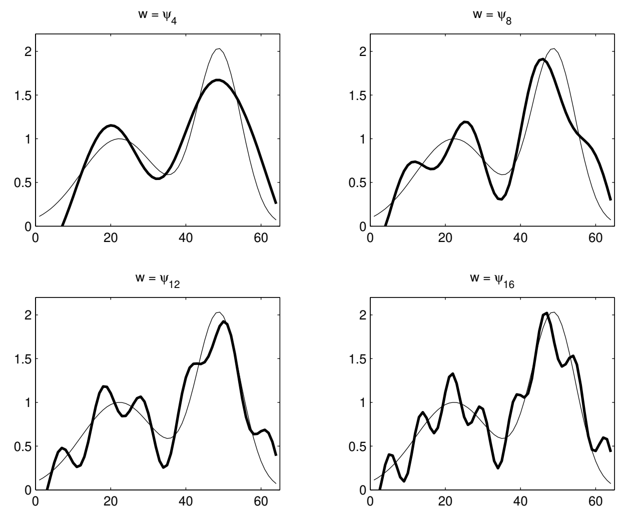

Figure 2.2: The exact solution (thin line) and the solution x_w to the shaw test problem for four different choices of w.

perturbation e are from a Gaussian distribution with zero mean and standard deviation such that $\|e\|_2/\|b\|_2 = 10^{-2}$. We choose $d = 0$, $\delta = \|x^{\rm exact}\|_2$ and $\epsilon = \|e\|_2$. The vector w is the normalized ith basis vector ψ_i in the discrete cosine transform for $i = 4$, 8, 12 and 16. The corresponding four solutions are shown in Fig. 2.2. Clearly, the solution x_w can have a highly oscillatory component, depending on the choice of w.

The conclusion is that the solution x_w to (1.1), albeit regularized, cannot be expected to be smooth in the same manner as the Tikhonov solutions.

3 The Basic Algorithm

As shown in [3], problem (1.1) can be reformulated in such a way that the core computational problem is a parametric programming problem, which is easier to handle numerically than the original problem. According to Theorem 2.5 in [3], the solution $\phi = w^T x$ to (1.1) is the *smallest* solution of the nonlinear equation

$$L(\phi) = \epsilon^2 \tag{3.1}$$

where the function $L(\phi)$ is defined as

$$L(\phi) = \min_{x\in\mathcal{T}(\phi)} \|A\,x - b\|_2, \quad \mathcal{T}(\phi) = \left\{x \mid \|x-d\|_2 \le \delta,\ w^T x = \phi\right\}. \tag{3.2}$$

Hence, in order to solve the original problem (1.1) we use a root finder to compute the smallest root of $L(\phi) - \epsilon^2$. Note that the computation of the function values of $L(\phi)$ requires the solution of the parametric programming problem in (3.2).

This problem, in turn, is solved by first removing the linear constraint $w^T x = \phi$ and then solving the resulting quadratically constrained least squares problem. Following [3], we introduce the orthogonal matrix V whose first column is w (recall that w has unit length):

$$V = (\,w\,,\overline{V}\,)$$

and we notice that V can be chosen as a single Householder transformation. Introducing the quantities

$$\begin{aligned}\overline{A} &= A\,\overline{V}\\ \overline{b} &= b - \phi\,A\,w\\ \overline{d} &= \overline{V}^T d\\ \overline{\delta}^{\,2} &= \delta^2 - (\phi - w^T d)^2,\end{aligned}$$

the solution to the problem in (3.2) is given by

$$x = V \begin{pmatrix}\phi\\ \overline{y}\end{pmatrix} = \phi\,w + \overline{V}\,\overline{y},$$

where the vector $\overline{y}$ solves the following least squares problem with a quadratic constraint:

$$\min \|\overline{A}\,\overline{y} - \overline{b}\|_2 \qquad \text{subject to} \qquad \|\overline{y} - \overline{d}\|_2 \le \overline{\delta}. \tag{3.3}$$

Therefore, the evaluation of $L(\phi)$ is equivalent to solving the trust-region subproblem (3.3).

4 Computing the Smallest Root

Assuming that we can evaluate the function $L(\phi)$ accurately, we need a robust algorithm for computing the smallest root of the nonlinear equation (3.1), including a reliable test for the existence of such a root. The following theorem is central.

THEOREM 4.1. *The function $L(\phi)$ given by (3.2) is defined for $\phi_* \le \phi \le \phi^*$, where*

$$\phi_* = w^T d - \delta, \qquad \phi^* = w^T d + \delta.$$

Moreover, $L(\phi)$ is continuous and convex in this interval.

PROOF. The function L is well defined as long as $\delta^2 - (\phi - w^T d)^2$ is nonnegative. It is straightforward to see that this holds only for $\phi \in [w^T d - \delta, w^T d + \delta]$. The continuity of L was proved in [3]. We will now show that this function is convex. First, assume that A has full column rank so that the ellipsoid $\{x \mid \|Ax - b\|_2 \le \epsilon\}$ is strictly convex. Take two points, ϕ_1 and ϕ_2, in the interval $[\phi_*, \phi^*]$, and let the corresponding, unique minimizers of $L(\phi)$ be denoted by x_1 and x_2. Define

$$\phi_\tau = (1-\tau)\phi_1 + \tau\phi_2,$$

for some $\tau \in [0,1]$. Then

$$\bar{x} := (1-\tau)x_1 + \tau x_2 \in \mathcal{T}(\phi_\tau),$$

since the sphere is convex and $w^T x$ is linear in x. It follows that

$$\begin{aligned} L(\phi_\tau) &= \min_{x \in \mathcal{T}(\phi_\tau)} \|A\,x - b\|_2 \\ &\le \|A\,\bar{x} - b\|_2 \\ &\le (1-\tau)\|A\,x_1 - b\|_2 + \tau\|A\,x_2 - b\|_2 \\ &= (1-\tau)L(\phi_1) + \tau L(\phi_2). \end{aligned}$$

If A is rank deficient and w is not orthogonal to the null-space of A, then the minimizers are unique and the above argument still holds. Finally, if w is orthogonal to the null-space, then the minimizers are not unique. However, the above proof can be applied to any pair of minimizers. Thus we have proved that $L(\phi)$ is convex. $\square$

It follows from this theorem that, depending on ϵ, there is either zero, one or two solutions to the equation $L(\phi) = \epsilon^2$. Geometrically, these three situations correspond to the ball and the ellipsoid not intersecting each other, being tangent to each other at a single point, and intersecting each other, respectively. The case where the ball lies exclusively within the ellipsoid can be taken care of initially, as discussed in Section 5. Hence, we are guaranteed that $L(\phi_*) > \epsilon^2$.

Assuming that there is one or two solutions, we can always approach the smallest root by starting the secant method at ϕ_* because the function $L(\phi)$ is convex. If the approach fails to find a root, it is likely that no root exists. These considerations lead to the following *safeguarded* algorithm for computing the smallest root.

1. Choose the starting iterates for the secant method to be ϕ_* and $\phi_* + 0.1\delta$.
2. If $L(\phi_* + 0.1\delta) < \epsilon^2$ then we have a bracket for the smallest root, and we can this bracket to compute the root.
3. Otherwise start the secant method, and stop when the step is small enough.
4. If, during the iterations, we encounter an iterate ϕ such that $L(\phi) < \epsilon^2$ then again we have a bracket for the smallest root.
5. If, during the iterations, we take a step outside the interval of definition, or a step "backwards," or encounter an increasing value of $L(\phi)$, then it is likely that no root exists.
6. In this case we compute the minimum of $L(\phi)$; if this minimum is greater than ϵ^2 then we are guaranteed that no solution exists, otherwise we once again have a bracket for the smallest root.

5 Implementation Issues

In this section, we discuss some issues concerning the implementation of the algorithm MLFIP, described in Sections 3 and 4. First of all, we mention that the special case where x satisfies $\|A\,x - b\|_2 < \epsilon$ and $\|x - d\|_2 = \delta$ can be checked stably and efficiently. The solution to the problem $\min w^T x$ s.t. $\|x - d\|_2 \le \delta$ occurs at the boundary, with $x = d - \delta w$; if this x satisfies $\|A\,x - b\|_2 < \epsilon$ then we are done. Hence, our algorithm initially makes this inexpensive check.

A second issue is the solution of the trust-region subproblems required to evaluate the function $L(\phi)$ in MLFIP. The supbroblems are solved by a large-scale "black-box" routine, namely the algorithm LSTRS [10, 11]. This algorithm only requires multiplications with the matrix $\overline{A} = A\,\overline{V}$ and its transpose, both of which are easy to implement efficiently because the matrix V is merely a single Householder transformation that requires only $\mathcal{O}(n)$ flops. Moreover, LSTRS has fixed storage requirements and is able to handle the singularities of ill-posed problems [12].

When the matrix A – and therefore also $\overline{A}$ – is ill conditioned, LSTRS will usually provide a nearly-optimal solution to the trust-region subproblems. However, our experience indicates that MLFIP requires a boundary solution. Therefore, we "force" LSTRS to compute this kind of solution by tightening the tolerances used in other possible stopping criteria, i.e., interior and nearly-optimal solutions. Note also that the nature of the ill-conditioning of A implies that the matrices used in LSTRS will have smallest eigenvalues very close to zero. Since LSTRS requires the computation of such eigenvalues, and since they are computed with a variant of the Lanczos method [7, 13] which in turns requires matrix-vector multiplications, special care must be taken in this case. In order to accurately compute these eigenvalues, a Tchebyshev Spectral Transformation was used. We refer the reader to [12] for more details.

For the root finder, we use Matlab's `fzero` function whenever a bracket for the desired root is available, and we use Matlab's `fminbnd` function to compute the minimum of $L(\phi)$ in step 6 of the algorithm in Section 4.

Finally, through the use of the safeguarded root finder outlined in the previous section, our implementation is *robust* in the sense that it is able to check for existence of a solution. If the minimum of the function $L(\phi)$ is positive then we are certain that there is no solution to problem (1.1).

6 Numerical Example

We illustrate the use of our algorithm by applying it to a problem in inverse heat conduction from [4]. The sideways heat equation in a quarter plane geometry can be formulated as a Volterra integral equation of the first kind [1],

$$\int_0^t k_\kappa(t-\tau)f(\tau)d\tau = g(t), \qquad 0 \le t \le 1,$$

where the functions f and g represent the temperature as a function of time on an inaccessible and an accessible side of a object, respectively. The kernel function is given by

$$k_\kappa(t) = \frac{1}{2t^{3/2}\sqrt{\pi\kappa}} \exp\left[\frac{1}{4\kappa t}\right].$$

The function k_κ is a Green's function for the heat conduction problem, and the parameter κ is the thermal diffusivity of the object.

The data[1] consist of measurements of the temperature g in a cooling experiment for a particle board. The coefficient κ is equal to 9.55. The integral equation was discretized using the rectangle rule, which yielded a Toeplitz matrix. Thus the matrix A is structured and matrix-vector multiplications can be computed fast using the FFT.

Our computations were done in Matlab 6.5. The problem dimensions are $m = n = 256$, and we first computed the Tikhonov solution x_λ (1.3) using $\lambda = 0.1$; this solution is shown as the solid line in Fig. 6.1. The behavior of the solution near $t = 1$ is unphysical: Since it is a cooling experiment, the solution should decrease monotonically. However, using data from the interval $[0, 1]$ it is impossible to compute any reasonable approximation of the solution for t close to 1, see [2]; therefore this behavior is expected.

The problem of computing confidence intervals for the Tikhonov solution is described in [3] and [8]. The key idea is to choose the vector $w(t_0)$ as samples of an approximate delta function located at some point t_0, and then – ideally – solve the problem (1.1) with $w = \pm w(t_0)$ for $d = 0$, δ equal to the norm of the exact solution, and ϵ equal to the approximate error level in the data. With assumptions on the statistical distribution of the measurement errors it is possible to quantify the level of confidence of the intervals, see [8]. We refrain from this here, as we are merely interested in the computational techniques in this paper.

[1] The data can be obtained from the authors.

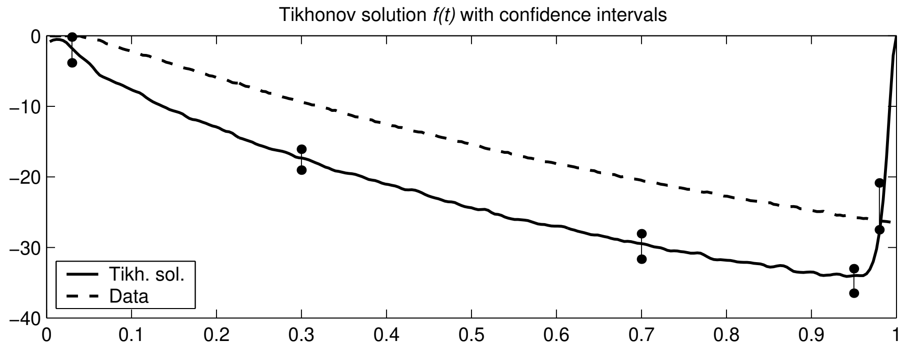

Figure 6.1: The Tikhonov solution x_λ (solid line), the measured data (dashed line) and five confidence intervals computed with $d = 0$, $\delta = 389$, and $\epsilon = 3.2$.

In our experiment, we choose $w(t_0)$ as samples of a Gaussian function with standard deviation equal to 0.5, and since the exact solution was not available we used $\delta = \|x_\lambda\|_2$ (in this case approximately 389). The error level of the data was estimated as follows: A visual examination of the data showed that the errors are below 0.2 (degrees Celsius). As we wanted to allow the residual to be large enough to accommodate errors of this magnitude, we chose $\epsilon = \sqrt{256 \cdot 0.2^2} = 3.2$. We then used the MLFIP algorithm to compute confidence intervals at $t_0 = 0.03$, 0.3, 0.7, 0.95 and 0.98 as shown if Fig. 6.1. We see that for the first four values of t_0, the confidence intervals are quite small, while near $t = 1$ the confidence interval becomes larger reflecting the unphysical nature of the solution here.

We remark that we can replace the constraint $\|x\|_2 \leq \delta$ with a constraint $\|L\,x\|_2 \leq \delta_L$ involving a semi-norm in which L represents the first derivative operator, as done in [4]. This can, e.g., be implemented via a standard-form transformation [6]. The use of the constraint $\|x\|_2 \leq \delta$ here allows us to better illustrate the usefulness of the confidence intervals.

7 Conclusion

We described MLFIP, a large-scale robust algorithm for the minimization of linear functionals defined on solutions of discrete ill-posed problems, based on an algorithm from [3]. MLFIP is robust in the sense that it checks for the existence of a solution. The method requires the solution of a sequence of large-scale trust-region subproblems, which are solved by means of the limited-memory method LSTRS [10]. We illustrated the effectiveness of MLFIP on the problem of computing confidence intervals for the solution of an inverse heat conduction problem with real data. Our experience indicates that MLFIP is a valuable new tool for the numerical treatment of inverse problems.

REFERENCES

1. A. S. Carasso, *Determining surface temperatures from interior observations*, SIAM J. Appl. Math. 42 (1982), pp. 558–574.

2. L. Eldén, *The numerical solution of a non-characteristic Cauchy problem for a parabolic equation;* in P. Deuflhard and E. Hairer (Eds.), *Numerical Treatment of Inverse Problems in Differential and Integral Equations, Proceedings of an International Workshop, Heidelberg, 1982*, pp. 246–268, Birkhäuser, Boston, 1983.
3. L. Eldén, *Algorithms for the computation of functionals defined on the solution of discrete ill-posed problems*, BIT, 30 (1990), pp. 466–483.
4. L. Eldén, F. Berntsson and T. Reginska, *Wavelet and Fourier methods for solving the sideways heat equation*, SIAM J. Sci. Comput., 21 (2001), pp. 2187–2205.
5. P. C. Hansen, *Regularization Tools: A Matlab package for analysis and solution of discrete ill-posed problems*, Numer. Algo., 6 (1994), pp. 1–35.
6. P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion,* SIAM, Philadelphia, 1998.
7. R. B. Lehoucq, D. C. Sorensen and C. Yang, *ARPACK User's Guide: Solution of Large Scale Eigenvalue Problems by Implicitly Restarted Arnoldi Methods*, SIAM, Philadelphia, 1998.
8. D. P. O'Leary and B. W. Rust, *Confidence intervals for inequality-constrained least squares problems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 473–489.
9. P. Paateri, *Extreme value estimation, a method for regularizing ill-posed inversion problems;* in A. N. Tikhonov (Ed.), *Ill-Posed Problems in Natural Sciences*, VSP, Utrecht, 1992, pp. 118–133.
10. M. Rojas, S. A. Santos and D. C. Sorensen, *A new matrix-free algorithm for the large-scale trust-region subproblem*, SIAM J. Optim., 11 (2000), pp. 611–646.
11. M. Rojas, S. A. Santos and D. C. Sorensen, *LSTRS: Matlab software for large-scale trust-region subproblems and regularization*, Technical Report 2003-4, Department of Mathematics, Wake Forest University, October 2003; submitted to ACM Trans. Math. Soft.
12. M. Rojas and D. C. Sorensen, *A trust-region approach to the regularization of large-scale discrete forms of ill-posed problems*, SIAM J. Sci. Comp., 26 (2002), pp. 1843–1861.
13. D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.