INSTITUTE OF PHYSICS PUBLISHING INVERSE PROBLEMS

Inverse Problems **18** (2002) 1291–1307 PII: S0266-5611(02)32278-0

An interior-point trust-region-based method for large-scale non-negative regularization

Marielba Rojas[1,3] **and Trond Steihaug**[2]

[1] CERFACS, 42 Avenue Gaspard Coriolis, 31057 Toulouse Cedex 1, France
[2] Department of Informatics, University of Bergen, Høyteknologisenteret, N-5020 Bergen, Norway

E-mail: mrojas@cerfacs.fr, mrojas@mthcsc.wfu.edu and trond@ii.uib.no

Received 4 January 2002, in final form 10 June 2002
Published 1 August 2002
Online at stacks.iop.org/IP/18/1291

Abstract

We present a new method for large-scale non-negative regularization based on a quadratically and non-negatively constrained quadratic problem. Such problems arise for example in the regularization of ill posed problems in image restoration where the matrices involved are very ill conditioned. The method is an interior-point iteration that requires the solution of a large-scale and possibly ill conditioned parametrized trust-region subproblem at each step. The method uses recently developed techniques for the large-scale trust-region subproblem. We describe the method and present preliminary numerical results on test problems and image restoration problems.

1. Introduction

We consider the problem

$$\begin{aligned} &\min \tfrac{1}{2}\|Ax - b\|^2 \\ &\text{s.t.}\ \ \|x\| \leqslant \Delta \\ &\qquad x \geqslant 0 \end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{m\times n}$, $m \geqslant n$, $b \in \mathbb{R}^m$ and $\Delta > 0$. Throughout the paper, $\|\cdot\|$ denotes the ℓ_2-norm. We assume that m and n are large, and that the matrix A might not be explicitly available, but that we know how to compute the action of A and A^T on vectors of appropriate dimensions.

Problem (1) is an important problem that arises for example in the regularization of ill posed problems from image restoration (cf [1]), where we want to recover an image from blurred and noisy data. In these problems, the matrix A is a discretized version of a blurring operator, and b is a vector representation of a degraded image. In image restoration problems, the norm constraint is a so-called regularization term that controls the growth in the size of

[3] Present address: Department of Mathematics, Wake Forest University, PO Box 7388, Winston-Salem, NC 27109, USA.

the least-squares solution observed in most ill posed problems with noisy data, and the non-negativity constraints reflect the fact that each component of the vector x represents either the colour value, or the light-intensity value, of a pixel in the digital representation of the image, and therefore must be non-negative.

Most techniques for image restoration do not take into account the non-negativity constraints. Instead, they solve the regularization problem, for example, the least-squares problem with the norm constraint only, and set to zero the negative components in the solution. This strategy clearly introduces some errors, but produces satisfactory results in certain cases, such as when the images are normal photographs. However, this is not the case in other applications. As pointed out in [14, 22], in astronomical imaging most of the pixel values in the desired solution are actually zero or nearly zero, and therefore setting negative values to zero might introduce considerable errors, and might yield a restored image with missing details or with artifacts.

Some of the methods for regularization with non-negativity constraints are [7, 9, 14, 22]. There are also methods that follow the approach of projections on convex sets or POCS methods such as [24, 33]. Finally, it is worth mentioning the collection of software in [31], which includes routines for least-squares problems with non-negativity constraints.

The methods in [7] are based on truncated singular-value decomposition (TSVD) regularization [16]. The approach can be used on large-scale problems by computing only a few singular values, although in general it is difficult to determine how many values should be computed. The authors propose methods based on a linearly constrained quadratic problem and its dual. The primal approach, at least in the current version, can be used only on small problems. The dual approach is suitable for large-scale problems, although the method might require additional regularization and it might fail to converge in certain cases. The authors report results for small problems only. The solutions computed by these methods have very attractive theoretical properties. The work in [9] proposes an active-set quadratic-programming method for a problem closely related to (1). This method has been successfully applied to problems of moderate size ($n \approx 2000$), but the current version is prohibitively expensive for larger problems, such as those in image restoration, where typically $n = 65\,536$. The methods in [14] are based on a quasi-Newton approach, and appear as promising strategies. The authors point out that good preconditioners are needed to improve efficiency, and more experiments are also needed to assess the effectiveness of the methods. The methods in [22] are iterative methods for linear systems that impose a non-negativity constraint at each step. The methods achieve regularization by early termination of the iteration, which is based on the heuristic that initial iterates will not be contaminated by the noise in the data, and which requires a procedure for determining when to stop the iteration. In practice, most regularization approaches based on iterative methods rely on visual inspection for stopping the iteration. Preconditioners are used in [22] to obtain computationally competitive methods.

In this work, we present a new method for large-scale non-negative regularization. Our method is not based on a heuristic and does not depend on the availability of a preconditioner. The method is matrix free in the sense that only matrix–vector products with A and A^T are required.

The organization of the paper is the following. In section 2, we present the properties of problem (1) and we show the relationship of the problem to the Tikhonov regularization approach with non-negativity constraints. We derive our method for problem (1) in section 3. In section 4, we present some preliminary numerical results on ill posed problems from inverse problems, including image restoration problems, and discuss some of the properties of the method. In section 5, we present some extensions to the main problem that can also be solved by the method in section 3. Concluding remarks are presented in section 6.

We shall use the following notation throughout the paper. We denote matrices by capital letters, vectors by lower-case letters and vector components by the corresponding Greek letter with a subscript. Given a vector $x = (\xi_1, \xi_2, \dots, \xi_n)^T$, we write $\operatorname{diag}(x)$ or $\operatorname{diag}(\xi_1, \xi_2, \dots, \xi_n)$ to denote a diagonal matrix with element ξ_i in the (i, i) position. Finally, we denote the vector of all ones by $e = (1, \dots, 1)^T$.

2. The problem

We observe that problem (1) always has a solution, which is unique when A has full rank. Next, we derive optimality conditions satisfied by the solutions of problem (1). Let $\lambda \in \mathbb{R}$ and $y \in \mathbb{R}^n$, then the Lagrangian functional associated with the problem is

$$\mathcal{L}(x, \lambda, y) = \frac{1}{2}x^T A^T A x - (A^T b)^T x + \frac{1}{2}b^T b - \frac{\lambda}{2}(\|x\|^2 - \Delta^2) + y^T x,$$

and the Karush–Kuhn–Tucker (KKT) first-order necessary conditions for a feasible point x and Lagrange multipliers λ and y to be a solution of problem (1) are

$$\begin{aligned} &\text{(i)} && (A^T A - \lambda I)x = A^T b - y \\ &\text{(ii)} && \lambda(\|x\|^2 - \Delta^2) = 0 \\ &\text{(iii)} && y^T x = 0 \\ &\text{(iv)} && \lambda \leqslant 0, \qquad y \leqslant 0. \end{aligned} \tag{2}$$

The vector of Lagrange multipliers (or dual variables) y and the *duality gap* $y^T x$ will play a key role in the algorithm in section 3.

Observe that, since (1) is a convex quadratic problem, conditions (2) are both necessary and sufficient. This fact helps us prove an equivalence between problem (1) and another constrained least-squares problem. Note that, to obtain non-negative regularized solutions, we could solve the following problem:

$$\begin{aligned} &\min \tfrac{1}{2}\|Ax - b\|^2 + \varepsilon^2\|x\|^2, \\ &\text{s.t.}\, x \geqslant 0 \end{aligned} \tag{3}$$

where $\varepsilon^2 > 0$. This is the classical Tikhonov regularization problem with additional non-negativity constraints. The problem is strictly convex and its solution is, therefore, unique. Necessary and sufficient conditions for a vector x and Lagrange multipliers $y \in \mathbb{R}^n$ to be a solution of problem (3) are

$$\begin{aligned} &\text{(i)} && (A^T A + \varepsilon^2 I)x = A^T b - y \\ &\text{(iii)} && y^T x = 0 \\ &\text{(iv)} && y \leqslant 0. \end{aligned} \tag{4}$$

Our next result, lemma 2.1, establishes a relationship between problems (1) and (3).

Lemma 2.1. *If x_λ is a solution of problem (1) with Lagrange multiplier $\lambda < 0$, then x_λ solves problem (3). Conversely, a solution x_ε of (3) solves problem (1) for $\Delta = \|x_\varepsilon\|$.*

Proof. The proof follows directly by carrying out the appropriate substitutions in the KKT conditions (2) and (4). The first part follows by setting $\varepsilon^2 = -\lambda$ in (4). The second part follows from the fact that x_ε solves (3) and $\Delta = \|x_\varepsilon\|$. □

In general, the case $\lambda = 0$ corresponds to the situation where the norm constraint is not active and so no regularization is accomplished. In this case we should reduce Δ. The case in which $\lambda = 0$ and the norm constraint is active can only arise when the matrix A is rank deficient and problem (5) may have an infinite number of solutions. Currently, there is no way of distinguishing between the two cases and so we also reduce Δ in the latter case.

Both problems (1) and (3) can be used to obtain non-negative regularized solutions. Problem (3) can be solved, for example, by the methods proposed in [6, 20]. The methods will work well as long as the problem is well posed, which in turn depends on having a good estimate of the Tikhonov regularization parameter ε. This parameter does not necessarily have a physical meaning in applications, and determining its optimal value is a difficult problem in itself. Most of the methods currently available require the solution of several problems of type (3) for different values of ε. This approach might be prohibitive in the large-scale setting. Recent and promising methods for computing the Tikhonov regularization parameter for large-scale problems have been proposed in [3, 4] and [17]. Note, however, that these techniques are not designed for the problem with non-negativity constraints.

The approach (1) is better suited for those applications in which the parameter Δ has a physical meaning. One such application is image restoration, where Δ is an estimate of the energy norm of the target image, or where we want the energy of the restored image to match a certain value (cf [1]). Applications in geophysics where Δ is known *a priori* are discussed in [13] and [27]. The method we present in this paper is intended for large-scale problems of type (1). The method can efficiently handle ill posed problems of type (1), including those instances for which a good estimate of Δ is not available.

3. The method

Before deriving the method, and for clarity of presentation, we define $H = A^T A$ and $g = -A^T b$, and formulate the following problem equivalent to (1):

$$\begin{aligned} &\min \tfrac{1}{2}x^T Hx + g^T x \\ &\text{s.t. } \|x\| \leqslant \Delta \\ &\quad x \geqslant 0. \end{aligned} \tag{5}$$

In order to develop our method for problem (5), we first eliminate the non-negativity constraints by restricting our attention to $x > 0$ and introducing a modified objective function. Several choices are possible for this function; for example, an entropy function is used in [22]. Although such choices can also be included in our approach, here we shall derive the method for the *logarithmic barrier function*, defined as

$$f_\mu(x) = \tfrac{1}{2}x^T Hx + g^T x - \mu \sum_{i=1}^{n} \log \xi_i,$$

where $x = (\xi_1, \xi_2, \dots, \xi_n)^T$ and $\mu > 0$ is the so-called *barrier* or *penalty parameter*. The use of the modified function yields a family of problems depending on μ, where each problem is given by

$$\begin{aligned} &\min\ f_\mu(x) \\ &\text{s.t. } \|x\| \leqslant \Delta. \end{aligned} \tag{6}$$

The Lagrangian functional associated with this problem is

$$\mathcal{L}(x, \lambda) = f_\mu(x) - \frac{\lambda}{2}(\|x\|^2 - \Delta^2),$$

where $\lambda \in \mathbb{R}$. Let $X = \mathrm{diag}(x)$, then the following are the KKT necessary conditions for a feasible point x and Lagrange multiplier λ to be a solution of problem (6):

$$\begin{aligned} &\text{(i)} && (H + \mu X^{-2} - \lambda I)x = -g \\ &\text{(ii)} && \lambda(\|x\|^2 - \Delta^2) = 0 \\ &\text{(iii)} && \lambda \leqslant 0. \end{aligned} \tag{7}$$

The idea of the method is then to solve a sequence of problems of type (6), while decreasing the parameter μ towards zero. Notice that by using problem (6) we have restricted the solution to have positive components only. This follows an interior-point approach (cf [5, 10, 32]), in which the iterates are feasible and positive.

We shall now introduce a further simplification by substituting the non-linear barrier problems (6) by quadratically constrained quadratic problems, or trust-region subproblems, where the objective function will be a quadratic approximation to the logarithmic barrier function, and where the trust-region radius Δ will remain fixed. The subproblems are constructed as follows.

Consider the second-order Taylor expansion of f_μ around a point x,

$$q_\mu(x + h) = f_\mu(x) + \nabla f_\mu(x)^T h + \tfrac{1}{2} h^T \nabla^2 f_\mu(x) h,$$

where $\nabla^2 f_\mu(x) = H + \mu X^{-2}$ and $\nabla f_\mu(x) = Hx + g - \mu X^{-1}e$, with $X = \mathrm{diag}(x)$. Let us now formulate the trust-region subproblem

$$\begin{aligned} &\min q_\mu(x + h) \\ &h \\ &\text{s.t. } \|x + h\| \leqslant \Delta, \end{aligned} \tag{8}$$

and notice that setting $z = x + h$ we obtain a new trust-region subproblem equivalent to (8), and given by

$$\begin{aligned} &\min \tfrac{1}{2} z^T (H + \mu X^{-2}) z + (g - 2\mu X^{-1}e)^T z \\ &z \\ &\text{s.t. } \|z\| \leqslant \Delta. \end{aligned} \tag{9}$$

As established in [11] and [29], necessary and sufficient conditions for a feasible point z and Lagrange multiplier $\lambda \in \mathbb{R}$ to be a solution of problem (9) are

$$\begin{aligned} &\text{(i)} && (H + \mu X^{-2} - \lambda I)z = 2\mu X^{-1}e - g \\ &\text{(ii)} && H + \mu X^{-2} - \lambda I \text{ positive semidefinite} \\ &\text{(iii)} && \lambda(\|z\|^2 - \Delta^2) = 0 \\ &\text{(iv)} && \lambda \leqslant 0. \end{aligned} \tag{10}$$

Note that (ii) always holds in the convex case.

Our method consists of solving a sequence of problems of type (9) for z, for different values of μ and x, while driving the barrier parameter μ towards zero, and preserving positive iterates. The latter is accomplished by means of a linesearch to be described in section 3.3. Denoting problem (9) by $P(x, \mu)$, we can now write our method as algorithm 3.1 in figure 1.

The tolerances $\varepsilon_y, \varepsilon_f, \varepsilon_x$ are used in the stopping criteria, while σ is used in step 2.4 for updating μ. Next, we describe each component of the algorithm in detail, namely, the update of the barrier parameter, the choice of initial values, the linesearch and the stopping criteria.

3.1. Update of the barrier parameter μ

We can derive a formula for updating the barrier parameter by first computing an approximation to the dual variables y in (2) in the following way. Recall from (2) (i), that y satisfies

$$y = -(H - \lambda I)x - g,$$

```
Algorithm 3.1 $TRUST_\mu$
Input: $H \in \mathbb{R}^{n\times n}$, symmetric, or procedure for computing $H$ times a vector;
$g \in \mathbb{R}^n$; $\Delta > 0$; $\varepsilon_y, \varepsilon_f, \varepsilon_x, \sigma \in (0,1)$.
Output: $x_* > 0$ satisfying (10) for $\mu$ close to zero.
1. Choose $x_0 > 0$, $\mu_0 > 0$, set $k = 0$
2. while not convergence do
     2.1 Solve $P(x_k, \mu_k)$ for $z_k$
     2.2 Set $h_k = z_k - x_k$
     2.3 Compute $\beta_k$ such that $x_k + \beta_k h_k > 0$
     2.3 Set $x_{k+1} = x_k + \beta_k h_k$
     2.4 Compute $\mu_{k+1}$ such that $\{\mu_k\} \to 0$
     2.5 Set $k = k + 1$
   end while
```

Figure 1. Method for trust-region subproblems with non-negativity constraints.

and notice that a solution z, λ of the trust-region subproblem (9) will satisfy (10) (i), which we can rewrite as

$$-(H - \lambda I)z - g = \mu X^{-2}z - 2\mu X^{-1}e. \tag{11}$$

We now define

$$\tilde{y} = -(H - \lambda I)z - g,$$

and use $\tilde{y}$ as an approximation to y. Observe that we can compute $\tilde{y}$ from (11) as

$$\tilde{y} = \mu(X^{-2}z - 2X^{-1}e), \tag{12}$$

and when $x = z$ we have the approximation to the duality gap in (2) (iii)

$$\tilde{y}^T x = -\mu n, \tag{13}$$

which leads to the following formula for μ:

$$\mu = \frac{1}{n}\tilde{y}^T x.$$

In practice, the update for μ will be

$$\mu_{k+1} = \frac{\sigma}{n}|\tilde{y}^T x|, \tag{14}$$

with $\sigma \in (0, 1)$ and $x = x_k$ or $x = x_{k+1}$, and with $\tilde{y}$ as in (12) for $\mu = \mu_k$, $x = x_k$ or $x = x_{k+1}$ and $z = z_k$.

3.2. *Choice of initial values* x_0, μ_0

To compute initial values for x and μ, we first solve the trust-region subproblem *without* the non-negativity constraints, i.e.

$$\begin{aligned} &\min \tfrac{1}{2}\|Ax - b\|, \\ &\text{s.t. } \|x\| \leqslant \Delta \end{aligned} \tag{15}$$

and we denote the solution to this problem and the corresponding Lagrange multiplier by x_{TRS} and λ_{TRS}, respectively. We use x_{TRS} and λ_{TRS} to compute an initial value for μ in the following way.

We first compute an approximate initial value for the dual variables y as

$$\tilde{y}_0 = -g - (H - \lambda_{\mathrm{TRS}} I)x_{\mathrm{TRS}},$$

and then compute μ_0 as

$$\mu_0 = \frac{\sigma}{n}|\tilde{y}_0^T x_{\mathrm{TRS}}|.$$

We then choose x_0 as either $x_0 = |x_{\mathrm{TRS}}|$ with zero components replaced by a small positive value, or $x_0 = x_{\mathrm{TRS}}$ with negative and zero components replaced by a small positive value, so $x_0 > 0$. We use x_0 to test for convergence as described in section 3.4.

3.3. Linesearch

A linesearch is necessary to ensure that the iterates x_k remain positive, since there is no guarantee that z_k computed in step 2.1 will have only positive components. The $(k+1)$th iterate is computed as $x_{k+1} = x_k + \beta_k h_k$, where $h_k = z_k - x_k$ and

$$\beta_k < \min_{i \text{ s.t. } 1 \leqslant i \leqslant n \text{ and } \zeta_i \leqslant 0} \frac{\xi_i}{|\eta_i|},$$

where $x_k = (\xi_1, \xi_2, \dots, \xi_n)^T$, $z_k = (\zeta_1, \zeta_2, \dots, \zeta_n)^T$ and $h_k = (\eta_1, \eta_2, \dots, \eta_n)^T$.

In practice, we use the following safeguarded formula to update the iterates:

$$x_{k+1} = x_k + \min\{1, 0.9995\beta_k\} h_k.$$

3.4. Stopping criteria

The stopping criteria rely on the change in value of the objective function, the proximity of the iterates and the size of the duality gap. For the latter, we compute $\tilde{y}_k$ according to (12), with $\mu = \mu_{k-1}$ or $\mu = \mu_k$, $x = x_{k-1}$ or $x = x_k$, and $z = z_{k-1}$ computed in step 2.1 of algorithm 3.1.

Let $f(x) = \frac{1}{2}x^T H x + g^T x$, and $\varepsilon_f, \varepsilon_x, \varepsilon_y \in (0, 1)$, then for $k \geqslant 1$, algorithm 3.1 proceeds until

$$|f(x_k) - f(x_{k-1})| \leqslant \varepsilon_f |f(x_k)| \qquad \text{or} \qquad \|x_k - x_{k-1}\| \leqslant \varepsilon_x \|x_k\|$$
$$\text{or} \qquad |\tilde{y}_k^T x_k| \leqslant \varepsilon_y \|x_k\|.$$

For $k = 0$ we only check the last condition for the initial values of x and $\tilde{y}$, i.e. $|\tilde{y}_0^T x_0| \leqslant \varepsilon_y \|x_0\|$, with $\tilde{y}_0$ and x_0 as in section 3.2.

4. Numerical results

In this section, we present numerical results that illustrate some observed properties of the method TRUST$_\mu$ and the performance of the method on image restoration problems. The results were obtained with a `MATLAB 5.3` implementation of algorithm 3.1 applied to constrained least-squares problems of type (1). We ran our experiments on a SunBlade 1000 with a 750 MHz processor and 1024 Megabytes of RAM running Solaris 5.8. The floating point arithmetic was IEEE standard double precision with machine precision $2^{-52} \approx 2.2204 \times 10^{-16}$.

This section is organized as follows. In section 4.1 we discuss the solution of the large-scale trust-region subproblems in step 2.1 of algorithm 3.1. In sections 4.2 and 4.3 we illustrate two aspects of the method. The first is the convergence behaviour with respect to the update of the barrier parameter μ. The second is the accuracy of the approximate solutions. For this purpose, we use test problems from the Regularization Tools package [15]. All the problems in this package are discretized versions of Fredholm integral equations of the first kind. In section 4.4 we report results on astronomical imaging problems from the RESTORE Tools package [18].

The following holds throughout this section:

- x_{TRS} denotes the solution of the trust-region subproblem (15),
- $x_{\text{TRS-P}} = \max\{0, x_{\text{TRS}}\}$,
- x_μ denotes the solution of the non-negatively constrained trust-region problem (1) computed with TRUST$_\mu$,
- x_{IP} denotes the discretized version of the exact solution of the inverse problem which was available for all tests,
- $\Delta = \|x_{\text{IP}}\|$ in all the experiments,
- formulae (12) and (14) were computed using the previous iterate and
- the ℓ_2-norm relative error in x with respect to y is computed as usual as $\frac{\|x-y\|}{\|y\|}$.

4.1. Solution of the trust-region subproblems

The trust-region subproblems in step 2.1 of algorithm 3.1 were solved with a `MATLAB 5.3` implementation of the method LSTRS from [26]. The method is based on matrix–vector products with A and A^T, and can handle the singularities associated with ill posed problems. The method has been successfully used to solve regularization problems through the trust-region approach (15) in seismic inversion and related problems [25, 27]. LSTRS was also used to compute an initial iterate for TRUST$_\mu$, as discussed in section 3.2.

LSTRS is an iterative method that requires the solution of a large-scale eigenvalue problem at each step. Unless otherwise indicated, the eigenvalue problems were solved by means of the implicitly restarted Lanczos method (IRLM) [30] as implemented in `ARPACK` [19]. The IRLM is particularly suitable for large-scale problems since it has low and fixed storage requirements and relies upon matrix–vector products only. In the current implementation of LSTRS, a Mexfile interface was used to access `ARPACK`. Note that all the capabilities of `ARPACK` are now available through the routine `eigs` in `MATLAB 6`. In all the experiments, the parameters for the IRLM were 11 Lanczos basis vectors with nine shifts (nine matrix–vector products) on each implicit restart, with a maximum of 13 implicit restarts allowed.

The eigenvalue problems in LSTRS have the form

$$\begin{pmatrix} \alpha & g^T \\ g & H \end{pmatrix} y = \lambda y, \tag{16}$$

where α is a scalar parameter updated at each iteration, and where we are interested in the smallest eigenvalue. The goal of LSTRS is to find an optimal value α_* for the parameter α. A solution to the trust-region subproblem can then be recovered from the solution of (16) for $\alpha = \alpha_*$. We refer the reader to [26] for more details.

One strategy that we have implemented in most of our experiments is to use the optimal value of α computed by LSTRS for the subproblem in the current iteration of TRUST$_\mu$ as the initial α for solving the subproblem in the next iteration. Intuitively, this would reduce the number of LSTRS iterations since we do not expect x and μ to change significantly as TRUST$_\mu$ converges, and therefore the trust-region subproblems in step 2.1 of algorithm 3.1 should not differ significantly. In practice, we have observed that using this strategy indeed reduces the number of LSTRS iterations and consequently the number of matrix–vector products required.

μ_k	$\frac{\Vert x_k - x_{\mathrm{TRS}}\Vert}{\Vert x_{\mathrm{TRS}}\Vert}$
1.0000e+00	2.8622e-01
1.0000e-01	4.0446e-01
1.0000e-02	4.6372e-01
1.0000e-03	2.0026e-01
1.0000e-04	1.4872e-01
1.0000e-05	8.1839e-02
1.0000e-06	3.0494e-02
1.0000e-07	7.7923e-03
1.0000e-08	1.3599e-03
1.0000e-09	1.6415e-04
1.0000c-10	1.6770c-05
1.0000e-11	1.6798e-06
1.0000e-12	1.6800e-07

(a)

μ_k	$\frac{\Vert x_k - x_{\mathrm{TRS}}\Vert}{\Vert x_{\mathrm{TRS}}\Vert}$
1.0000e-01	2.8723e-01
1.0000c-02	4.0645c-01
1.0000e-04	1.4009e-01
1.0000e-08	5.4001e-04
1.0000c-16	1.2630c-10
1.0000e-32	0

(b)

Figure 2. Convergence rate of TRUST$_\mu$ iterates for (a) linear and (b) quadratic updates of μ.

The tolerances for LSTRS were chosen as follows.

Purpose	Value
Relative accuracy in the norm of trust-region solution, ε_Δ	$*$
Interior solution, $\varepsilon_{\mathrm{Int}}$	0
Nearly optimal solution in the so-called *hard case*, $\varepsilon_{\mathrm{HC}}$	$*$
Update of the parameter α, ε_α	10^{-8}
Small eigenvector component, ε_ν	10^{-2}

We shall indicate the values of ε_Δ and $\varepsilon_{\mathrm{HC}}$ when we describe each particular experiment.

4.2. Convergence rate with respect to μ

In these experiments, we have chosen problem **foxgood** from [15], and set the dimensions to $m = n = 300$. As described in section 4.1, the method LSTRS was used to compute x_{TRS}, a solution to (15). LSTRS computed a positive solution for problem **foxgood**, thus we could use this solution to test the convergence of the TRUST$_\mu$ iterates.

The eigenvalue problems in LSTRS were solved with the Matlab routine `eig`, and the initial value of the parameter α was set to zero in each call to LSTRS. The tolerances for LSTRS were $\varepsilon_\Delta = 10^{-3}$ and $\varepsilon_{\mathrm{HC}} = 10^{-10}$. For this experiment, we did not used the stopping criteria described in section 3.4; instead, we let the method TRUST$_\mu$ run for a large number of iterations to be able to observe the convergence behaviour.

The convergence behaviour of TRUST$_\mu$ for linear and quadratic updates of μ is shown in figure 2, where x_k denotes the TRUST$_\mu$ iterates. These results seem to indicate that the method TRUST$_\mu$ has the very desirable property of the convergence rate of the sequence of iterates being determined by the update of the barrier parameter.

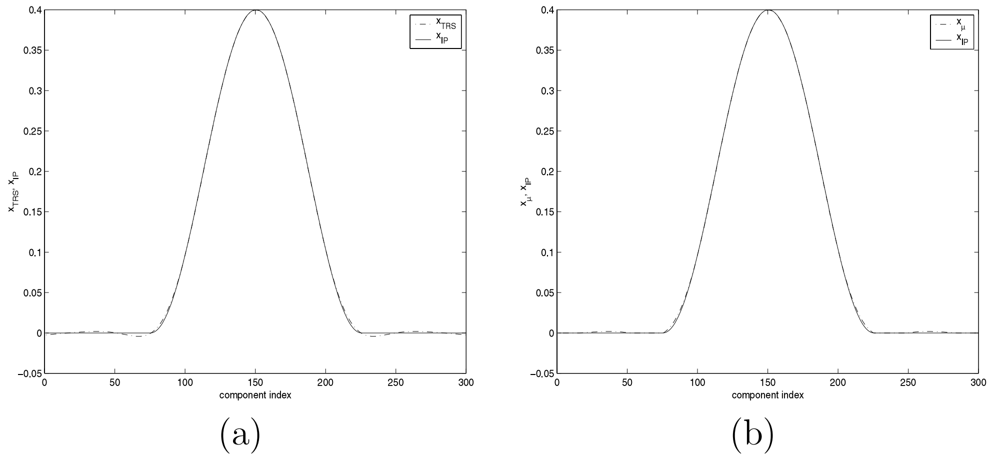


Figure 3. Problem **phillips**, $n = 300$. (a) LSTRS and (b) TRUST$_\mu$ solutions. Relative error in LSTRS solution: 1.0065×10^{-2}. Relative error in TRUST$_\mu$ solution: 6.9218×10^{-3}.

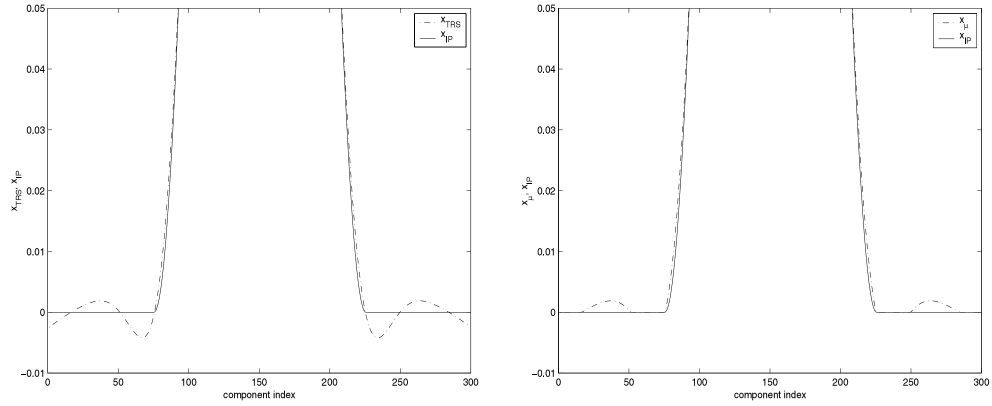
4.3. Accuracy of the regularized solution

In this section, we compare the accuracy of regularized non-negative solutions computed with the method TRUST$_\mu$ with respect to regularized solutions that do not take into account the non-negativity constraints. For this purpose, we chose problem **phillips** from [15], with dimensions $m = n = 300$. We again computed x_{TRS}, a solution to the trust-region subproblem (15), by means of LSTRS, and we used it to compare the two regularization approaches: with and without the non-negativity constraints. The eigenvalue problems were solved with ARPACK. The tolerances for LSTRS were $\varepsilon_\Delta = 10^{-3}$ and $\varepsilon_{\mathrm{HC}} = 10^{-10}$. The tolerances for TRUST$_\mu$ were $\varepsilon_y = 10^{-12}$ and $\varepsilon_f = \varepsilon_x = 10^{-5}$. The update of μ according to (14) was computed with $\sigma = 0.01$.

LSTRS required four iterations and 525 matrix–vector products to compute x_{TRS}. This solution has a relative error of the order of 10^{-2} with respect to x_{IP}. Both x_{IP} and x_{TRS} are shown in figure 3(a) (solid and dashed–dotted curves, respectively). As we can observe in figure 3(a), and more clearly in a closer look in figure 4(a), x_{TRS} has some negative components. Figure 4(a) also illustrates the well known Gibbs phenomenon, which is usually seen with band-pass filters (cf [23]).

We then applied the method TRUST$_\mu$ to solve problem (1) for the same test problem. We used the same LSTRS parameters as before and the initial value for α described in section 4.1. We set $x_0 = |x_{\mathrm{TRS}}|$ with zero components replaced by 10^{-5}. We obtained the solution x_μ shown in figure 3(b) (dashed–dotted curve), which has a relative error of the order of 10^{-3} with respect to x_{IP}, and has only positive components. Figure 4(b) shows that the components in x_μ remain positive, and that the ripples are considerably smaller compared with those in figure 4(a). Finally, figure 5 shows the difference between x_μ and $x_{\mathrm{TRS\text{-}P}}$. We observe from this figure that, once zero components are replaced by a small constant, $x_{\mathrm{TRS\text{-}P}}$ might provide a better starting point for TRUST$_\mu$ than $|x_{\mathrm{TRS}}|$.

TRUST$_\mu$ required one iteration of the main loop in algorithm 3.1 (i.e. two calls to LSTRS), five LSTRS iterations and 631 matrix–vector products. The results are summarized in table 1. The value of μ was of the order of 10^{-16}. As we can observe, the cost is only slightly higher than that needed to compute the first LSTRS solution, and the result is considerably



(a) (b)

Figure 4. Problem **phillips**, $n = 300$. Close-up of (a) TRS and (b) TRUST$_\mu$ solutions.

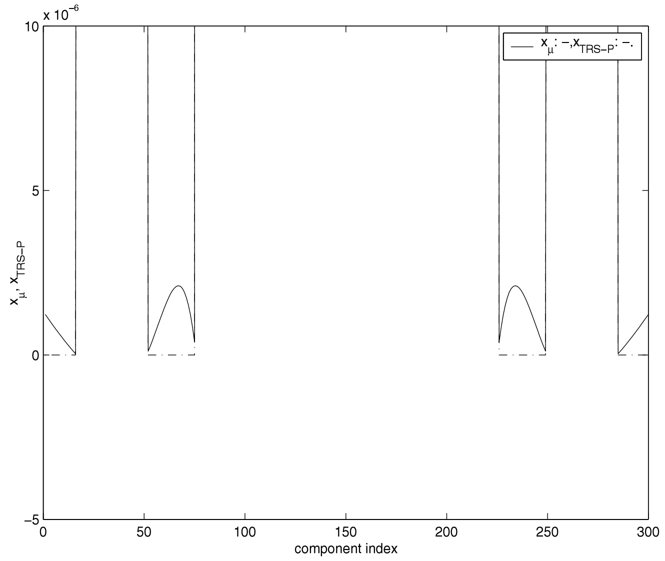


Figure 5. Problem **phillips**, $n = 300$. Comparison between $x_{\text{TRS-P}}$ and x_μ. Relative error in $x_{\text{TRS-P}}$ with respect to x_μ: 3.6638×10^{-6}.

more accurate. These results seem to indicate that, although solving the problem with non-negativity constraints involves an additional computational cost, such a cost is not too high and might be justified by the possibility of obtaining a more accurate regularized solution.

4.4. Image restoration problems

In this section, we present results on image restoration problems from [18]. In imaging applications such as image restoration, image formation is modelled by an integral equation of the first kind

$$\gamma(s) = \int K(s,t)\phi(t)\,\mathrm{d}t, \tag{17}$$

Table 1. Accuracy of regularized solution.

Method	Matrix–vector products	LSTRS iterations	Relative error in approximate solution
LSTRS	525	4	1.0065×10^{-2}
TRUST$_\mu$	631	5	6.9218×10^{-3}
TRUST$_\mu$ iterations: 2 (initialization and one iteration of the main loop)			

where the kernel $K(s,t)$ is called the *point spread function* (PSF), that determines the image of a single point source under the imaging system, $\phi(t)$ is the true image and $\gamma(s)$ is called a blurred version of $\phi(t)$. Imaging systems are called *space invariant* when the PSF acts uniformly across source and image spaces, i.e. a translation in the source space corresponds to a proportional translation on the image space. Most imaging systems can be modelled by a space invariant PSF. If the system is not space invariant it is called *space variant.* Such systems arise for example when recording images of objects that move with different velocities with respect to the recording device.

The discretization of equation (17) yields a linear system of equations

$$Ax = b, \tag{18}$$

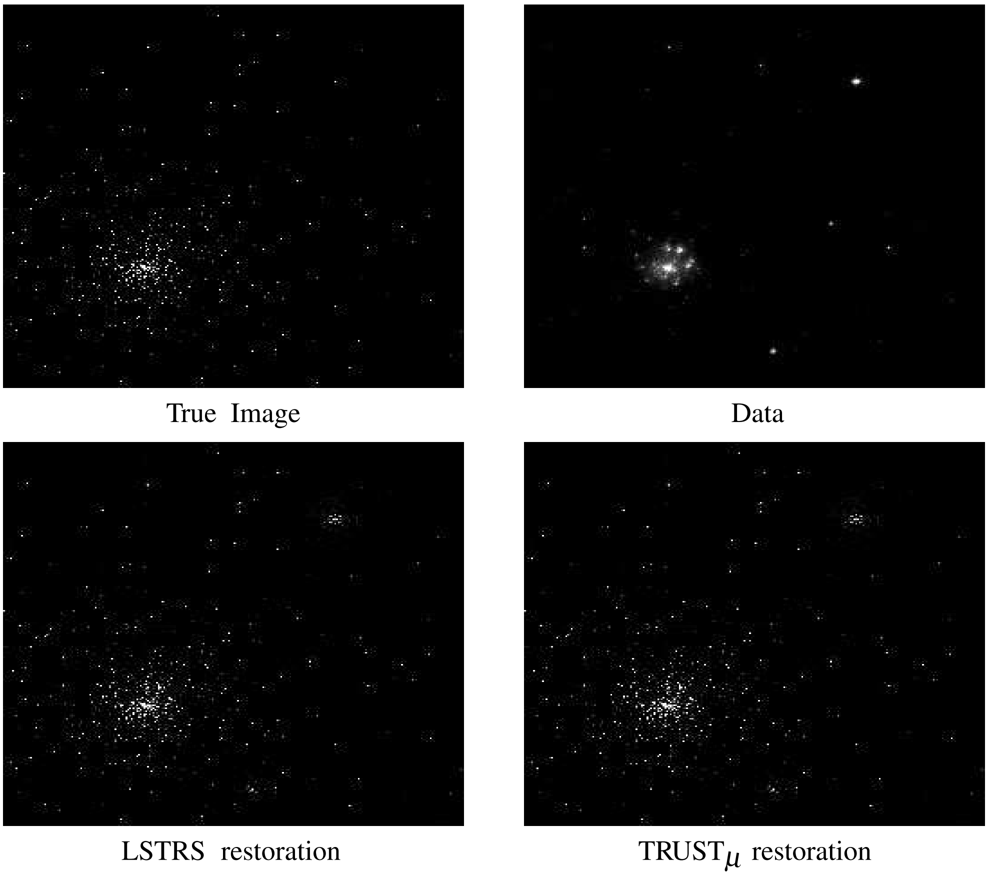
where $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$. The matrix A is a discretized version of the *blurring* operator constructed from $K(s,t)$. The vector $b = \bar{b} + \eta$ is a vector representation of a version of the true image degraded by blur and noise, with $\bar{b}$ being a discretized version of the blurred image $\gamma(s)$, and η representing noise. Problem (17) is ill posed, and the discrete problem (18) usually inherits this feature in the sense that the matrix A is highly ill conditioned, with a singular spectrum that decays to zero gradually, and high-frequency components of the singular vectors corresponding to small singular values. In most problems of interest, n is of the order of tenths of thousandths. Note also that straightforward discretization leads to matrices A that are block-Toeplitz with Toeplitz blocks. Thus, matrix–vector products involving A and A^T can be efficiently computed by means of the FFT at a cost of $\mathcal{O}(n \log n)$. This is the way the matrix–vector products are implemented in [18].

4.4.1. Star cluster. In this section, we consider the problem of restoring the image of a star cluster, consisting of simulated data used by astronomers to test image restoration methods for the Hubble space telescope (HST). Such methods are needed to restore images recorded with the HST before the mirrors in the camera were corrected. See [21] and the references therein for more details.

The problem is space variant and this should be taken into account when modelling the imaging system as suggested in [21]. Therefore, we used a combination of four PSFs described in [21] where the source space is decomposed in four subdomains with a different PSF acting on each of them. The matrix A represents the blurring operator, constructed from the four PSFs. The discrete problem is of dimension $n = 65\,536$. The true image, the image recorded by the HST or data image and the restorations obtained with LSTRS and TRUST$_\mu$ are shown in figure 6.

The tolerances for LSTRS were $\varepsilon_\Delta = 10^{-2}$ and $\varepsilon_{\mathrm{HC}} = 10^{-1}$. The tolerances for TRUST$_\mu$ were $\varepsilon_y = 10^{-2}$ and $\varepsilon_f = \varepsilon_x = 10^{-5}$. The update of μ according to (14) was computed with $\sigma = 0.01$. The initial value was computed as $x_0 = x_{\mathrm{TRS}}$, with negative and zero components replaced by 10^{-5}.

LSTRS required seven iterations and 872 matrix–vector products to compute x_{TRS} in figure 6 (bottom left), which has a relative error of 1.6743×10^{-1} with respect to x_{IP}. TRUST$_\mu$



True Image

Data

LSTRS restoration

TRUST$_\mu$ restoration

Figure 6. Problem: star cluster, $n = 65\,536$. Relative error in LSTRS solution, 1.6743×10^{-1}; TRUST$_\mu$ solution, 1.2358×10^{-1}.

required one iteration of the main loop in algorithm 3.1, eight LSTRS iterations and 973 matrix–vector products to compute the solution shown in figure 6 (bottom right), which has a relative error of 1.2358×10^{-1} with respect to x_{IP}, and only positive components. The results are summarized in table 2. The value of μ was of the order of 10^{-7}. The storage requirement was 11 vectors of dimension 65 536. The relative error in $x_{\mathrm{TRS\text{-}P}}$ with respect to x_μ was 3.4153×10^{-1}. In this example we observed the same behaviour as in section 4.3; namely, with a moderate additional cost over the trust-region solution, we computed a positive solution with improved accuracy.

4.4.2. Satellite. In this section we present another example from astronomical imaging: restoring the image of a satellite in space. The problem was developed at the US Air Force Phillips Laboratory, Laser and Imaging Directorate, Kirtland Air Force Base New Mexico, and is available from [18]. More details about the problem can be found in [14] and the references therein. The imaging system is again modelled as an integral equation of the first kind (17), and is space invariant. Therefore only one PSF was used to construct the blurring operator. The discretized problem is of dimension $n = 65\,536$.

We first used LSTRS to compute the initial values as before. In this case, the strategy produced a not too clear solution and required 12 iterations and 1363 matrix–vector products.

Table 2. Performance of LSTRS and TRUST$_\mu$ on the star-cluster problem.

Method	Matrix–vector products	LSTRS iterations	Relative error in solution
LSTRS	872	7	1.6743×10^{-1}
TRUST$_\mu$	973	8	1.2358×10^{-1}
TRUST$_\mu$ iterations: 2 (initialization and one iteration of the main loop)			

Table 3. Performance of LSTRS and TRUST$_\mu$ on satellite problem.

Method	Matrix–vector products	Relative error in solution
CGLS	51	3.5995×10^{-1}
TRUST$_\mu$	51	3.5848×10^{-1}
TRUST$_\mu$ iterations: 1 (initialization step)		

We believe that the performance of LSTRS can be greatly improved by investigating the features of this problem. However, our purpose here was a preliminary study of the performance of the method TRUST$_\mu$, and not of the particular test problems.

We then tried the following strategy to compute initial values for TRUST$_\mu$. Since in the previous example an interior solution was detected by LSTRS, we decided to compute the initial values by first computing x_{LS}, an approximate solution to the unconstrained least-squares problem. We then set $x_0 = |x_{\text{LS}}|$ with zero components replaced by 10^{-5}. Since x_{LS} corresponds to an interior solution of the trust-region subproblem (15), we set $\lambda = 0$ and $\alpha = -g^T x_{\text{LS}}$. The tolerances for TRUST$_\mu$ and the value of σ were as in section 4.4.1.

We used the conjugate gradient method on the normal equations (CGLS) [2, 12] to solve the normal equations to a prescribed accuracy of the least-squares residual. CGLS required 51 matrix–vector products and so did TRUST $_\mu$, because in this case the initial iterate x_0 already satisfied the stopping criteria. The value of μ was of the order of 10^{-16}. The storage requirement was five vectors of dimension 65 536. The relative error in $x_{\text{TRS-P}}$ with respect to x_μ was 1.0226×10^{-1}. Figure 7 shows the true image, the data and the LSTRS and TRUST$_\mu$ restorations. The performances are summarized in table 3.

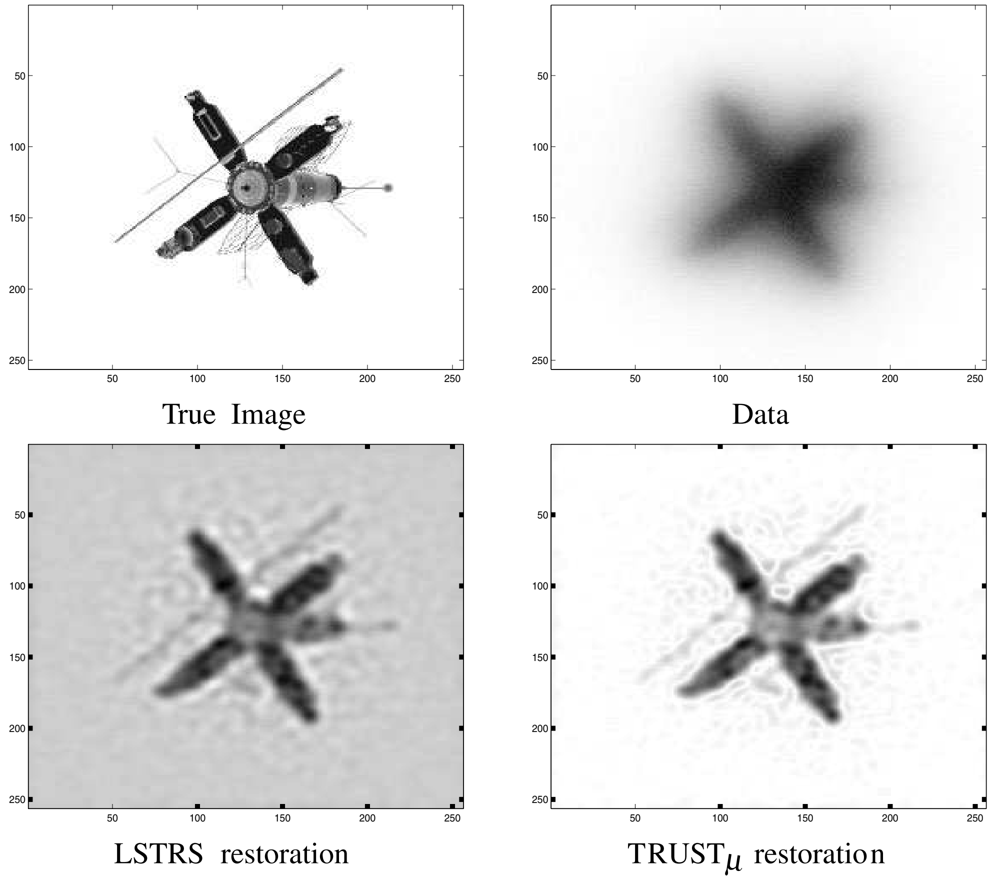
5. Extensions

Several extensions of problem (5) can be solved with TRUST$_\mu$. We describe three such generalizations in this section.

In the first place, we observe that we did not make any assumptions on convexity in the derivation of the algorithm. Thus, the method can be used to solve general quadratically and non-negatively constrained quadratic problems in which the Hessian is indefinite. This is a very desirable feature since it is often the case in ill posed problems that round-off errors will turn a positive semidefinite matrix into an indefinite one.

We can also treat problems in which the norm constraint is of the form $\|Lx\| \leqslant \Delta$ when L is a rectangular, full-rank matrix. In this case, the methods in [8, 16] can be used to transform the problem into the standard form with $L = I$.

Finally, we note that general linear inequality constraints can be handled in a straightforward manner. Assume that the constraints are of the form $G^T x \geqslant c$, where $G \in \mathbb{R}^{n\times p}$ and $c \in \mathbb{R}^p$, and let $\omega = G^T x$. Then, we only need to replace X^{-1} by $G^T \text{diag}(w)^{-1}$, and X^{-2} by $G^T \text{diag}(w)^{-2} G$. These changes only affect the gradient vector and



True Image

Data

LSTRS restoration

TRUST$_{\mu}$ restoration

Figure 7. Problem: satellite, $n = 65\,536$. Relative error in LSTRS solution, 0.3599; TRUST$_{\mu}$ solution, 0.3585.

the Hessian (i.e. the matrix–vector products) in the trust-region problem (9), and, therefore, can be incorporated in a straightforward way into the algorithm.

6. Conclusions

We presented the method TRUST$_{\mu}$ for large-scale non-negative regularization. The method combines interior-point and trust-region strategies to solve a quadratic problem with a norm constraint and non-negativity constraints. The method is not based on a heuristic, and it does not depend on the availability of a preconditioner. The method relies only upon matrix–vector products with the Hessian matrix, and has low and fixed storage requirements.

We used our method TRUST$_{\mu}$ to compute regularized non-negative solutions to inverse problems, including test problems from astronomical imaging. For the problems considered, TRUST$_{\mu}$ computed positive restorations with moderate computational cost and improved accuracy (see also [28]). Although more experiments are needed to assess the effectiveness of the method, the initial results are encouraging and present TRUST$_{\mu}$ as a promising method for large-scale non-negative regularization.

Acknowledgments

MR (partly) and TS were supported by the Research Council of Norway. We would like to thank Jim Nagy for providing the examples in section 4.4. We also thank Per Christian Hansen for providing valuable references, and for very helpful comments on an earlier version of this manuscript. Thanks also to Lars Eldén and Per Christian Hansen for interesting discussions about the extensions to general linear inequality constraints. Finally, we would like to thank the three anonymous referees for their insightful comments and suggestions.

References

[1] Bertero M and Bocacci P 1998 *Introduction to Inverse Problems in Imaging* (Bristol: Institute of Physics Publishing)

[2] Björck Å 1996 *Numerical Methods for Least Squares Problems* (Philadelphia, PA: SIAM)

[3] Calvetti D, Golub G H and Reichel L 1999 Estimation of the L-curve via Lanczos bidiagonalization *BIT* **39** 603–19

[4] Calvetti D, Reichel L and Zhang Q 1999 Iterative exponential filtering for large discrete ill-posed problems *Numer. Math.* (*Online First, DOI*10.1007/s002119900077*(July 28)*)

[5] Carpenter T J and Shanno D F 1993 An interior point method for quadratic programs based on conjugate projected gradients *Comput. Optim. Appl.* **2** 5–28

[6] Coleman T F and Li Y 1996 An interior trust region approach for nonlinear minimization subject to bounds *SIAM J. Optim.* **6** 418–45

[7] de Villiers G D, McNally B and Pike E R 1999 Positive solutions to linear inverse problems *Inverse Problems* **15** 615–35

[8] Eldén L 1977 Algorithms for the regularization of ill-conditioned least squares problems *BIT* **17** 134–45

[9] Fehmers G C, Kamp L P J and Sluijter F W 1998 An algorithm for quadratic optimization with one quadratic constraints and bounds on the variables *Inverse Problems* **14** 893–901

[10] Fiacco A V and McCormick G P 1990 *Nonlinear Programming: Sequential Unconstrained Minimization (Classics in Applied Mathematics Series)* (Philadelphia, PA: SIAM)

[11] Gay D 1981 Computing optimal locally constrained steps *SIAM J. Sci. Stat. Comput.* **2** 186–97

[12] Golub G H and Van Loan C F 1996 *Matrix Computations* 3rd edn (Baltimore, MD: John Hopkins University Press)

[13] Gockenbach M S and Symes W W 1997 Duality for inverse problems in wave propagation *Large Scale Optimization (New York Institute for Mathematics and its Applications)* ed L Biegler, T Coleman, F Santosa and A Conn (New York: Springer) pp 37–61

[14] Hanke M, Nagy J G and Vogel C 2000 Quasi-Newton approach to nonnegative image restorations *Linear. Algebr. Appl.* **316** 223–36

[15] Hansen P C 1994 Regularization tools: a MATLAB package for analysis and solution of discrete ill-posed problems *Numer. Algorithms* **6** 1–35

[16] Hansen P C 1997 Rank-deficient and discrete ill-posed problems *Numerical Aspects of Linear Inversion* (Philadelphia, PA: SIAM)

[17] Kilmer M E and O'Leary D P 2001 Choosing regularization parameters in iterative methods for ill-posed problems *SIAM J. Matrix Anal. Appl.* **22** 1204–21

[18] Lee K P, Nagy J G and Perrone L Iterative methods for image restoration: a Matlab object oriented approach, available from http://www.mathcs.emory.edu/∼nagy/RestoreTools/index.html

[19] Lehoucq R B, Sorensen D C and Yang C 1998 *ARPACK User's Guide: Solution of Large Scale Eigenvalue Problems by Implicitly Restarted Arnoldi Methods* (Philadelphia, PA: SIAM)

[20] Moré J J and Toraldo G 1991 On the solution of large quadratic programming problems with bound constraints *SIAM J. Optim.* **1** 93–113

[21] Nagy J G and O'Leary D P 1998 Restoring images degraded by spatially-variant blur *SIAM J. Sci. Comput.* **19** 1063–82

[22] Nagy J G and Strakos Z 2000 Enforcing nonnegativity in image reconstruction algorithms *Mathematical Modeling, Estimation, and Imaging* vol 4121, ed D C Wilson *et al* pp 182–90

[23] Parks T W and Burrus C S 1987 *Digital Filter Design* (New York: Wiley)

[24] Piana M and Bertero M 1997 Projected Landweber method and preconditioning *Inverse Problems* **13** 441–64

[25] Rojas M 1998 A large-scale trust-region approach to the regularization of discrete ill-posed problems *PhD Thesis*, *Technical Report* TR98-19 Department of Computational and Applied Mathematics, Rice University, Houston

[26] Rojas M, Santos S A and Sorensen D C 2000 A new matrix-free algorithm for the large-scale trust-region subproblem *SIAM J. Optim.* **11** 611–46

[27] Rojas M and Sorensen D C 2002 A trust-region approach to the regularization of large-scale discrete forms of ill-posed problems *SIAM J. Sci. Comput.* **26** 1843–61

[28] Rojas M and Steihaug T 2002 Large-scale optimization techniques for nonnegative image restorations *Proc. SPIE* **4791** at press

[29] Sorensen D C 1982 Newton's method with a model trust region modification *SIAM J. Numer. Anal.* **19** 409–26

[30] Sorensen D C 1992 Implicit application of polynomial filters in a k-step Arnoldi method *SIAM J. Matrix Anal. Appl.* **13** 357–85

[31] Tikhonov A N, Goncharsky A V, Stepanov V V and Yagola A G 1995 *Numerical Methods for the Solution of Ill-posed Problems* (Dordrecht: Kluwer)

[32] Ye Y 1997 *Interior Point Algorithms: Theory and Analysis* (New York: Wiley)

[33] Youla D C and Webb H 1982 Image restoration by the method of convex projections: part 1. Theory *IEEE Trans. Med. Imaging* **1** 81–94