Consiglio Nazionale delle Ricerche

Image regularization by nonnegatively constrained conjugate gradient

P. Favati, G. Lotti, O. Menchi, F. Romani

IIT TR-04/2017

Technical Report

Giugno 2017

Istituto di Informatica e Telematica

Image regularization by nonnegatively constrained conjugate gradient

P. Favati [a,*], G. Lotti [b], O. Menchi [c], F. Romani [c]

[a] *IIT - CNR Via G. Moruzzi 1, 56124 Pisa, Italy.*

[b] *Dip. di Matematica, University of Parma, Parco Area delle Scienze 53/A, 43124 Parma, Italy.*

[c] *Dip. di Informatica, University of Pisa, Largo Pontecorvo 3, 56127 Pisa, Italy.*

Abstract

In the image reconstruction context the nonnegativity of the computed solution is often required. Conjugate Gradient (`CG`), used as a reliable regularization tool, may give solutions with negative entries, particularly evident when large nearly zero plateaus are present. The active constraints set, detected by projection onto the nonnegative orthant, turns out to be largely incomplete leading to poor effects on the accuracy of the reconstructed image. In this paper an inner-outer method based on `CG` is proposed to compute nonnegative reconstructed images with a strategy which enlarges subsequently the active constraints set. This method appears to be especially suitable for the reconstruction of images having large nearly zero backgrounds. The numerical experimentation validates the effectiveness of the proposed method when compared to other strategies for nonnegative reconstruction.

Key words: Image Reconstruction, Conjugate Gradient, Nonnegativity Constraints.

1 Introduction

A Fredholm integral equation of the first kind

$$g(s) = \int \widehat{K}(s,t) f(t)\, dt \tag{1}$$

* Corresponding author.
Email addresses: `paola.favati@iit.cnr.it` (P. Favati), `grazia.lotti@unipr.it` (G. Lotti), `menchi@di.unipi.it` (O. Menchi), `romani@di.unipi.it` (F. Romani).

is often used for modeling the image formation process, where $f(t)$ and $g(s)$ represent a real object and its image, respectively. The kernel $K(s,t)$, called the *point spread function* (PSF) and assumed to be square integrable, represents the imaging system and is responsible for the blurring of the image. In practical applications the *blurred image* $g(s)$ is not available, being replaced by a finite set $\boldsymbol{g}$ of measured quantities, and is degraded by the noise which affects the process of image recording. Hence the problem of restoring $f(t)$ from $\boldsymbol{g}$ is an ill-posed problem. The linear system obtained by the discretization of (1) inherits this feature, in the sense that the resulting matrix is severely ill-conditioned, and regularization methods must be used to solve it [2,14]. This kind of problem arises for example in the reconstruction of astronomical images taken by a telescope and of medical and microscopy images.

One of the main features of the problem is the nonnegativity of the functions involved in (1). When discretized, the equation leads to a linear problem whose solution is constrained to be nonnegative. Iterative methods applied as regularization techniques may give solutions with negative entries. A projection onto the nonnegative orthant may have poor effects on the accuracy of the reconstructed image. In this paper an inner-outer method based on `CG` is proposed to compute nonnegative reconstructed images with a strategy which enlarges subsequently the active constraints set. This method appears to be especially suitable for images having large zero backgrounds. For this type of problems, one naturally wonders if the zeros of the original image are correctly reconstructed. As a matter of fact, an algorithm can fail by putting to zero nonzero values of the original image (false positive) or by giving nonzero values in correspondence to zero values of the original image (false negative). This behavior will be analyzed using the well-known measures of Information Retrieval, namely the $\mathtt{F}_1$ score, which takes into account both types of errors.

The outline of the paper is the following: first, in Section 2 the problem under consideration is introduced and in Section 3 some strategies for nonnegative regularization present in the literature are recalled. In Section 4 our proposed inner-outer algorithm is motivated and described. In Section 5 the results of a numerical experimentation which compares the performance of our algorithm with those of three chosen projection methods are presented and discussed.

Notation: Throughout the paper, $\|\boldsymbol{v}\|$ denotes the Euclidean norm of a vector $\boldsymbol{v}$, i.e. $\|\boldsymbol{v}\|^2 = \boldsymbol{v}^T\boldsymbol{v}$. The elementwise multiplication and division between two vectors are denoted by $\odot$ and $\oslash$.

2 The problem

Let $\widehat{\boldsymbol{b}} = A\widehat{\boldsymbol{x}}$ be the discretized version of equation (1). In image reconstruction problems the N-vector $\widehat{\boldsymbol{x}}$ stores columnwise the pixels of an $n \times n$ original object, with $N = n^2$, and $\widehat{\boldsymbol{b}}$ analogously stores the blurred image. The imaging system is represented by a large not necessarily square matrix A, often severely ill-conditioned with singular values decaying to zero without significant gap to indicate numerical rank. Matrix A might not be explicitly available, as long as the products $A\boldsymbol{x}$ and $A^T\boldsymbol{x}$ are computable for any vector $\boldsymbol{x}$. A common special case is the one which occurs when the PSF is bandlimited space invariant, i.e. invariant with respect to translations and with a bounded support, and A is a square 2-level Toeplitz matrix with a limited bandwidth. Moreover, it may happen that the image has sufficiently large zero background along the boundary, so that periodic boundary conditions can be safely imposed. In this case A becomes a 2-level circulant matrix and the matrix-by-vector product can be computed by low cost Fourier transforms. This is the structure we assume for our numerical examples, but the proposed algorithm can be applied equally well to general matrices A.

Vector $\widehat{\boldsymbol{b}}$ is not exactly known, because of the noise due to the fluctuations in the counting process of the acquisition of the image which obeys to Poisson statistics, and to the readout noise caused by imperfections of the recording device which is Gaussian distributed. Hence only a noisy image $\boldsymbol{b} = \widehat{\boldsymbol{b}} + \boldsymbol{\eta}$ is available, where the noise level is measured by

$$\eta = \frac{\|\boldsymbol{b} - \widehat{\boldsymbol{b}}\|}{\|\widehat{\boldsymbol{b}}\|}, \tag{2}$$

and in some cases can be roughly estimated. The system to be solved is thus

$$A\boldsymbol{x} = \boldsymbol{b}. \tag{3}$$

The ith component of the vectors $\widehat{\boldsymbol{x}}$, $\widehat{\boldsymbol{b}}$ and $\boldsymbol{b}$ represents respectively the light intensity or the radiation emitted by the ith pixel of the object, arriving at the ith pixel of the blurred image and recorded in the ith pixel of the noisy image. The component a_{ij} of matrix A measures the fraction of the light or of the rays emitted by the ith pixel of the object which arrives at the jth pixel of the image. Because of the ill-conditioning of A and of the presence of the noise, the solution $A^\dagger\,\boldsymbol{b}$ of (3), where $A^\dagger$ is the Moore-Penrose generalized inverse, may be quite different from the original image $\widehat{\boldsymbol{x}}$.

All the quantities involved in the problem, i.e. A, $\widehat{\boldsymbol{x}}$, $\widehat{\boldsymbol{b}}$ and $\boldsymbol{b}$, are assumed componentwise nonnegative and it is reasonable to expect the approximation

of $\widehat{\boldsymbol{x}}$ obtained by solving (3) to be nonnegative. The constrained *least squares* approximation to $\widehat{\boldsymbol{x}}$ is given by

$$\boldsymbol{x}_{ls} = \underset{\boldsymbol{x}\geq\boldsymbol{0}}{\operatorname{argmin}}\ \phi(\boldsymbol{x}), \quad \text{where} \quad \phi(\boldsymbol{x}) = \frac{1}{2}\,\|\boldsymbol{b} - A\boldsymbol{x}\|^2. \tag{4}$$

The gradient and the Hessian of $\phi(\boldsymbol{x})$ are

$$\operatorname{grad}_x \phi(\boldsymbol{x}) = A^T A\boldsymbol{x} - A^T\boldsymbol{b}, \qquad H(\phi(\boldsymbol{x})) = A^T A.$$

Since $H(\phi(\boldsymbol{x}))$ is positive semidefinite, $\phi(\boldsymbol{x})$ is convex. Its minimum points are found by solving the system $\operatorname{grad}_x \phi(\boldsymbol{x}) = 0$, i.e. the so-called *normal equations*

$$A^T A\boldsymbol{x} = A^T\boldsymbol{b}. \tag{5}$$

Due to the large dimension of system (5) and to the presence of the noise $\boldsymbol{\eta}$, a regularization method must be employed, coupled with suitable strategies for enforcing nonnegativity. Iterative methods enjoying the semiconvergence property are often used. According to this property, an integer K exists such that the vectors $\boldsymbol{x}_k$ computed in the first K iterations are minimally affected by the noise and approach solution $\widehat{\boldsymbol{x}}$. After the Kth iteration, the vectors $\boldsymbol{x}_k$ are progressively contaminated by the noise and move away from $\widehat{\boldsymbol{x}}$ toward $A^\dagger\boldsymbol{b}$ which can be largely different from $A^\dagger\widehat{\boldsymbol{b}}$. A good terminating procedure is hence needed to detect the correct index K where to stop the iteration.

In the following we assume that both $A\boldsymbol{x} \neq \boldsymbol{0}$ and $A^T\boldsymbol{x} \neq \boldsymbol{0}$ for any $\boldsymbol{x} \geq \boldsymbol{0}$ with $\boldsymbol{x} \neq \boldsymbol{0}$, and that $A\boldsymbol{e} > \boldsymbol{0}$ and $A^T\boldsymbol{e} > \boldsymbol{0}$, where $\boldsymbol{e}$ is the vector of all ones (i.e. the sums by rows and columns of A are all nonzero).

3 Nonnegativity strategies

We recall here how classic and more recent nonnegativity techniques can be applied to iterative regularization methods.

Actually, the two classic methods "Expectation Maximization" (`EM`) [21,17,23] and "Iterative Space Reconstruction Algorithm" (`ISRA`) [5] enjoy the nonnegativity feature in a natural way when $\boldsymbol{x}_0 > \boldsymbol{0}$, in the sense that the zero components of $\boldsymbol{x}_k$ produced at kth iteration are kept at all the subsequent iterations. Unfortunately, they suffer from slow convergence rate, as shown in [9].

Descent methods have the form

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k, \tag{6}$$

where the direction $\boldsymbol{p}_k$ satisfies $\boldsymbol{p}_k^T \boldsymbol{g}_k < 0$, with $\boldsymbol{g}_k = \mathrm{grad}_x\, \phi(\boldsymbol{x}_k)$, and the step size α_k is chosen in such a way that $\phi(\boldsymbol{x}_{k+1}) < \phi(\boldsymbol{x}_k)$. The iteration is carried out for $k = 0, \ldots, K-1$ steps, until $\boldsymbol{x}_K$ satisfies the stopping condition imposed by the regularization request. The choice $\alpha_k = -\boldsymbol{g}_k^T \boldsymbol{p}_k \,/\, \|A\boldsymbol{p}_k\|^2$ satisfies the minimum problem

$$\phi(\boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k) = \min_{\alpha} \phi(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k),$$

but the vectors $\boldsymbol{x}_k$ so computed may have negative components even if $\widehat{\boldsymbol{x}}$ is nonnegative.

A simple technique for imposing nonnegativity consists in choosing a reduced step size α_k to guarantee that $\boldsymbol{x}_{k+1} \geq \boldsymbol{0}$ if $\boldsymbol{x}_k \geq \boldsymbol{0}$. This technique, for example, is implemented in `MRNSD`, which is a modified version of the "Residual Norm Steepest Descent" method (`RNSD`) [1,19]. The drawback of this technique is that small α_k may occur, inducing slow convergence rate.

Another technique consists in performing at each iteration the projection of $\boldsymbol{x}_k$ onto the nonnegative orthant by setting to zero its negative components.

The Conjugate Gradient method (`CG`) is known for its good convergence rate and its regularizing properties (see [13,14,20]). `CG` has the form (6) with directions $\boldsymbol{p}_k$ that are A^TA-conjugate. Unfortunately, if nearly zero plateaus are present in the original image $\widehat{\boldsymbol{x}}$, `CG` may produce many negative components in the neighborhood of these areas, so a projection step should be performed. Theoretically, the projection should be applied at each iteration, but in the case of `CG` the directions $\boldsymbol{p}_k$ would stop being A^TA-conjugate, i.e. the method would lose its most important feature, which is the basis of its success. For this reason, when `CG` is used, the projection is only applied to the final iteration $\boldsymbol{x}_K$ which satisfies the given stopping condition, but its projected vector $\widetilde{\boldsymbol{x}}_K$ might not satisfy the stopping condition. In this event $\widetilde{\boldsymbol{x}}_K$ could not be accepted as a regularized solution.

Thanks to its fast convergence rate, `CG` is used as a basis for regularizing nonnegative schemes. As an inner-outer scheme, [4] suggests to apply an iterative improvement to $\widetilde{\boldsymbol{x}}_K$. The residual vector $\boldsymbol{r}_K = \boldsymbol{b} - A\widetilde{\boldsymbol{x}}_K$ is computed and the system $A\boldsymbol{y} = -\boldsymbol{r}_K$ is solved using again `CG` with a zero starting vector. The regularized solution is used to update $\widetilde{\boldsymbol{x}}_K$. If the vector so obtained still has negative components, it is projected and the improvement step is repeated. In [18] this first scheme is called "Projected Restarted Conjugate Gradient"

(PRCG) and is tested against a more refined scheme which restricts the update to the components corresponding to the nonactive constraints having positive Lagrangian multipliers. This second scheme is called Algorithm 1 in [18] (for simplicity we call it here Alg1). Also the algorithm we are going to propose will adopt an inner-outer scheme.

The projection is used also by the "Scaled Gradient Projection" (SGP) methods [3]. The methods of this class do not adopt an inner-outer scheme and have the form (6) with directions $\boldsymbol{p}_k$ and step size α_k which depend on a diagonal scaling matrix. Their performances have been analyzed in [6] and have shown good reconstruction accuracy and convergence rate. One of the most effective methods generalizes ISRA and is identified by the acronym SGP-GcB in [6] (for simplicity we call it here just SGP). The direction is computed in the following way. Denoting $\boldsymbol{w}_k = \boldsymbol{x}_k \oslash (A^T A \boldsymbol{x}_k)$, $\boldsymbol{s}_k = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$ and $\boldsymbol{z}_k = \boldsymbol{w}_k \odot (\boldsymbol{g}_k - \boldsymbol{g}_{k-1})$, the vector

$$\boldsymbol{u}_k = \boldsymbol{x}_k - \gamma\, \boldsymbol{w}_k \odot \boldsymbol{g}_k, \quad \text{where} \quad \gamma = \boldsymbol{s}_k^T \boldsymbol{z}_k \,/\, \|\boldsymbol{z}_k\|^2,$$

is computed. Let $\overline{\boldsymbol{u}}_k$ be its projection. The direction and step size at the kth step are

$$\boldsymbol{p}_k = \overline{\boldsymbol{u}}_k - \boldsymbol{x}_k, \quad \text{where} \quad \alpha_k = \max\{\theta, \min\{1, -\boldsymbol{g}_k^T \boldsymbol{p}_k \,/\, \|A\boldsymbol{p}_k\|^2\}\},$$

and $\theta > 0$ is a bound for α_k from below (typically $\theta = 10^{-3}$). The bounds on α_k guarantee that $\boldsymbol{x}_{k+1} \geq \boldsymbol{0}$ if $\boldsymbol{x}_k \geq \boldsymbol{0}$. Note that this method applies the projection to the direction $\boldsymbol{u}_k$ and not to the approximated solution $\boldsymbol{x}_k$.

On the basis of the convergence rate, the three methods PRCG, Alg1 and SGP will be considered in order to test the performance of our algorithm.

4 The algorithm we propose

In this section we describe an algorithm, called "Inner-Outer CG" (IOCG), based on restarted CG, coupled with a projection technique which exploits both the regularizing properties and the good convergence rate of CG.

IOCG and Alg1 share the same inner-outer scheme, with CG used for the inner process and an active set based technique for dealing with the constraints. The main difference is that Alg1 lets the constraints enter or leave the active set at each outer step, while with IOCG the constraints that belong to the active set are not allowed to leave it in the subsequent outer steps and the active set can only expand. The progressive enlargement of the active set

ensures convergence. In a certain sense, this behavior reminds the `EM` and `ISRA` behaviors.

`IOCG` detects a sequence of progressively enlarged sets $\mathcal{A}^{(h)}$, $h = 0, 1, \ldots, H$ of active constraints for the solution. On each set $\mathcal{A}^{(h)}$ the algorithm computes a regularized solution $\boldsymbol{y}^{(h)}$ with the components belonging to $\mathcal{A}^{(h)}$ fixed to zero, stopping when no further enlargement is required.

Note: the projection $\widetilde{\boldsymbol{y}}$ of a vector $\boldsymbol{y}$ onto the nonnegative orthant can be formally implemented by means of a projection matrix P, i.e. a diagonal matrix of zeros and ones. In the code of `IOCG` we will refer to a function `project` which constructs $\widetilde{\boldsymbol{y}}$ and the diagonal $\boldsymbol{d}$ of P.

4.1 The outline

We recall here some basic facts on which `IOCG` relies. In the formulas a permutation matrix $\mathit{\Pi}$ appears, having only the descriptive role of shifting the zeros upward, in order to simplify the notation. Since its use is irrelevant, no reference to $\mathit{\Pi}$ will appear in the code.

Let $\boldsymbol{x}_k$, $k = 1, 2, \ldots$, be the sequence of vectors obtained by applying `CG` to the normal equations (5), starting with a vector $\boldsymbol{x}_0$. Because of the semiconvergence property, an *optimal* index K exists such that

$$\|\widehat{\boldsymbol{x}} - \boldsymbol{x}_K\| \leq \|\widehat{\boldsymbol{x}} - \boldsymbol{x}_k\|, \quad \text{for any } k.$$

If $\boldsymbol{x}_K \geq 0$, then it is accepted as the regularized solution of (4). Otherwise, let P be its projection matrix and $\widetilde{\boldsymbol{x}}_K = P\boldsymbol{x}_K$. Since $\widehat{\boldsymbol{x}} \geq \boldsymbol{0}$, it is

$$\|\widehat{\boldsymbol{x}} - P\boldsymbol{x}_K\| < \|\widehat{\boldsymbol{x}} - \boldsymbol{x}_K\|.$$

Let $\mathit{\Pi}$ be a permutation matrix such that

$$\mathit{\Pi}\, P\boldsymbol{x}_K = \begin{bmatrix} \boldsymbol{0} \\ \underline{\boldsymbol{x}}_K \end{bmatrix}, \quad \text{with} \quad \underline{\boldsymbol{x}}_K > 0.$$

Applying the same projection and permutation to $\widehat{\boldsymbol{x}}$ we have

$$\mathit{\Pi} P\widehat{\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{0} \\ \underline{\boldsymbol{x}} \end{bmatrix} \quad \text{where} \quad \mathit{\Pi}\widehat{\boldsymbol{x}} = \begin{bmatrix} \overline{\boldsymbol{x}} \\ \underline{\boldsymbol{x}} \end{bmatrix},$$

then

$$\|\widehat{\boldsymbol{x}} - P\boldsymbol{x}_K\|^2 = \|\Pi\widehat{\boldsymbol{x}} - \Pi\, P\boldsymbol{x}_K\|^2 = \|\overline{\boldsymbol{x}}\|^2 + \|\underline{\boldsymbol{x}} - \underline{\boldsymbol{x}}_K\|^2. \tag{7}$$

The basic assumption on which `IOCG` relies is that the negative components of $\boldsymbol{x}_K$, which have been set to zero in $\widetilde{\boldsymbol{x}}_K$, really correspond to nearly zero components of $\widehat{\boldsymbol{x}}$, so that the zero components of $P\boldsymbol{x}_K$ are correctly placed in the active constraints set $\mathcal{A}$. The experimentation will show that this assumption is reasonable for images having a large zero background.

The matrix $AP\Pi^T = [\,O\,|\,\underline{A}\,]$ is obtained by zeroing the columns with the indices in $\mathcal{A}$ and shifting them to the left. Then

$$\Pi PA^TAP\Pi^T = \left[\begin{array}{c|c} O & O \\ \hline O & \underline{A}^T\underline{A} \end{array}\right],$$

and system (5) becomes $\Pi PA^TAP\Pi^T\boldsymbol{y} = \Pi PA^T\boldsymbol{b}$, where $\boldsymbol{x} = P\Pi^T\boldsymbol{y}$. Setting

$$\boldsymbol{y} = \begin{bmatrix} \overline{\boldsymbol{y}} \\ \underline{\boldsymbol{y}} \end{bmatrix} \quad \text{and} \quad \Pi PA^T\boldsymbol{b} = \begin{bmatrix} \boldsymbol{0} \\ \underline{\boldsymbol{c}} \end{bmatrix},$$

we get

$$\underline{A}^T\underline{A}\,\underline{\boldsymbol{y}} = \underline{\boldsymbol{c}}. \tag{8}$$

The smallest $\|\overline{\boldsymbol{x}}\|$ in (7), the most effective the approximation of $\underline{\boldsymbol{x}}$ obtained by solving (8) in a regularization sense. Then we apply `CG` to system (8) starting with $\underline{\boldsymbol{y}}_0 = \underline{\boldsymbol{x}}_K$. The semiconvergence will push the iterates $\underline{\boldsymbol{y}}_k$ toward $\underline{\boldsymbol{x}}$, i.e. after K' iterations a vector $\underline{\boldsymbol{y}}_{K'}$ is obtained such that $\|\underline{\boldsymbol{x}} - \underline{\boldsymbol{y}}_{K'}\| \le \|\underline{\boldsymbol{x}} - \underline{\boldsymbol{y}}_k\|$, for any k. The vector

$$\boldsymbol{y}_{K'} = \begin{bmatrix} \boldsymbol{0} \\ \underline{\boldsymbol{y}}_{K'} \end{bmatrix}$$

verifies

$$\|\Pi\widehat{\boldsymbol{x}} - \boldsymbol{y}_{K'}\| \le \|\Pi\widehat{\boldsymbol{x}} - \Pi\, P\boldsymbol{x}_K\| = \|\widehat{\boldsymbol{x}} - P\boldsymbol{x}_K\|.$$

If also $\underline{\boldsymbol{y}}_{K'}$ has negative components, the previous arguments are repeated to further improve the approximation to $\widehat{\boldsymbol{x}}$ starting from $\boldsymbol{y}_{K'}$, and so on.

4.2 The inner-outer structure

The previous considerations suggest an algorithm of inner-outer type for finding a regularized nonnegative solution of problem (4), which appears to be especially suitable for the reconstruction of images having large nearly zero backgrounds. The algorithm consists of two loops:

(a) the outer loop generates a sequence of projection matrices $P^{(h)}$, $h = 0, 1, \ldots$, starting with $P^{(0)} = I$ and for any h calls the inner loop;

(b) starting with $\boldsymbol{x}_0^{(h)} \geq \mathbf{0}$, the inner loop computes the sequence $\boldsymbol{x}_k^{(h)}$, $k = 1, \ldots, K_h$ by applying `CG` to the system

$$A^{(h)T} A^{(h)} \boldsymbol{x} = A^{(h)T} \boldsymbol{b}, \quad \text{where} \quad A^{(h)} = AP^{(h)}. \tag{9}$$

The first inner loop is stopped at index K_0 according to a suitable inner stopping rule. The vector $\boldsymbol{y}^{(0)} = \boldsymbol{x}_{K_0}^{(0)}$ is assumed as the regularized solution of (9) for $h = 0$. If $\boldsymbol{y}^{(0)}$ has negative components, a new inner loop starts: the vector $\boldsymbol{y}^{(0)}$ is projected and the corresponding projection matrix $P^{(1)}$ is constructed. The starting vector for the new inner loop is $\boldsymbol{x}_0^{(1)} = P^{(1)} \boldsymbol{y}^{(0)}$ and a new regularized solution $\boldsymbol{y}^{(1)} = \boldsymbol{x}_{K_1}^{(1)}$ of (9) is computed for $h = 1$, and so on.

As the outer loop goes on, the hth initial vector $\boldsymbol{x}_0^{(h)}$ has more zeros than the previous initial vector $\boldsymbol{x}_0^{(h-1)}$. The stopping condition for the outer loop is satisfied by the first h such that $\boldsymbol{y}^{(h)}$ has all nonnegative components. This vector is assumed as the regularized solution $\boldsymbol{x}_{reg}$ of problem (3).

Thanks to the inner-outer strategy, an excessive precision in the computation of the last $\boldsymbol{x}_{K_h}^{(h)}$ is not required at the hth outer phase. For this reason we will impose an upper bound to the number of iterations allowed during an inner loop through a parameter k_{max}.

The kth iteration of `CG` applied to (9) is

$$\begin{aligned}
&\boldsymbol{z}_k^{(h)} = A^{(h)} \boldsymbol{p}_k^{(h)}, \quad \alpha_k^{(h)} = \|\boldsymbol{q}_k^{(h)}\|^2 / \|\boldsymbol{z}_k^{(h)}\|^2, \quad \boldsymbol{x}_{k+1}^{(h)} = \boldsymbol{x}_k^{(h)} + \alpha_k^{(h)} \boldsymbol{p}_k^{(h)}, \\
&\boldsymbol{r}_{k+1}^{(h)} = \boldsymbol{r}_k^{(h)} - \alpha_k^{(h)} \boldsymbol{z}_k^{(h)}, \quad \boldsymbol{q}_{k+1}^{(h)} = A^{(h)T} \boldsymbol{r}_{k+1}^{(h)}, \quad \beta_k^{(h)} = \|\boldsymbol{q}_{k+1}^{(h)}\|^2 / \|\boldsymbol{q}_k^{(h)}\|^2, \\
&\boldsymbol{p}_{k+1}^{(h)} = \boldsymbol{q}_{k+1}^{(h)} + \beta_k^{(h)} \boldsymbol{p}_k^{(h)},
\end{aligned} \tag{10}$$

where $\boldsymbol{x}_0^{(h)}$ is the projection of $\boldsymbol{x}_{K_{h-1}}^{(h-1)}$, $\boldsymbol{r}_0^{(h)} = \boldsymbol{b} - A^{(h)} \boldsymbol{x}_0^{(h)}$ and $\boldsymbol{p}_0^{(h)} = \boldsymbol{q}_0^{(h)} = A^{(h)T} \boldsymbol{r}_0^{(h)}$.

Due to the presence of the zeros introduced by the matrix $P^{(h)}$, system (9) has an effective size smaller than the size N of system (5). This fact could be exploited to reduce the computational cost of `CG` at the hth inner loop. However, when A has a structure that can be exploited to reduce the cost of the matrix-vector products (which accounts for the greatest part of the computational cost of the method) and matrix $A^{(h)}$ loses this structure (as it happens with Toeplitz matrices), it is more convenient to keep the matrix A unchanged and to transfer the effect of $P^{(h)}$ to the vectors, by setting

$$\boldsymbol{q}_{k+1}^{(h)} = \boldsymbol{d}^{(h)} \odot A^T \boldsymbol{r}_{k+1}^{(h)}, \tag{11}$$

where $\boldsymbol{d}^{(h)}$ is the diagonal of $P^{(h)}$. By effect of the recursion, also the vectors $\boldsymbol{x}_k^{(h)}$ and $\boldsymbol{p}_k^{(h)}$ have zero in the positions indicated by $\boldsymbol{d}^{(h)}$ for any k if this holds for $k = 0$.

4.3 Implementation

A rough code for `IOCG` is given at the end of the section. It calls the function `inner` which applies `CG` with the stopping condition defined by `isc`. A function `project` is also used, but not coded.

For our experiments we have chosen Toeplitz matrices having the circulant structure which arises when periodic boundary conditions are set. For these matrices the product of A and A^T by a vector can be performed using a low cost algorithm based on the fast Fourier transform (FFT) [12].

A circulant matrix A of size m is diagonalized by the Fourier matrix $\mathcal{F}$, whose elements are

$$f_{r,s} = \frac{1}{\sqrt{m}}\,\omega^{rs}, \quad r, s = 0, \ldots, m-1, \quad \text{with} \quad \omega = \exp(2\pi \boldsymbol{i}/m).$$

Denoting by $\boldsymbol{a}^T$ the first row of A, it holds $A = \sqrt{m}\,\mathcal{F}\,\mathrm{diag}(\mathcal{F}\,\boldsymbol{a})\,\mathcal{F}^*$, where $\mathcal{F}^*$ is the inverse (i.e. transpose conjugate) of $\mathcal{F}$. Hence the product $\boldsymbol{z} = A\boldsymbol{v}$, where $\boldsymbol{v}$ and $\boldsymbol{z}$ are m-vectors, can be so computed

$$\widetilde{\boldsymbol{a}} = \mathcal{F}\boldsymbol{a}, \quad \widetilde{\boldsymbol{v}} = \mathcal{F}^*\boldsymbol{v}, \quad \boldsymbol{z} = \sqrt{m}\,\mathcal{F}\,(\widetilde{\boldsymbol{a}} \odot \widetilde{\boldsymbol{v}}). \tag{12}$$

To obtain the product $\boldsymbol{z} = A^T\boldsymbol{v}$ it is sufficient to take the first column of A as $\boldsymbol{a}$ or to replace $\widetilde{\boldsymbol{a}}$ by its conjugate $\widetilde{\boldsymbol{a}}^*$. The multiplications by $\mathcal{F}$ and $\mathcal{F}^*$ can be efficiently computed by calling two FFT routines, with a computational cost of order $O(m \log m)$ (in our case $m = n^2$). Using (12), the single step (10)

of `CG` requires 4 FFT computations and 7 vector or scalar by vector products, giving a cost of order $O(n^2 \log n)$ per iteration of the algorithm.

A further reduction of the cost when A is circulant is obtained by shifting the computation from the signal domain where A, $\boldsymbol{x}$ and $\boldsymbol{b}$ live to the frequency domain, i.e. by replacing matrices and vectors with their transformations in the Fourier basis. All the instructions of the k-th iteration of `CG` in (10) can be performed in the frequency domain, except instruction (11) which must be performed in the signal domain, as shown in the code of function `inner`. An appropriate *inner_stop_cond* is specified by means of function `isc`. The quantities $\widetilde{\boldsymbol{a}} = \mathcal{F}\boldsymbol{a}$ and $\widetilde{\boldsymbol{b}} = \mathcal{F}^*\boldsymbol{b}$ are global.

4.4 Stopping conditions

The execution flow of the algorithm is ruled by the stopping conditions for the outer and the inner loop. From a theoretical point of view the two stopping conditions could be independent, but in practice it is better to relate them. In fact, nearly stall situations where only few components are eliminated by each inner step for the current outer step should be avoided. This suggests to repeat the outer iterations until the Boolean

$$\mathit{outer\_stop\_cond} = (\min(\boldsymbol{y}^{(h)}) < \tau) \;\&\; (k_{in} > k_{min}) \;\&\; (h \leq h_{max})$$

becomes false. The constant τ is a projection threshold, k_{min} is a small fixed integer, k_{in} is the number of iterations performed in the current inner step, h_{max} bounds the number of allowed outer iterations (in the experiments we have set $\tau = -10^{-15}$, $k_{min} = 4$ and $h_{max} = 512$).

For the inner condition, we note that it is not worthwhile to solve (9) too accurately, for example by determining with a good precision small positive components which would be probably eliminated in a further step. Moreover, the choice of the index k at which the hth inner loop should be stopped may be critical. We assume a stopping condition of the form

$$\mathit{inner\_stop\_cond} = (\delta_k^{(h)} < \delta_{k-1}^{(h)}) \;\&\; (k \leq k_{max}), \tag{13}$$

where k_{max} bounds the number of allowed inner iterations and $\delta_k^{(h)}$ is a quantity that must be specified in such a way that its minimum is reached when the noise starts contaminating the computation. Ideally, the inner hth loop should be stopped when the relative error $\epsilon_k^{(h)} = \|\widehat{\boldsymbol{x}} - \boldsymbol{x}_k^{(h)}\| \,/\, \|\widehat{\boldsymbol{x}}\|$ starts increasing, but of course such an ideal condition cannot be exploited in a realistic context, where a more practical condition must be implemented.

For an acceptable approximation of the optimal index we use the *Generalized Cross Validation* method (GCV) described in [11,25]. The performance of GCV has been tested in [7,8] and it resulted that GCV is more effective than other stopping rules when information about the entity of the noise is not available.

Given a sequence $\boldsymbol{x}_k$ generated by an iterative method applied to solve system (3), the GCV functional is defined as

$$V_k = \frac{N \, \|\boldsymbol{b} - A\boldsymbol{x}_k\|^2}{(N - \mathrm{trace}\,(A_k))^2}\,, \tag{14}$$

where the *influence* matrix A_k is such that $A_k\boldsymbol{b} = A\boldsymbol{x}_k$. The minimizer of V_k can be taken as an estimate for the optimal index.

If the applied iterative method is `CG`, it is not easy to say how A_k depends on $\boldsymbol{b}$ and an approximated estimate of the trace of A_k must be provided. Different reliable techniques for estimating the trace were proposed in [8]. When A has a 2-level circulant structure, a simple technique can be obtained by looking for a 2-level circulant approximation C_k of A_k such that $A\boldsymbol{x}_k \approx C_k\,\boldsymbol{b}$. Denoting by $\boldsymbol{c}_k^T$ the first row of C_k, we have $C_k\,\boldsymbol{b} = \mathcal{F}\,\mathrm{diag}(\mathcal{F}\boldsymbol{c}_k)\,\mathcal{F}^*\boldsymbol{b}$, hence

$$\mathcal{F}\boldsymbol{c}_k \approx \mathcal{F}^*(A\boldsymbol{x}_k) \oslash \mathcal{F}^*\boldsymbol{b} = n\,\widetilde{\boldsymbol{a}} \oslash \widetilde{\boldsymbol{b}} \odot \widetilde{\boldsymbol{x}}_k, \quad \text{where} \quad \widetilde{\boldsymbol{x}}_k = \mathcal{F}^*\boldsymbol{x}_k,$$

and $\mathrm{trace}\,(A_k) \approx \sum_i (\mathcal{F}\boldsymbol{c}_k)_i$. Thus in the hth outer loop we use (13) with

$$\delta_k^{(h)} = V_k^{(h)} = N \, \|\boldsymbol{r}_k^{(h)}\|^2 / (N - \mathrm{trace}\,(A_k^{(h)}))^2. \tag{15}$$

A function `isc` which constructs the sequence $\delta_k^{(h)}$ according to (15) is included in the code.

Another stopping condition which could be applied is based on the *discrepancy rule*, according which the iteration should stop when the residual vector $\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k$ verifies $\|\boldsymbol{r}_k\| \le \vartheta\,\eta$, with $\vartheta \ge 1$ a constant and η as given in (2).

To validate the use of the GCV functional, we have performed a preliminary ad-hoc experimentation on the problems considered in the next section, for which the exact solution $\widehat{\boldsymbol{x}}$ is known. By comparing the minima of the sequences $\epsilon_k^{(h)}$ and $V_k^{(h)}$ obtained for fixed h, we have found that the two indications are comparable, even if in the majority of cases the index indicated by GCV is slightly smaller than the one indicated by the error, leading to an anticipated stop of the inner loop. This outcome is considered not damaging in a regularizing context. On the contrary, the use of the discrepancy rule resulted in too anticipated stops of the inner loop.

```
function x_reg = IOCG (A, b)
    ã = F a;  b̃ = F* b;  s̃ = ã ⊘ b̃;
    x_0^(0) = A^T b;  d^(0) = ones(N, 1);
    outer_stop_cond = true;
    h = 0;
    while  outer_stop_cond
        y = inner(x_0^(h), d^(h));
        [x_0^(h+1), d^(h+1)] = project(y);
        h = h + 1;
        outer_stop_cond = (min(y) < τ) & (k_in > k_min) & (h ≤ h_max);
    end
    x_reg = x_0^(h+1);
end

function [y, k_in] = inner (x_0, d)
the function is specifically coded for circulant matrices
    k = 0;
    x̃_0 = F* x_0;  r̃_0 = b̃ − n ã ⊙ x̃_0;
    p̃_0 = q̃_0 = F*(d ⊙ F(n ã* ⊙ r̃_0));
    inner_stop_cond = true;
    while  inner_stop_cond
        z̃_k = n ã ⊙ p̃_k;
        α_k = ‖q̃_k‖²/‖z̃_k‖²;
        x̃_{k+1} = x̃_k + α_k p̃_k;
        r̃_{k+1} = r̃_k − α_k z̃_k;
        q̃_{k+1} = F*(d ⊙ F(n ã* ⊙ r̃_{k+1}));
        β_k = ‖q̃_{k+1}‖²/‖q̃_k‖²;
        p̃_{k+1} = q̃_{k+1} + β_k p̃_k;
        k = k + 1;
        inner_stop_cond = isc(x̃_k, r̃_k);
    end
    y = F x̃_{k−1};
    k_in = k − 1;
end

function  χ = isc(x̃_k, r̃_k)
    t = n s̃ ⊙ x̃_k;  δ_k = N ‖r̃_k‖²/(N − Σ_i t_i);
    χ = (δ_k < δ_{k−1}) & (k ≤ k_max);
end
```

---

5 Numerical experiments

The numerical experimentation has been conducted with double precision arithmetic. We consider three reference objects, namely a satellite image [16], the star cluster Messier 5 [26] and the Hoffman phantom [15], widely used in the literature for testing image reconstruction algorithms. The corresponding images (see Fig. 1) are of size $N = 256^2$, normalized between 0 and 1. The percentage of zeros is 90% for the satellite and 75% for the others. In the images the zeros are associated to a white background. We have adopted this version instead of the standard version which associates zeros to black points, to improve readability. Note that the satellite and the phantom have a large solid interconnected zero background, while the background of the cluster is punctuated by many stars and no large zero areas exist. This difference, as we will see, has a certain effect on the reconstructions.

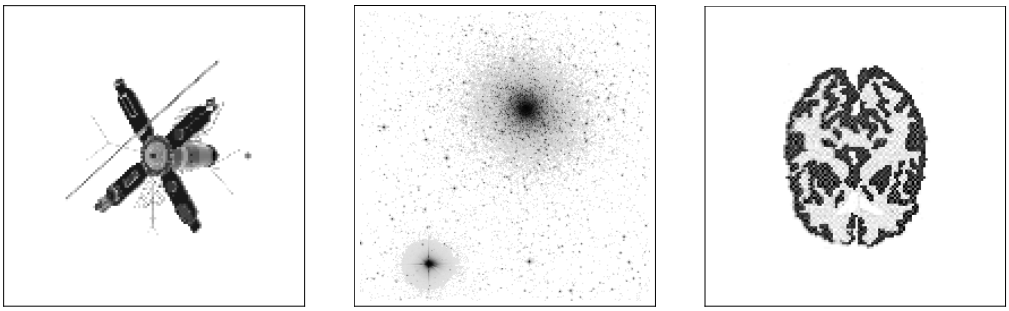

Fig. 1. the satellite image (left), the star cluster (middle) and the Hoffman phantom (right).

The matrix A which performs the blur is a 2-level Toeplitz matrix generated by a positive space invariant bandlimited PSF with a bandwidth $\nu = 8$, normalized in such a way that the sum of the elements is equal to 1. For the astronomical images (satellite and cluster) we consider a motion-type PSF, which simulates the one taken by a ground-based telescope, represented by the following mask:

$$m_{i,j} = \exp(-\alpha(i+j)^2 - \beta(i-j)^2), \quad -\nu \le i,j \le \nu, \quad \alpha = 0.04, \quad \beta = 0.02.$$

For the medical image (the phantom), we consider a Gaussian PSF represented by the following mask:

$$m_{i,j} = \exp(-\alpha\, i^2 - \beta j^2), \quad -\nu \le i,j \le \nu, \quad \alpha = \beta = 0.1.$$

Since the images have sufficiently large zero background along the boundary, the coefficient matrix is approximated by a 2-level circulant matrix.

For each image, 5 different noisy test problems are generated with vector

$\widehat{\boldsymbol{b}} = A\widehat{\boldsymbol{x}}$ by adding both Poisson and Gaussian noise in different proportions. The test problems so obtained have a relative noise level η ranging from 1.3% to 8.1% and are solved by `IOCG` algorithm with a maximum number $k_{max} = 10$ of allowable inner iterations. Experiments with larger values of k_{max} have also been carried out, without significant differences from the point of view of the error, but with a waste of iterations.

The three methods `Alg1`, `PRCG` and `SGP` recalled in Section 3 are used for comparison purposes. The stopping rule is GCV for `IOCG` and `SGP`, and the discrepancy rule with $\vartheta = 1$ for `PRCG` and `Alg1` as suggested in [4,18]. Since `SGP` cannot start with the null vector, the vector $A^T\boldsymbol{b}$ is chosen as initial starting point for all the methods.

Besides the standard measures used to evaluate the effectiveness of a numerical iterative algorithm, i.e. the iteration number and the obtained final error, we borrow from Information Retrieval [22] a tool to measure the ability of the algorithm to correctly detect the zeros of the original image. Let $\boldsymbol{x}_{\text{reg}}$ be the computed solution. Denote "true zeros" the zeros of $\widehat{\boldsymbol{x}}$ and define

- `tp` as the number of zeros in $\boldsymbol{x}_{\text{reg}}$ which are also true zeros (true positives),
- `fp` as the number of zeros in $\boldsymbol{x}_{\text{reg}}$ which are not true zeros (false positives),
- `fn` as the number of nonzeros in $\boldsymbol{x}_{\text{reg}}$ which are true zeros (false negatives),
- `tn` as the number of nonzeros in $\boldsymbol{x}_{\text{reg}}$ which are true nonzeros (true negatives),
- by $\mathtt{p} = \mathtt{tp}/(\mathtt{tp} + \mathtt{fp})$ the *precision*,
- by $\mathtt{r} = \mathtt{tp}/(\mathtt{tp} + \mathtt{fn})$ the *recall*.

The precision can be seen as a qualitative measure of exactness and the recall can be seen as a quantitative measure of completeness. In simple terms, a large `p` indicates that the algorithm returns substantially more positive results than irrelevant ones, while a large `r` means that the algorithm returns most of the relevant results. A measure which combines `p` and `r` is the harmonic mean of the two,

$$\mathtt{F}_1 = \frac{2}{1/\mathtt{p} + 1/\mathtt{r}}$$

called *balanced* $\mathtt{F}_1$ *score*. The largest $\mathtt{F}_1$, the best the performance of the algorithm.

Tables 1, 2 and 3 show the results corresponding to the five different noise levels for the three problems. Column "it" lists the total number of iterations,

column "err" lists the final relative errors $\|\widehat{\boldsymbol{x}} - \boldsymbol{x}_{reg}\|/\|\widehat{\boldsymbol{x}}\|$, column $\mathtt{F}_1$ lists the balanced $\mathtt{F}_1$ score.

	IOCG			Alg1			PRCG			SGP		
η	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$
1.66%	78	0.227	0.88	33	0.244	0.72	67	0.246	0.67	126	0.234	0.56
2.36%	61	0.230	0.85	20	0.255	0.58	31	0.259	0.51	117	0.236	0.77
4.10%	42	0.239	0.78	12	0.277	0.32	11	0.281	0.30	64	0.245	0.58
5.73%	37	0.247	0.76	11	0.280	0.33	12	0.284	0.27	47	0.254	0.60
7.71%	32	0.253	0.72	6	0.308	0.20	6	0.311	0.17	36	0.260	0.36

Table 1
Results corresponding to five levels of noise for the satellite image.

	IOCG			Alg1			PRCG			SGP		
η	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$
1.31%	100	0.430	0.66	51	0.438	0.51	96	0.438	0.53	47	0.431	0.08
2.40%	42	0.439	0.60	28	0.446	0.43	34	0.446	0.44	41	0.433	0.11
3.22%	39	0.441	0.60	19	0.449	0.43	23	0.449	0.39	39	0.435	0.04
5.26%	21	0.448	0.48	12	0.455	0.23	10	0.456	0.29	34	0.441	0.01
8.08%	19	0.452	0.48	8	0.460	0.24	8	0.460	0.23	16	0.446	0.01

Table 2
Results corresponding to five levels of noise for the star cluster.

	IOCG			Alg1			PRCG			SGP		
η	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$	it	err	$\mathtt{F}_1$
1.69%	57	0.295	0.87	28	0.309	0.64	36	0.310	0.65	99	0.299	0.84
2.42%	42	0.300	0.83	20	0.319	0.53	19	0.320	0.50	68	0.304	0.79
4.22%	37	0.307	0.79	9	0.335	0.36	9	0.336	0.34	48	0.313	0.64
5.87%	21	0.317	0.68	10	0.338	0.25	10	0.339	0.28	28	0.322	0.41
8.10%	19	0.322	0.65	6	0.360	0.22	6	0.361	0.19	26	0.329	0.64

Table 3
Results corresponding to five levels of noise for the Hoffman phantom.

From the tables we note that from the error point of view, IOCG outperforms all the other methods with the satellite and the phantom images, and only Alg1 and PRCG with the cluster image. In the latter case SGP does better. This means that IOCG is particularly suitable for images having a large solid zero background. The favorable result of IOCG on the other inner-outer methods can in part be due to the fact that inner IOCG employs GCV as a stopping condition, while the other methods employ the discrepancy rule for both inner and outer steps, as clearly shown by the too small number of iterations, especially for large noises η.

Figure 2 shows a typical plot of the relative error histories obtained by applying `IOCG` (solid line), `Alg1` (dotted line) and `SGP` (dashed line). It is limited to the first 60 iterates for the Hoffman phantom with $\eta = 1.69\%$. The errors of `PRCG` are not shown because nearly overlapped by the errors of `Alg1`.

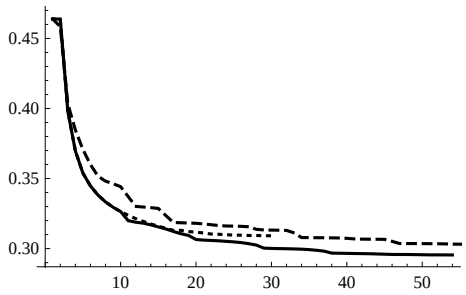

Fig. 2. relative error histories of `IOCG` (solid lines), `Alg1` (dotted line) and `SGP` (dashed lines) for the Hoffman phantom with $\eta = 1.69\%$.

We must note that improvements of the error of such small entities like the ones appearing in the tables produce only slight improvements in the quality of the reconstructed images. Figures 3, 4 and 5 show enlarged central parts of the images reconstructed using `IOCG`, `Alg1` and `SGP`, compared to the blurred images (the images produces by `PRCG` do not differ from the ones produced by `Alg1`).

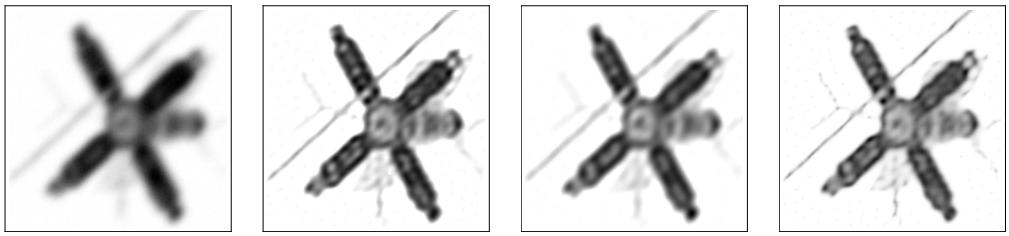

Fig. 3. blurred image (on the left) and reconstructed images by `IOCG`, by `Alg1` and by `SGP` (from left to right) for the satellite with $\eta = 2.36\%$.

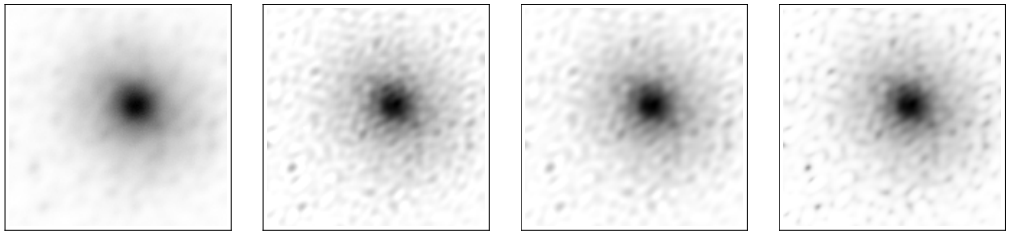

Fig. 4. blurred image (on the left) and reconstructed images by `IOCG`, by `Alg1` and by `SGP` (from left to right) for the star cluster with $\eta = 2.40\%$.

The parameter $\mathtt{F}_1$, which decreases with the increasing of the noise, shows that larger active sets imposed on the reconstructed solution effectively correspond

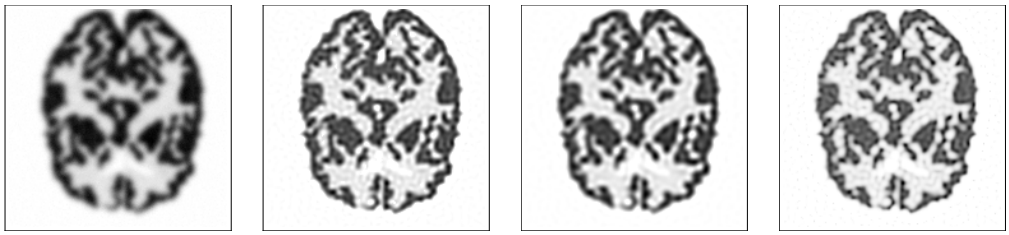

Fig. 5. blurred image (on the left) and reconstructed images by `IOCG`, by `Alg1` and by `SGP` (from left to right) for the Hoffman phantom with $\eta = 2.42\%$.

to zero components of the original solution. For example, the $\mathtt{F}_1 = 0.88$ of `IOCG` for the satellite with $\eta = 1.66\%$ is given by

$$\mathtt{tp} = 46785, \quad \mathtt{fp} = 44, \quad \mathtt{fn} = 12119, \quad \mathtt{tn} = 6588,$$

while the corresponding score of 0.72 for `Alg1` is given by

$$\mathtt{tp} = 32834, \quad \mathtt{fp} = 10, \quad \mathtt{fn} = 26070, \quad \mathtt{tn} = 6622,$$

showing that `Alg1` produces less true results and more false results than `IOCG`.

The very small $\mathtt{F}_1$ score of `SGP` applied to the star cluster depends on the fact that `SGP` does not apply the projection to the computed solutions but to the directions, and this produces solutions with a reduced number of zeros, especially when there is not a solid background in the original image. Anyway, this does not prevent an overall acceptable approximation.

6 Conclusions

The inner-outer method `IOCG` has been proposed for the reconstruction of images having large nearly zero backgrounds. The method, tested on ill-posed problems, appears to be more reliable than three considered nonnegatively constrained regularization methods from the point of view of the relative error and the number of the reconstructed zeros, even at the cost of a larger number of iterations.

References

[1] J. M. Bardsley and J. G. Nagy, *Covariance-preconditioned iterative methods for nonnegatively constrained astronomical imaging*, Siam J. Matrix Anal. Appl. **27** (2006), pp. 1184–1197.

[2] M. Bertero and P. Boccacci, *Introduction to Inverse Problems in Imaging*, Institute of Physics Publishing, Bristol, 1998.

[3] S. Bonettini, R. Zanella and L. Zanni, *A scaled gradient projection method for constrained image deblurring*, Inverse Problems **25** (2009), 015002.

[4] D. Calvetti, G. Landi, L. Reichel and F. Sgallari, *Nonnegativity and iterative methods for ill-posed problems*, Inverse Problems **20** (2004), pp. 1747-1758.

[5] M. E. Daube-Witherspoon and G. Muehllehner, *An iterative image space reconstruction algorithm suitable for volume ECT*, IEEE Trans. Med. Imgng **5** (1986), pp. 61–66.

[6] P. Favati, G. Lotti, O. Menchi and F. Romani, *Performance analysis of maximum likelihood methods for regularization problems with nonnegativity constraints*, Inverse Problem **26** (2010), 085013.

[7] P. Favati, G. Lotti, O. Menchi and F. Romani, *Stopping rules for iterative methods in nonnegatively constrained deconvolution*, Applied Numerical Mathematics **75** (2014), pp. 154–166.
http://dx.doi.org/10.1016/j.apnum.2013.07.006

[8] P. Favati, G. Lotti, O. Menchi and F. Romani, *Generalized Cross-Validation applied to Conjugate Gradient for discrete ill-posed problems*, Applied Mathematics and Computation, **243** (2014), pp. 258–268.

[9] P. Favati, G. Lotti, O. Menchi and F. Romani, "Regularization by conjugate gradient of nonnegatively constrained least squares", Tech. Report IIT-CNR TR-08/2014.

[10] A. Foi, M. Trimeche, V. Katkovnik and E. Egiazarian, *Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data*, IEEE Transactions (2007), pp. 1–18.

[11] G. H. Golub, M. Heath and G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, **21** (1979), pp. 215-223.

[12] G. H. Golub and C. F. Van Loan, *Matrix Computation*, second edition, John Hopkins University Press, Baltimore, MD, 1989.

[13] M. Hanke, *Conjugate gradient type methods for ill-posed problems*, Pitman Research Notes in Mathematics, Longman, Harlow, 1995.

[14] P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM Monographs on Mathematical Modeling and Computation, Philadelphia, 1998.

[15] E. J. Hoffman, P. D. Cutler, W. M. Digby and J. C. Mazziotta, *3-D phantom to simulate cerebral blood flow and metabolic images for PET*, IEEE Trans. Nucl. Sci. **37** (1990), pp. 616–620.

[16] K. P. Lee, J. G. Nagy and L. Perrone, *Iterative methods for image restoration: a Matlab object oriented approach*, 2002. Available from http://www.mathcs.emory.edu/∼nagy/RestoreTools.

[17] L. B. Lucy, *An Iteration Technique for the Rectification of Observed Distributions*, Astron. Journal **79** (1990), pp. 745–754.

[18] S Morigi, L. Reichel, F. Sgallari and F. Zama, *An iterative method for linear discrete ill-posed problems with box constriants*, J. Comput. Appl. Math. bf 198 (2007), pp. 505-520.

[19] J. Nagy and Z. Strakos, *Enforcing nonnegativity in image reconstruction algorithms*, Math. Modeling, Estimation, and Imaging, D. C. Wilson et al., Eds., 4121 (2000), pp. 182–190.

[20] R. Plato, *Optimal algorithms for linear ill-posed problems yield regularization methods*, Num. Funct. Anal. and Optimiz., **11** (1990), pp. 111–118.

[21] W. Richardson, *Bayesian-based iterative method of image restoration*, Journal of the Optical Society of America **62** (1972), pp. 55–59.

[22] C. J. van Rijsbergen, *Information Retrieval*, Butterworth, London, BG, 1979.

[23] L. A. Shepp and Y. Vardi, *Maximum likelihood reconstruction for emission tomography*, IEEE Trans. Med. Imag., MI-1 (1982), pp. 113–122.

[24] C. Vogel, *Computational Methods for Inverse Problems*, SIAM Frontier in Applied Mathematics, Philadelphia, 2002.

[25] G. Wahba, *Practical approximate solutions to linear operator equations when the data are noisy*, SIAM J. Numer. Anal., **14** (1977), pp. 651–667.

[26] http://www.skycrumbles.net/img_info/m5/