New Concepts in Imaging: Optical and Statistical Models
D. Mary, C. Theys and C. Aime (eds)
EAS Publications Series, **59** *(2013) 325–356*

SCALED GRADIENT PROJECTION METHODS FOR ASTRONOMICAL IMAGING

M. Bertero[1], P. Boccacci[1], M. Prato[2] and L. Zanni[2]

Abstract. We describe recently proposed algorithms, denoted *scaled gradient projection* (SGP) methods, which provide efficient and accurate reconstructions of astronomical images. We restrict the presentation to the case of data affected by Poisson noise and of nonnegative solutions; both maximum likelihood and Bayesian approaches are considered. Numerical results are presented for discussing the practical behaviour of the SGP methods.

1 Introduction

Image deconvolution is an important tool for reducing the effects of noise and blurring in astronomical imaging. In this paper we assume that blurring is described by a space invariant *point spread function* (PSF) and that a model of the PSF is available, accounting for both telescope diffraction and adaptive optics (AO) correction of the atmospheric blur. Therefore we do not consider topics such as space-variant deblurring or blind deconvolution.

Since the deconvolution problem is ill-posed, it should be formulated by using all the information available on the image formation process: not only the PSF is required but also a knowledge of the statistical properties of the noise affecting the data. These properties are used, for instance, for reformulating deconvolution as a *maximum likelihood* (ML) problem, which is also ill-posed in many instances (even if, presumably, with a lower degree of ill-posedness). Then prior information on the unknown astronomical target is required and this, if available, can be taken into account by extending the ML approach to a *Bayesian* approach. In both cases one reformulates deconvolution as a discrete variational problem and therefore the use of methods derived from numerical optimization becomes essential.

As concerns noise modeling, a crucial point is that astronomical images are typically detected by charged coupled device (CCD) cameras so that one can use,

[1] DIBRIS, Università di Genova, via Dodecaneso 45, 16146 Genova, Italy
[2] DISFIM, Università di Modena e Reggio Emilia, via Campi 213/b, 41125 Modena, Italy

DOI: 10.1051/eas/1359015

for instance, the accurate model described by Snyder *et al.* (1993). According to this model, if we denote by y_i the value of the image y detected at pixel i, then (after correction for flat field, bad pixels etc.) y_i is given by

$$y_i = y_i^{(\mathrm{obj})} + y_i^{(\mathrm{back})} + y_i^{(\mathrm{ron})}, \tag{1.1}$$

where $y_i^{(\mathrm{obj})}$ is the number of photoelectrons due to radiation from the object, $y_i^{(\mathrm{back})}$ is the number of photoelectrons due to internal and external background, dark current, etc., and $y_i^{(\mathrm{ron})}$ is the contribution of the *read-out noise* (RON) due to the amplifier. The first two terms are realizations of Poisson random variables (r.v.) while the third is a realization of an additive Gaussian r.v.. Therefore the noise affecting the data is a mixture of Poisson (due to photon counting) and additive Gaussian noise, due to RON. However, a refined model taking into account this particular structure of the noise does not provide significant improvement with respect to a simplified model also proposed by Snyder *et al.* (1993) (for a comparison see, for instance, Benvenuto *et al.* 2008, 2012). Indeed, Snyder *et al.* propose that, after the substitution $y_i \to y_i + \sigma^2$, where σ^2 is the variance of the RON, the RON can be treated as the realization of a Poisson r.v. with mean and variance being the same as σ^2. In this paper we use this approximation, which is quite accurate in the case of *near infrared* (NIR) observations, characterized by a large background emission. In conclusion we assume that the data y_i, shifted by σ^2, are realizations of suitable Poisson r.v.s.

In the framework of this model several iterative methods have been proposed for solving the ML or the Bayes problem. These methods are, in general, easy to implement but very slow: they require a large number of iterations, so that the computational cost can become excessive for the present and future large telescopes, able to acquire images of several mega-pixels so that the problem of image deconvolution in Astronomy becomes a large scale one.

An interesting property of some of the proposed algorithms is that they are first-order optimization methods using as a descent direction a suitable (diagonal) scaling of the negative gradient of the objective function. As a consequence, using these scalings, it is possible to apply a recently proposed approach denoted as *scaled gradient projection* (SGP) method and described in its general form by Bonettini *et al.* (2009). As shown by several numerical experiments this approach can provide a considerable speed-up of the standard algorithms.

In this paper SGP is not only considered for single-image deconvolution, the typical problem arising in the improvement of images provided by telescopes consisting of a monolithic mirror, but also for multiple-image deconvolution, a problem arising when different images of the same astronomical target are available. A significant application of this approach is the deconvolution of the images of the future Fizeau interferometer of the Large Binocular Telescope (LBT) called LINC-NIRVANA (Herbst *et al.* 2003).

LBT (`http://www.lbto.org`) is the world's largest optical and infrared telescope since it consists of two 8.4 m primary mirrors with the total light-gathering power of a single 11.8 m telescope. The two mirrors have an elevation over an

azimuth mounting and the elevation optical support structure moves on two large C-shaped rings (see Fig. 1). They are mounted with a 14.4 m centre separation, hence with an edge-to-edge distance of 22.8 m. This particular structure makes possible Fizeau interferometry, with a maximum baseline of 22.8 m, corresponding to a theoretical resolution of a 22.8 m mirror in the direction of the line joining the two centres.

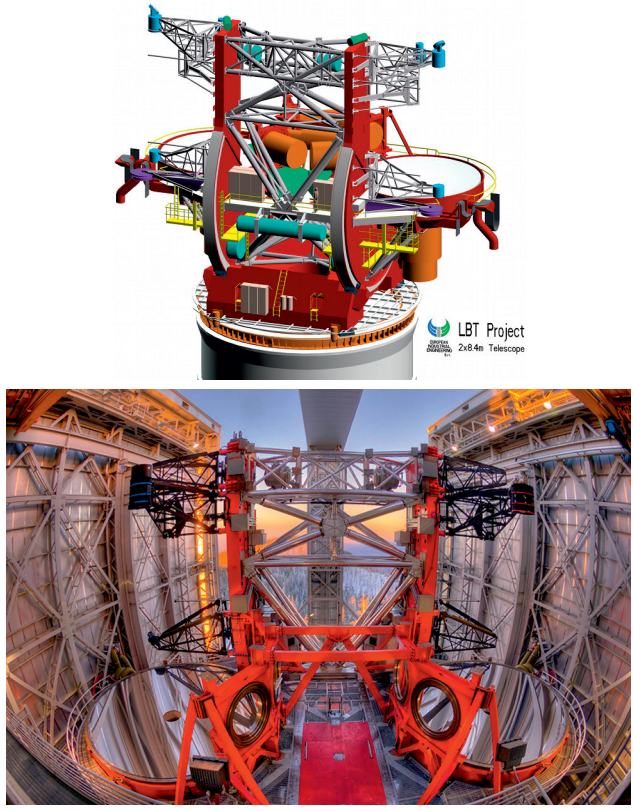

Fig. 1. A design view of LBT (*upper panel*), and a fish-eye image of the opposite side of LBT inside the enclosure (*lower panel*), as it appears to the visitors of the observatory (photo courtesy of W. Ruyopakam and the Large Binocular Telescope Observatory).

LINC-NIRVANA (LN for short) will operate as a true imager. Indeed, in the Fizeau mode, the two beams from the primary mirrors are combined in a common focal plane (not in the pupil plane as with essentially all the existing interferometers). LN is in an advanced realization phase by a consortium of German and Italian institutions, leaded by the Max Planck Institute for Astronomy in Heidelberg (`http://www.mpia.de/LINC/`). When completed, the instrument will be mounted in the centre of the platform of LBT (clearly visible in the lower panel of Fig. 1). It will be fully commissioned and available for scientific studies in 2014.

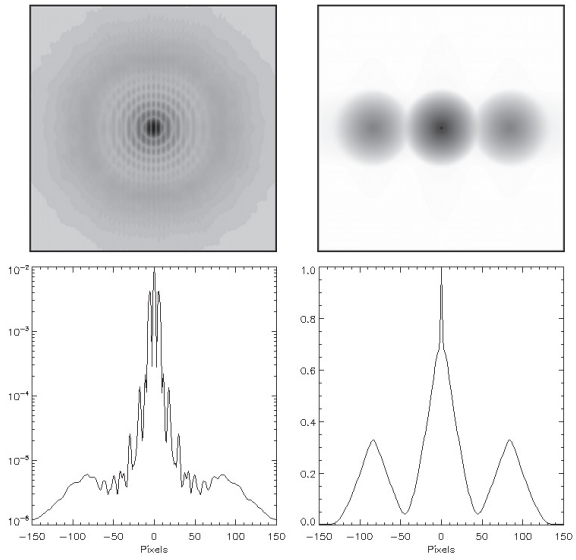

Fig. 2. Simulated PSF of LINC-NIRVANA with SR = 70% (*upper-left panel*), and the corresponding MTF (*upper-right panel*), both represented with reversed gray scale. The fringes are orthogonal to the baseline. In the lower panels we show the cut of the PSF along the baseline (*left*) and the cut of the MTF along the same direction (*right*).

In Figure 2 we show a simulated point spread function (PSF) with SR = 70%, together with the corresponding modular transfer function (MTF), *i.e.* the modulus of the Fourier transform of the PSF. This PSF, as well as others used in this paper, has been obtained with the code LOST (Arcidiacono *et al.* 2004). It is monochromatic ($\lambda = 2.2$ μm, *i.e.* K band), and, as clearly appears from this figure, it is the PSF of a 8.4 m telescope modulated by the interferometric fringes; accordingly the central disc of the MTF corresponds to the band of a 8.4 m mirror while the two side disks are replicas, due to interferometry, with a weaker intensity than the central one. These disks contain the precious additional information on the target due to interferometry.

As follows from this analysis, LN images will be characterized by an anisotropic resolution: that of a 22.8 m telescope in the direction of the baseline, and that of a 8.4 m in the orthogonal direction. Therefore, in order to get the maximum resolution in all directions, it will be necessary to acquire different images of the same astronomical target with different orientations of the baseline and to combine these images into a unique high-resolution image by means of suitable image reconstruction methods. In other words LN will routinely require multiple-image deconvolution.

The paper is organized as follows. In Section 2 we outline the mathematical model based on the approximation of the RON mentioned above and we describe the main algorithms introduced for solving the ML and the Bayesian problems both

for single and multiple-image deconvolution. Moreover we recall an approach, proposed in Bertero & Boccacci (2005), for boundary effect correction. In Section 3 we describe the algorithm SGP in the particular case of the nonnegativity constraint and the optimization of its parameters. In Section 4 we demonstrate its efficiency by several numerical experiments and finally in Section 5 we derive some conclusions.

2 Mathematical modeling

As outlined in the Introduction we assume that the value y_i of an astronomical image y detected at pixel i is the realization of a Poisson r.v. Y_i with unknown expected value λ_i. A further assumption is that the r.v.s. associated with different pixels are statistically independent. As a consequence their joint probability distribution is given by

$$P_Y(y|\lambda) = \prod_{i \in S} \frac{e^{-\lambda_i} \lambda_i^{y_i}}{y_i!}, \tag{2.1}$$

the data being assumed to be integer numbers and S being the set of the index values.

In the case of a linear model for image formation, with the imaging system described by a space-invariant PSF, the unknown expected value is given by

$$\lambda_i = (Hx)_i + b_i, \quad Hx = K * x, \tag{2.2}$$

where: x is the unknown astronomical target; b the background emission, including the σ^2 term due to the RON; H the imaging matrix and K the PSF of the system satisfying the conditions

$$K_i \ \geq \ 0, \quad \sum_{i \in S} K_i = 1. \tag{2.3}$$

Assuming that b and K are known, the image restoration problem requires the development of methods for providing an estimate of x, given y.

2.1 Maximum likelihood approach

In the ML approach, given the detected image y, as well as b and K, one introduces the likelihood function, which is the function of x defined by

$$\mathcal{L}_y(x) = P_Y(y|Hx + b) \tag{2.4}$$

and obtained by inserting the image y and the model (2.2) in Equation (2.1). Then, a ML estimate of the unknown object is any image x^* which maximizes the likelihood function. However, since the likelihood is the product of a large number of functions, it is more convenient to take the negative logarithm of the likelihood

and minimize the resulting function. It is easy to see that, by rearranging terms independent of x, the negative logarithm of $\mathcal{L}_y(x)$ is given by

$$f_0(x;y) = \sum_{i \in S} \left\{ y_i \ln \frac{y_i}{(Hx+b)_i} + (Hx+b)_i - y_i \right\}, \tag{2.5}$$

which is the so-called generalized *Kullback-Leibler* (KL) *divergence* of the computed data $Hx + b$ from the detected data y. This function is nonnegative and is zero iff $Hx + b = y$; it is also convex and coercive, *i.e.* $f_0(x;y) \to +\infty$ if $||x||_2 \to +\infty$. The KL-divergence is not a metric distance, because it is not symmetric in the two terms and does not satisfy the triangle inequality. However it can be taken as a measure of the discrepancy between $Hx+b$ and y; it will be called the *data fidelity function*. The properties of $f_0(x;y)$ imply the existence of global minima of this function on the nonnegative orthant and therefore the existence of nonnegative ML estimates of the unknown. If all data are strictly positive and the imaging matrix is nonsingular, then $f_0(x;y)$ is strictly convex, a sufficient condition for the uniqueness of the solution.

As shown in Barrett & Meyers (2003), the nonnegative minimizers of $f_0(x;y)$ are sparse objects, *i.e.* they consist of bright spots over a black background (sometimes are called *star-night solutions*). Therefore they can be reliable solutions in the case of simple astronomical objects, such as binaries or open star clusters, but they are not in the case of more complex objects, such as nebulae, galaxies etc..

The standard algorithm for the minimization of $f_0(x;y)$ is the so-called *Richardson-Lucy* (RL) algorithm (Richardson 1972; Lucy 1974), defined by

$$x^{(k+1)} = x^{(k)} \cdot H^T \frac{y}{Hx^{(k)} + b}, \tag{2.6}$$

where the $\cdot$ denotes Hadamard product of two vectors and similarly the fraction symbol indicates component-wise division of two vectors. In the case $b = 0$ convergence of the iteration to the minimizers of $f_0(x;y)$ has been proved, but it is important to remark that the algorithm has also well-known "regularization" properties: in the case of complex objects sensible solutions can be obtained by a suitable early stopping of the iterations, even if this approach may not provide satisfactory results in some specific cases, for instance in the case of objects with sharp structures. Then a more refined regularization can be obtained by the use of prior information on the solution in a Bayesian framework (see, the next subsection).

An extension of the previous approach is required when different images of the same object are available. This problem, as discussed in the Introduction, is fundamental for the future Fizeau interferometer of LBT or for the "co-adding" method of images with different PSFs proposed by Lucy & Hook (1992).

Let p be the number of detected images $y^{(j)}$, j=1,..,p, with corresponding PSFs $K^{(j)}$, all normalized to unit volume, $H^{(j)}x = K^{(j)} * x$, and backgrounds $b^{(j)}$ (including the term σ^2 due to RON). It is quite natural to assume that the p images are statistically independent, so that the likelihood of the problem is the

product of the likelihoods associated to the different images. If we assume again Poisson statistics, and we take the negative logarithm of the likelihood, then the ML estimates are the minimizers of a data-fidelity function which is the sum of KL divergences, one for each image, *i.e.*

$$f_0(x;y) = \sum_{j=1}^{p} \sum_{i \in S} \left\{ y_i^{(j)} \ln \frac{y_i^{(j)}}{(H^{(j)}x + b^{(j)})_i} + (H^{(j)}x + b^{(j)})_i - y_i^{(j)} \right\}. \tag{2.7}$$

If we apply the standard expectation maximization method (Shepp & Vardi 1982) to this problem, we obtain the iterative algorithm

$$x^{(k+1)} = \frac{1}{p} x^{(k)} \cdot \sum_{j=1}^{p} (H^{(j)})^T \frac{y^{(j)}}{H^{(j)}x^{(k)} + b^{(j)}}, \tag{2.8}$$

which we call the *multiple-image* RL method (multiple RL, for short).

For the reconstruction of LN images an acceleration of this algorithm is proposed in Bertero & Boccacci (2000) by exploiting an analogy between the images of the interferometer and the projections in tomography. In this approach called OSEM (ordered subset expectation maximization; Hudson & Larkin 1994), the sum over the p images in Equation (2.8) is replaced by a cycle over the same images. To avoid oscillations of the reconstructions within the cycle, a preliminary step is the normalization of the different images to the same flux, if different integration times are used in the acquisition process. The method OSEM is summarized in Algorithm 1.

Algorithm 1 Ordered subset expectation maximization (OSEM) method

Choose the starting point $x^{(0)} > 0$.

For $k = 0, 1, 2, ...$ do the following steps:

Step 1. Set $h^{(0)} = x^{(k)}$;

Step 2. For $j = 1, ..., p$ compute

$$h^{(j)} = h^{(j-1)} \cdot (H^{(j)})^T \frac{y^{(j)}}{H^{(j)}h^{(j-1)} + b^{(j)}}; \tag{2.9}$$

Step 3. Set $x^{(k+1)} = h^{(p)}$.

End

As follows from practice and theoretical remarks, this approach reduces the number of iterations by a factor p. However, the computational cost of one multiple RL iteration is lower than that of one OSEM iteration: we need $3p + 1$ FFTs in the first case and $4p$ FFTs in the second. In conclusion, the increase in efficiency provided by OSEM is roughly given by $(3p + 1)/4$. When $p = 3$ (the number of images provided by the interferometer will presumably be small), the efficiency

is higher by a factor of 2.5, and a factor of 4.7 when $p = 6$. These results must be taken into account when evaluating the efficiency of SGP with respect to that of multiple RL. We can add that the convergence of SGP is proved while that of OSEM is not, even if it has always been verified in our numerical experiments.

2.2 Bayesian approach

As already remarked, the regularization of the ML estimates obtained by an early stopping of the previous algorithms may not be satisfactory in some cases. A more general kind of regularization can be obtained with the so-called Bayesian approach. In this approach one assumes that the unknown object is also a realization of a suitable r.v. X whose probability distribution expresses information available on its properties, such as smoothness, sharp details etc..

A frequently used probability distribution has the following form, which is typical in statistical mechanics

$$P_X(x) = \frac{1}{Z} e^{-\beta\ f_1(x)}, \tag{2.10}$$

where Z (also called the partition function) is a normalization constant, β is a hyper-parameter, playing the role of a regularization parameter in our application, and $f_1(x)$ is a potential function characterizing the known properties of the unknown object, called in the following *regularization function* or also *regularizer*. $P_X(x)$ is called the *prior*.

If the probability distribution $P_Y(y|Hx + b)$, obtained by combining Equations (2.1) and (2.2), is interpreted as the conditional probability of Y for a given value of X, then, from Bayes formulas we obtain that the conditional probability of X for a given value of Y is given by

$$P_X(x|y) = \frac{P_Y(y|Hx + b)P_X(x)}{P_Y(y)}, \tag{2.11}$$

where $P_Y(y)$ is the marginal probability distribution of Y.

If in this equation we insert the detected image y, we obtain a function of x which is called the *posterior probability* of x and is essentially the product of the likelihood and the prior (the value of the marginal distribution of Y computed in y is a constant which can be neglected). The maximizers of this function are the *maximum a posteriori* (MAP) estimates of the unknown object. By taking again the negative log of this function we find that they are the nonnegative minimizers of the function

$$f_\beta(x; y) = f_0(x; y) + \beta f_1(x), \tag{2.12}$$

where the second term is the negative log of the prior. If the function $f_1(x)$ is convex and nonnegative, then $f_\beta(x; y)$ is also convex and nonnegative; moreover it is also coercive, thanks to the coercivity of $f_0(x; y)$, so that MAP estimates of the unknown object exist. Given the regularizer, a crucial point in this approach is the choice of the regularization parameter β. This point will be briefly discussed in the following.

For our purposes an interesting algorithm for the minimization of $f_\beta(x;y)$ is the so-called *split-gradient method* (SGM) proposed by Lantéri *et al.* (2002), which consists in a simple modification of the RL algorithm. If $f_1(x)$ is differentiable and $U_1(x), V_1(x)$ is a pair of nonnegative functions such that

$$-\nabla_x f_1(x) = U_1(x) - V_1(x), \tag{2.13}$$

then the algorithm is as follows

$$x^{(k+1)} = \frac{x^{(k)}}{\hat{1} + \beta\ V_1(x^{(k)})} \cdot \left\{ H^T \frac{y}{Hx^{(k)} + b} + \beta\ U_1(x^{(k)}) \right\}, \tag{2.14}$$

where $\hat{1} = (1, \ldots, 1)^T$. The choice of the pair $U_1(x), V_1(x)$ is not unique but, for each one of the standard regularizers, one can find a quite natural choice (Lantéri *et al.* 2002). As concerns the extension to the case of multiple image deconvolution (Bertero *et al.* 2011), the updating rule of SGM becomes

$$x^{(k+1)} = \frac{x^{(k)}}{p\hat{1} + \beta V_1(x^{(k)})} \cdot \left\{ \sum_{j=1}^{p} (H^{(j)})^T \frac{y^{(j)}}{H^{(j)}x^{(k)} + b^{(j)}} + \beta U_1(x^{(k)}) \right\}, \tag{2.15}$$

while the OSEM algorithm, with regularization, is given by Algorithm 1 where Equation (2.9) is replaced by

$$h^{(j)} = \frac{h^{(j-1)}}{\hat{1} + \frac{\beta}{p} V_1(h^{(j-1)})} \cdot \left\{ (H^{(j)})^T \frac{y^{(j)}}{H^{(j)}h^{(j-1)} + b^{(j)}} + \frac{\beta}{p} U_1(h^{(j-1)}) \right\}. \tag{2.16}$$

2.3 Boundary effect corrections

If the target x is not completely contained in the image domain, then the previous deconvolution methods produce annoying boundary artifacts. It is not the purpose of this paper to discuss the different methods for solving this problem. We focus on an approach proposed in Bertero & Boccacci (2005) for single-image and in Anconelli *et al.* (2006) for multiple-image deconvolution. Here we present the approach in the case of multiple images (single image corresponds to $p = 1$).

The idea is to reconstruct the object x over a domain broader than that of the detected images and to merge, by zero padding, the arrays of the images and the object into arrays of dimensions that enable their Fourier transform to be computed effectively by means of FFT. We denote by $\bar{S}$ the set of values of the index labeling the pixels of the broader arrays containing S, and by R that of the object array contributing to S, so that $S \subset R \subset \bar{S}$. It is obvious that also the PSFs must be defined over $\bar{S}$ and that this can be done in different ways, depending on the specific problem one is considering. We point out that they must be normalized to unit volume over $\bar{S}$. We also note that R corresponds to the part of the object contributing to the detected images and that it depends on the extent of the PSFs. The reconstruction of x outside S is unreliable in most

cases, but its reconstruction inside S is practically free of boundary artifacts, as shown in the papers cited above and in the experiments of Section 4.

In order to estimate the reconstruction domain R we can proceed as follows. Let M_S be the characteristic function (mask) of S in $\bar{S}$, *i.e.* the array which is 1 inside S and 0 outside; moreover, let us introduce the following arrays, defined over $\bar{S}$, which appear in the computation of the gradient of $f_0(x;y)$ as defined below

$$\gamma^{(j)} = K_-^{(j)} * M_S \ , \ \gamma = \sum_{j=1}^{p} \gamma^{(j)}, \tag{2.17}$$

where $(K_-^{(j)})_i = (K^{(j)})_{-i}$. These arrays are essentially the images of M_S in $\bar{S}$ and are computable by FFT. Their extent outside S (they can be either very small or zero in pixels of $\bar{S}$ outside S) depends on the extent of the PSF and therefore they can be used for defining the reconstruction domain R. Given a thresholding value ϵ, we define R as follows

$$R = \{l \in \bar{S} \mid \gamma_l^{(j)} \geq \epsilon \ ; j = 1, .., p\}; \tag{2.18}$$

Next, if M_R is the characteristic function of R, we introduce the following matrices $H^{(j)}$ and $(H^{(j)})^T$

$$H^{(j)}x = M_S \cdot K^{(j)} * (M_R \cdot x), \tag{2.19}$$

$$(H^{(j)})^T h = M_R \cdot K_-^{(j)} * (M_S \cdot h), \tag{2.20}$$

where, in the second equation, h denotes a generic array defined over $\bar{S}$. Again, both matrices can be computed by means of FFT. We point out that, in the case of a regularization function containing the discrete gradient of x, it could be convenient to slightly modify the definition of M_R: not use exactly the characteristic function of R, but an array which is 1 over R and tends smoothly to 0 outside R (obtained, for instance, by convolving the characteristic function of R with a suitable Gaussian). In this way one can avoid discontinuities at the boundary of R in $\bar{S}$.

With the previous definitions, the data fidelity function is given again by Equation (2.7), with S replaced by $\bar{S}$ and the matrices $H^{(j)}$ defined as in the previous equation. Then the multiple RL algorithm, with regularization and boundary effect correction, is given by

$$x^{(k+1)} = \frac{M_R \cdot x^{(k)}}{\gamma + \beta V_1(x^{(k)})} \cdot \left\{ \sum_{j=1}^{p} (H^{(j)})^T \frac{y^{(j)}}{H^{(j)}x^{(k)} + b^{(j)}} + \beta U_1(x^{(k)}) \right\}, \tag{2.21}$$

the quotient being zero in the pixels outside R. Similarly, the OSEM algorithm, with regularization and boundary effect correction, is given by Algorithm 1 where Equation (2.9) is replaced by

$$h^{(j)} = \frac{M_R \cdot h^{(j-1)}}{\gamma^{(j)} + \frac{\beta}{p} V_1(h^{(j-1)})} \cdot \left\{ (H^{(j)})^T \frac{y^{(j)}}{H^{(j)}h^{(j-1)} + b^{(j)}} + \frac{\beta}{p} U_1(h^{(j-1)}) \right\}. \tag{2.22}$$

We stress again that the convergence of OSEM is not proved in the case of noisy data but that it has been always verified numerically in our applications to astronomical imaging.

3 The scaled gradient projection method

Let us consider, for generality, the case of multiple images with boundary effect correction and regularization. It is easy to verify that the gradient of $f_\beta(x;y)$, with x restricted to R, is given by

$$\nabla_x f_\beta(x;y) = M_R \cdot \left(\gamma - \sum_{j=1}^{p} (H^{(j)})^T \frac{y^{(j)}}{H^{(j)}x + b^{(j)}} \right) + \beta \nabla f_1(x), \tag{3.1}$$

where the definitions and notations introduced in the previous sections are used. If x is an admissible image, $x \geq 0$, then it is also easy to verify that, for each $\alpha \in (0,1]$ the image

$$x_\alpha = x - \alpha \frac{x}{\gamma + \beta\ V_1(x)} \nabla_x f_\beta(x;y), \tag{3.2}$$

where $V_1(x)$ is the array related to the gradient of $f_1(x)$ (see Eq. (2.13)), is also an admissible image. If we do the substitutions $x_\alpha = x^{(k+1)}$, $x = x^{(k)}$ and $\alpha = 1$, we re-obtain the algorithm of Equation (2.21).

Since all the algorithms in the previous section can be obtained as particular cases of this one, we can conclude that all these algorithms are scaled gradient method, with a descent direction which is also feasible just thanks to the scaling of the gradient which has been introduced. This property may suggest that all the scalings previously considered may be very useful for designing efficient first order methods and this is just what is obtained thanks to the SGP method proposed in Bonettini *et al.* (2009).

3.1 The algorithm

In many astronomical applications both ML and Bayes problems are particular cases of the following general convex optimization problem

$$\min \quad f(x), \quad \text{sub.to} \quad x \geq 0, \tag{3.3}$$

where f is a continuously differentiable, nonnegative, convex and coercive function. In the following we denote as P_+ the projection onto the nonnegative orthant, *i.e.* the operator setting to zero the negative component of a vector. Moreover, we introduce the set $\mathcal{D}$ of the diagonal positive definite matrices, whose diagonal elements have values between L_1 and L_2, for given thresholds $0 < L_1 < L_2$. Then the general SGP can be stated as in Algorithm 2.

In practice, at iteration k, given the step-length α_k and the scaling matrix $D_k \in \mathcal{D}$, a descent direction $d^{(k)}$ is obtained as difference between the projection of the

Algorithm 2 Scaled gradient projection (SGP) method

Choose the starting point $x^{(0)} \geq 0$ and set the parameters $\eta, \theta \in (0,1)$, $0 < \alpha_{min} < \alpha_{max}$.

FOR $k = 0, 1, 2, \ldots$ DO THE FOLLOWING STEPS:

STEP 1. Choose the parameter $\alpha_k \in [\alpha_{min}, \alpha_{max}]$ and the scaling matrix $D_k \in \mathcal{D}$;

STEP 2. Projection:
$$z^{(k)} = P_+(x^{(k)} - \alpha_k D_k \nabla f(x^{(k)}));$$

STEP 3. Descent direction: $d^{(k)} = z^{(k)} - x^{(k)}$;

STEP 4. Set $\lambda_k = 1$;

STEP 5. Backtracking loop:
let $f_{new} = f(x^{(k)} + \lambda_k d^{(k)})$;
IF
$$f_{new} \leq f(x^{(k)}) + \eta \lambda_k \nabla f(x^{(k)})^T d^{(k)}$$
THEN
go to step 6;
ELSE
set $\lambda_k = \theta \lambda_k$ and go to step 5.
ENDIF

STEP 6. Set $x^{(k+1)} = x^{(k)} + \lambda_k d^{(k)}$.

END

vector $x^{(k)} - \alpha_k D_k \nabla f(x^{(k)})$ and the current iteration $x^{(k)}$. The descent direction is then used to define the new approximation $x^{(k+1)} = x^{(k)} + \lambda_k d^{(k)}$, where the line-search parameter λ_k is defined by a standard Armijo line-search procedure that ensures the monotone reduction of the objective function at each iteration. The global convergence can be obtained by following Birgin *et al.* (2000, 2003) and Bonettini *et al.* (2009), where the more general case based on non-monotone line-search procedures is also considered. We emphasize that any choice of the step-length $\alpha_k \in [\alpha_{min}, \alpha_{max}]$ and the scaling matrix $D_k \in \mathcal{D}$ are allowed; this freedom of choice can then be fruitfully exploited for introducing performance improvements, as discussed in the next section.

3.2 Scaling matrix and step-length

The choice of the scaling matrix has to be addressed with the goal of improving the convergence rate of the image reconstruction process while avoiding to increase excessively the computational cost of the single iteration. In the case of twice continuously differentiable objective function, a possible choice is to use a diagonal scaling matrix whose nontrivial elements approximate the diagonal entries of the

inverse of the Hessian matrix $\nabla^2 f(x)$, for example by choosing

$$(D_k)_{ii} = \min\left\{L_2, \max\left\{L_1, \left(\left(\nabla^2 f(x^{(k)})\right)_{ii}\right)^{-1}\right\}\right\}. \tag{3.4}$$

However, since the computation of the diagonal entries of the Hessian might represent an expensive task, the commonly used choice for the scaling matrix is the one suggested by the RL algorithm and its regularized versions, namely

$$D_k = \mathrm{diag}\left(\min\left\{L_2, \max\left\{L_1, \frac{x^{(k)}}{\gamma + \eta V_1(x^{(k)})}\right\}\right\}\right), \tag{3.5}$$

where only the indexes in R are considered in the case of boundary effect correction. In several applications of SGP to image deblurring the above scaling matrix has been shown to be very successful in accelerating the approximation of suited reconstructions, in comparison with gradient projection based approaches that avoid the use of scaling matrices (Bonettini *et al.* 2009, 2012).

As concerns the step-length parameter, an effective selection strategy is obtained by adapting to the context of the scaled gradient projection methods the two Barzilai & Borwein (1988) rules (hereafter denoted by BB), which are widely used in standard non-scaled gradient methods for unconstrained minimization problems. For the non-scaled case, the recent literature suggests effective alternation strategies of two BB step-length updating rules, derived by a careful analysis of their properties in the case of unconstrained minimization of quadratic functions. In particular, their ability in approximating the eigenvalues of the objective Hessian is exploited to design adaptive alternation strategies able to improve significantly the convergence rate of the gradient scheme (Zhou *et al.* 2006; Frassoldati *et al.* 2008). Numerical evidence is available that confirms the efficiency of these alternated BB rules also in case of nonlinear constrained minimization problems (Serafini *et al.* 2005; Loris *et al.* 2009).

When the scaled direction $D_k \nabla f(x^{(k)})$ is exploited within a step of the form $x^{(k)} - \alpha_k D_k \nabla f(x^{(k)})$, the standard BB step-length rules can be generalized as follows:

$$\alpha_k^{(BB1)} = \frac{(s^{(k-1)})^T D_k^{-1} D_k^{-1} s^{(k-1)}}{(s^{(k-1)})^T D_k^{-1} t^{(k-1)}}, \tag{3.6}$$

$$\alpha_k^{(BB2)} = \frac{(s^{(k-1)})^T D_k t^{(k-1)}}{(t^{(k-1)})^T D_k D_k t^{(k-1)}}, \tag{3.7}$$

where $s^{(k-1)} = x^{(k)} - x^{(k-1)}$ and $t^{(k-1)} = \nabla f(x^{(k)}) - \nabla f(x^{(k-1)})$; when $D_k = I$ the above formulas lead to the standard BB rules.

In SGP, the values produced by these rules are constrained into the interval $[\alpha_{min}, \alpha_{max}]$ in the following way:

IF $(s^{(k-1)})^T D_k^{-1} t^{(k-1)} \leq 0$ THEN
  $\alpha_k^{(1)} = \min\{10 \cdot \alpha_{k-1},\ \alpha_{max}\}$;
ELSE

$$\alpha_k^{(1)} = \min\left\{\alpha_{max},\ \max\left\{\alpha_{min},\ \alpha_k^{(BB1)}\right\}\right\};$$

ENDIF

IF $(s^{(k-1)})^T D_k t^{(k-1)} \leq 0$ THEN

$$\alpha_k^{(2)} = \min\left\{10 \cdot \alpha_{k-1},\ \alpha_{max}\right\};$$

ELSE

$$\alpha_k^{(2)} = \min\left\{\alpha_{max},\ \max\left\{\alpha_{min},\ \alpha_k^{(BB2)}\right\}\right\};$$

ENDIF

The criterion adopted in SGP for alternating between the above step-lengths is derived from that proposed in Frassoldati *et al.* (2008) and can be stated as follows:

IF $\alpha_k^{(2)}/\alpha_k^{(1)} \leq \tau_k$ THEN

$$\alpha_k = \min_{j=\max\{1,k+1-M_\alpha\},\ldots,k} \alpha_j^{(2)}; \tag{3.8}$$

$$\tau_{k+1} = 0.9 \cdot \tau_k;$$

ELSE

$$\alpha_k = \alpha_k^{(1)}; \qquad \tau_{k+1} = 1.1 \cdot \tau_k;$$

ENDIF

where M_α is a prefixed positive integer and $\tau_1 \in (0, 1)$. In contrast to the criterion proposed in Frassoldati *et al.* (2008), that is thought for the non-scaled case ($D_k = I$) and uses a constant threshold $\tau_k = \tau \in (0, 1)$ in the switching condition, here a variable threshold is exploited with the aim of avoiding the selection of the same rule for a too large number of iterations. A wide computational study suggests that this alternation criterion is more suitable in terms of convergence rate than the strategy proposed by Zhou *et al.* (2006) and the use of a single BB rule (Bonettini *et al.* 2009; Favati *et al.* 2010; Zanella *et al.* 2009). Furthermore, in our experience, the use of the BB values provided by Equation (3.8) (that are generally lower than those provided by $\alpha_k^{(1)}$) in the first iterations slightly improves the reconstruction accuracy and, consequently, in the proposed SGP version we start the step-length alternation only after the first 20 iterations.

3.3 Choice of the parameters and implementation

Even if the number of SGP parameters is certainly higher than those of the RL and OSEM approaches, the huge amount of tests carried out in several applications has led to an optimization of these values, which allows the user to have at his disposal a robust approach without the need of an expensive problem-dependent parameter tuning. In the following we provide some comments on each of these parameters:

- $x^{(0)}$: although any array can be used as starting point of the algorithm, the two commonly used images are either the detected one (or one of the detected images in the case of multiple deconvolution) or a constant image with pixel values equal to the background-subtracted flux (or mean flux in the case of multiple deconvolution) of the noisy data divided by the number

of pixels. If the boundary effect correction is considered, only the pixels in the object array R become equal to this constant, while the remaining values of $\bar{S}$ are set to zero. Our experience showed no clear preference of the former choice with respect to the latter, that is typically used in the standard RL approach;

- η, θ: the sufficient decrease parameter η and the step-reduction parameter θ control, respectively, the severity of the objective function decrease condition and the number of backtracking reductions. The parameter η has been set to 10^{-4} as usually done in literature (see, for example, Birgin *et al.* 2000), while the value $\theta = 0.4$ resulted to be a good compromise to get a sufficiently large step size calculated with a low number of reductions;

- $\alpha_{min}, \alpha_{max}, \alpha_0$: the bounds $\alpha_{min}, \alpha_{max}$ of the step-length parameter α_k are safeguard values that have to be considered for the algorithm to ensure the theoretical convergence. Usually, a very large range $(\alpha_{min}, \alpha_{max})$ is exploited in combination with BB-like step-length selections (Birgin *et al.* 2000, set such values to 10^{-30} and 10^{30}); we found that the interval $(10^{-5}, 10^{5})$ is suited both for working with the rules (3.6)-(3.7) and for avoiding extreme step-length values. As far as the starting parameter α_0 concerns, the value 1.3 has been chosen to have an initial step slightly longer than the RL one;

- initial value for τ_k: as previously observed, the switching condition between the step-length (3.8) and the value $\alpha_k^{(1)}$ works after the first 20 iterations and we choose the value 0.5 as first value for the switching parameter τ_k. In our experience, in the considered imaging applications, the values provided by Equation (3.7) are generally lower than those given by (3.6) and the starting value chosen for τ_k seems well suited to activate the alternation between the two step-length rules (remember that in the non-scaled case, if $(s^{(k-1)})^T t^{(k-1)} > 0$, the inequality $\alpha_k^{(BB2)} \leq \alpha_k^{(BB1)}$ holds).

- M_α: in case of non-scaled gradient schemes for unconstrained quadratic minimization, the use of the minimum of the step-lengths $\alpha_{k-j}^{(BB2)}, j = 0, \ldots, M_\alpha$ increased the ability of the first BB rule to approximate, in the subsequent iterations, the inverse of the Hessian's smallest eigenvalues, with interesting convergence rate improvements (Frassoldati *et al.* 2008). In Bonettini *et al.* (2009), by using the setting $M_\alpha = 3$, the importance of this trick is numerically confirmed also on more general minimization problems and in case of scaled gradient projection methods; for this reason we adopted the same setting also for our SGP version.

- L_1, L_2: while in the original paper of Bonettini *et al.* (2009) the choice of the bounds (L_1, L_2) for the scaling matrices was a couple of fixed values $(10^{-10}, 10^{10})$, independent of the data, we prefer to make automatically these bounds suitable for images of any scale. In details, one step of the RL method is performed and the parameters (L_1, L_2) are tuned according to

the min/max positive values $y_{\min}/y_{\max}$ of the resulting image; moreover, for avoiding too close bounds, the following rule is implemented

IF $y_{\max}/y_{\min} < 50$ THEN
  $L_1 = y_{\min}/10$;
  $L_2 = y_{\max} \cdot 10$;
ELSE
  $L_1 = y_{\min}$;
  $L_2 = y_{\max}$;
ENDIF

The above parameter settings are at the basis of the SGP versions currently available for ML deconvolution of astronomical images, which we briefly describe.

- IDL implementation: an Interactive Data Language (IDL) package for the single and multiple deconvolution of 2D images corrupted by Poisson noise, with the optional inclusion of the boundary effect correction.
- IDL-GPU implementation: an extended version of the above IDL implementation able to exploit the resources available on Graphics Processing Units (GPUs). This SGP version is obtained by means of the CUDA (Compute Unified Device Architecture) technology, developed by NVIDIA for programming their GPUs. The CUDA framework is available within an IDL implementation through the GPUlib, a software library developed by Tech-X Corporation, that enables GPU-accelerated computation.
- Matlab implementation: a Matlab package for the deconvolution of 2D and 3D images through the minimization of the function (2.5) and the early stopping of the iterations.

These implementations and the relative documentation can be downloaded from the URL `http://www.unife.it/prin/software`. A complete C++ and C++/CUDA library collecting all the described SGP versions is in progress and will be soon available by request.

4 Numerical experiments

The application of SGP to ML problems described in Section 3 is presented, discussed and illustrated with several numerical examples in Prato *et al.* (2012). In this section we show the SGP behaviour by discussing a few of the numerical experiments presented in Prato *et al.* (2012) as well as a numerical experiment of regularized deconvolution described in Staglianò *et al.* (2011).

In the case of methods for ML problems a crucial point is the choice of the number of iterations, *i.e.* the introduction of sensible stopping rules providing sensible solutions. On the other hand, in the case of regularization methods, the crucial point is the choice of the regularization parameter. We first discuss the stopping of the iterative methods for ML reconstructions.

In the case of the reconstruction of stellar objects such as binaries, clusters etc., SGP can be pushed to convergence; in other words, iteration can be stopped when the following condition is satisfied

$$|f_0(x^{(k)}; y) - f_0(x^{(k-1)}; y)| \;\leq\; tol\; f_0(x^{(k-1)}; y), \tag{4.1}$$

where *tol* is a parameter selected by the user (in most cases we use $tol = 10^{-7}$, but a larger value can be selected to reduce the number of iterations if a poorer accuracy of the result is sufficient). We remark that the application of this criterion does not require an additional cost because $f_0(x^{(k)}; y)$ is already computed within the algorithm.

In the case of early stopping the choice of the stopping rule is a difficult task. In numerical simulations the reference object is known, let us denote it as $\tilde{x}$, and therefore at each iteration one can compute (with a small additional cost) some "distance" between $\tilde{x}$ and $x^{(k)}$. A frequently used indicator is the relative r.m.s. error defined by

$$\rho^{(k)} = \frac{||x^{(k)} - \tilde{x}||_2}{||\tilde{x}||_2}, \tag{4.2}$$

or other indicators in terms of ℓ_1-norm, KL divergence etc.. Iterations can be stopped when $\rho^{(k)}$ reaches a minimum value, thus defining a reconstruction which is "optimal" according to this criterion.

Obviously such a strategy can not be applied in the case of real data. In the vein of a discrepancy principle used for Tikhonov regularization, one can introduce the following quantity, which must be computed at each iteration and can be called a "discrepancy function"

$$D_y^{(k)} = \frac{1}{\#S} \left\| \frac{Hx^{(k)} + b - y}{\sqrt{Hx^{(k)} + b}} \right\|^2. \tag{4.3}$$

It is derived from Bardsley & Goldes (2009) while in Staglianò *et al.* (2011) it is shown that this quantity is a decreasing function of k; moreover, in the latter paper, it is proposed, on the basis of statistical considerations, that iterations could be stopped when $D_y^{(k)} \leq 1$. Another criterion, also based on a statistical analysis, is proposed in Bertero *et al.* (2010). In this case the "discrepancy function" is defined by

$$D_y^{(k)} = \frac{2}{\#S} f_0(x^{(k)}; y), \tag{4.4}$$

and its computation does not require any additional cost. It is proved that it is a decreasing function of k and again iterations can be stopped when $D_y^{(k)} \leq 1$. Examples of the application of this criterion are given in Bertero *et al.* (2010).

In the case of regularized solutions, for a given value of the regularization parameter, the iterations must be pushed to convergence using, for instance, a criterion similar to (4.1), with $f_0(x^{(k)}; y)$ replaced by $f_\beta(x^{(k)}; y)$. The problem is to select a value of β. Again, one must use different strategies in the case of simulated and real data.

In the first case, if we denote as x^*_β the minimizer of $f_\beta(x;y)$ (in practice, its approximation computed by means of an iterative method), then one can introduce again a relative r.m.s. error using a "distance" between x^*_β and $\tilde{x}$, for instance in terms of the ℓ_2-norm,

$$\rho(\beta) = \frac{||x^*_\beta - \tilde{x}||_2}{||\tilde{x}||_2}, \tag{4.5}$$

(or another indicator) and searching for the value of β minimizing this quantity. This approach obviously requires the computation of x^*_β for several values of β and can be computationally expensive.

In the case of real data one can use the discrepancy function introduced by Bardsley & Goldes (2009)

$$D_y(\beta) = \frac{1}{\#S} \left\| \frac{Hx^*_\beta + b - y}{\sqrt{Hx^*_\beta + b}} \right\|^2, \tag{4.6}$$

or that introduced by Bertero *et al.* (2010)

$$D_y(\beta) = \frac{2}{\#S} f_0(x^*_\beta; y). \tag{4.7}$$

In both cases one must search for the value of β satisfying the equation $D(\beta) = 1$. A secant-like method can be used for solving this equation; if a tolerance 10^{-3} is used, in general only 4-5 iterations are required. This approach can be useful also in the case of simulations because the value of β minimizing the error (4.5) can be searched in a neighborhood of the value provided by the discrepancy principle.

4.1 Acceleration of the RL method

In this section we show the effectiveness of SGP with respect to the RL and OSEM approaches, highlighting the speedups achievable thanks to both the algorithmic acceleration provided by SGP and the parallel implementation of the codes on GPU. We consider 256×256 HST images of the planetary nebula NGC 7027 and the galaxy NGC 6946, with two different integrated magnitudes (m) of 10 and 15, not corresponding to the effective magnitudes of these objects but introduced for obtaining simulated images with different noise levels. Such images have been convolved with an ideal PSF, simulated assuming a telescope of diameter 8.25 m, a wavelength of 2.2 μm, and a pixel size of 50 mas. A constant background term of about 13.5 mag arcsec^{-2}, corresponding to observations in K-band, is added and the resulting images are perturbed with Poisson noise and additive Gaussian noise with $\sigma = 10$ e$^-$/px. Original objects and the corresponding blurred and noisy images are shown in Figure 3. As suggested in Snyder *et al.* (1994), compensation for RON is obtained in the deconvolution algorithms by adding the constant $\sigma^2 = 100$ to the images and the background. We obtained test problems of larger size (up to 2048×2048) by means of a Fourier-based rebinning, preserving

the same background and the same noise level. The results are reported in Tables 1 and 2, where we highlight both the speedup observed between GPU and serial implementations (labeled "Par") and the one provided by the use of SGP instead of RL (labeled "Alg").

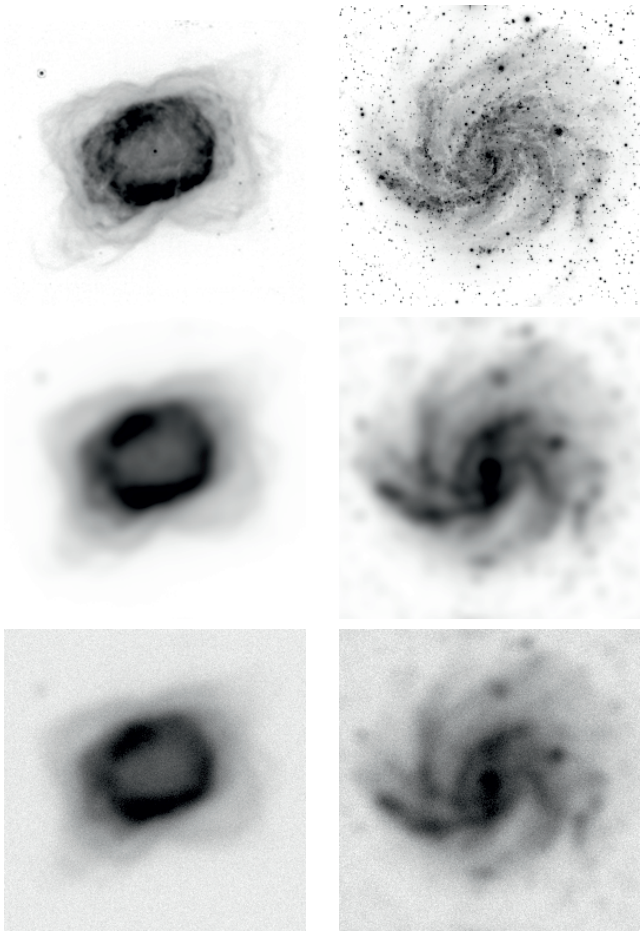

Fig. 3. Original images (*top panels*) and blurred noisy images with $m = 10$ (*middle panels*) and $m = 15$ (*bottom panels*).

As concerns the multiple-image deconvolution problem, we test the efficiency of multiple RL, OSEM, and SGP (applied to multiple RL), by means of synthetic images of LN. In particular, we simulate a model of an open star cluster based on an image of the Pleiades, by selecting the nine brightest stars characterized by the following name, position and magnitude.

Table 1. Relative r.m.s. errors, computational times, and speedups obtained by the accelerating features of SGP with respect to RL ("Alg") and by the GPU implementations ("Par"), for the nebula NGC 7027 with different image sizes. Iterations are stopped at a minimum relative r.m.s. error in the serial algorithms.

$m = 10$					
Algorithm	Size	Err	Sec	SpUp (Par)	SpUp (Alg)
RL	256^2	0.051	783.9	-	-
It = 10000*	512^2	0.051	4527	-	-
	1024^2	0.051	17610	-	-
	2048^2	0.051	80026	-	-
RL_CUDA	256^2	0.051	35.63	22.0	-
It = 10000*	512^2	0.051	69.77	64.9	-
	1024^2	0.051	149.5	118	-
	2048^2	0.051	469.1	171	-
SGP	256^2	0.052	26.14	-	30.0
It = 272	512^2	0.051	143.6	-	31.5
	1024^2	0.051	554.0	-	31.8
	2048^2	0.051	2493	-	32.1
SGP_CUDA	256^2	0.052	1.797	14.5	19.8
It = 272	512^2	0.052	3.469	41.4	20.1
	1024^2	0.052	8.016	69.1	18.7
	2048^2	0.052	25.66	97.2	18.3
$m = 15$					
Algorithm	Size	Err	Sec	SpUp (Par)	SpUp (Alg)
RL	256^2	0.068	48.27	-	-
It = 612	512^2	0.064	278.7	-	-
	1024^2	0.062	1068	-	-
	2048^2	0.062	4897	-	-
RL_CUDA	256^2	0.068	2.219	21.8	-
It = 612	512^2	0.064	4.109	67.8	-
	1024^2	0.062	9.250	115	-
	2048^2	0.062	29.13	168	-
SGP	256^2	0.068	3.016	-	16.0
It = 31	512^2	0.064	16.95	-	16.4
	1024^2	0.062	65.22	-	16.4
	2048^2	0.061	290.8	-	16.8
SGP_CUDA	256^2	0.068	0.218	13.8	10.2
It = 31	512^2	0.064	0.421	40.3	9.76
	1024^2	0.062	1.063	61.4	8.70
	2048^2	0.061	3.406	85.4	8.55

Table 2. Relative r.m.s. errors, computational times, and speedups obtained by the accelerating features of SGP with respect to RL ("Alg") and by the GPU implementations ("Par"), for the galaxy NGC 6946 with different image sizes. Iterations are stopped at a minimum relative r.m.s. error in the serial algorithms.

$m = 10$					
Algorithm	Size	Err	Sec	SpUp (Par)	SpUp (Alg)
RL	256^2	0.293	786.0	-	-
It = 10000*	512^2	0.293	4545	-	-
	1024^2	0.293	17402	-	-
	2048^2	0.293	80022	-	-
RL_CUDA	256^2	0.293	36.64	21.5	-
It = 10000*	512^2	0.293	67.94	66.9	-
	1024^2	0.293	146.7	119	-
	2048^2	0.293	463.9	172	-
SGP	256^2	0.292	88.72	-	8.86
It = 928	512^2	0.291	484.3	-	9.38
	1024^2	0.291	1854	-	9.19
	2048^2	0.291	8386	-	9.54
SGP_CUDA	256^2	0.293	7.219	12.3	5.08
It = 928	512^2	0.293	11.14	43.5	6.10
	1024^2	0.293	25.86	71.7	5.67
	2048^2	0.293	81.02	104	5.73
$m = 15$					
Algorithm	Size	Err	Sec	SpUp (Par)	SpUp (Alg)
RL	256^2	0.311	114.9	-	-
It = 1461	512^2	0.307	644.3	-	-
	1024^2	0.306	2574	-	-
	2048^2	0.306	11689	-	-
RL_CUDA	256^2	0.311	5.375	21.4	-
It = 1461	512^2	0.307	9.656	66.7	-
	1024^2	0.306	22.41	115	-
	2048^2	0.306	68.44	171	-
SGP	256^2	0.311	3.672	-	31.3
It = 38	512^2	0.308	20.36	-	31.6
	1024^2	0.307	78.20	-	32.9
	2048^2	0.306	354.0	-	33.0
SGP_CUDA	256^2	0.311	0.266	13.8	20.2
It = 38	512^2	0.307	0.531	38.3	18.2
	1024^2	0.307	1.344	58.2	16.7
	2048^2	0.306	4.188	84.5	16.3

Star Name	X	Y	m
ALCYONE	228	246	12.86
ATLAS	156	237	13.62
ELECTRA	340	247	13.70
MAIA	299	295	13.86
MEROPE	277	216	14.17
TAYGETA	326	313	14.29
PLEIONE	155	253	15.09
CELAENO	343	280	15.44
ASTEROPE	296	330	15.64

The coordinate values are deduced from the picture found in the Wikipedia page (`http://en.wikipedia.org/wiki/Pleiades`), resized to a 256×256 pixels image, and immersed in a 512×512 pixels image. In this way we generated a relatively compact cluster in the center of the image. These objects are convolved with three PSFs corresponding to three equispaced orientations of the baseline, $0°$, $60°$, and $120°$, obtained by rotating the PSF described in the Introduction and shown in Figure 2. Background emission in K band (13.5 mag/arcsec^2) is added to the results, which are also perturbed with Poisson and Gaussian ($\sigma = 10$ e$^-$/px) noise. The object and one of the corresponding blurred and noisy images are shown in Figure 4.

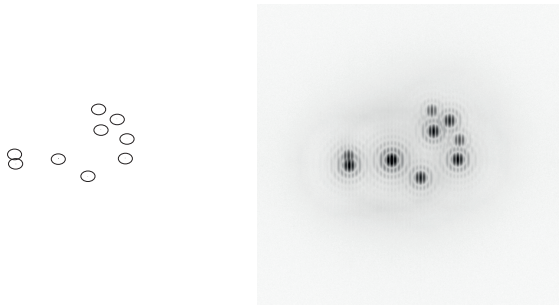

Fig. 4. Star cluster data: simulated object (*left panel*, stars are marked by circles) and corresponding blurred and noisy image (*right panel*).

In this case, iterations are pushed to convergence and therefore the stopping rule is given by the condition (4.1). We use different values of *tol*, specifically 10^{-3}, 10^{-5}, and 10^{-7}. In order to measure the quality of the reconstruction, we introduce an average relative error of the magnitudes defined by

$$\text{av\_rel\_er} = \frac{1}{q} \sum_{j=1}^{q} \frac{|m_j - \widetilde{m}_j|}{\widetilde{m}_j}, \tag{4.8}$$

where q is the number of stars (in our case $q = 9$) and $\widetilde{m}_j$ and m_j are respectively the true and the reconstructed magnitudes. The results are reported in Table 3.

Table 3. Reconstruction of the star cluster with three 512×512 equispaced images. The error is the average relative error in the magnitudes defined in Equation (4.8).

	tol = 1e-3			
Algorithm	It	Err	Sec	SpUp
RL	319	2.39e-4	393.4	-
RL_CUDA	319	2.38e-4	4.641	84.8
OSEM	151	1.63e-4	220.8	-
OSEM_CUDA	151	1.62e-4	2.421	91.2
SGP	71	1.35e-3	97.80	-
SGP_CUDA	71	1.29e-3	1.641	59.6
	tol = 1e-5			
Algorithm	It	Err	Sec	SpUp
RL	1385	6.65e-5	1703	-
RL_CUDA	1385	6.64e-5	19.38	87.9
OSEM	675	5.64e-5	980.6	-
OSEM_CUDA	675	5.64e-5	10.75	91.2
SGP	337	5.89e-4	455.2	-
SGP_CUDA	337	1.79e-4	7.187	63.3
	tol = 1e-7			
Algorithm	It	Err	Sec	SpUp
RL	7472	5.64e-5	9180	-
RL_CUDA	7472	5.98e-5	104.8	87.6
OSEM	3750	6.13e-5	5442	-
OSEM_CUDA	3750	5.98e-5	59.52	91.4
SGP	572	7.37e-5	772.6	-
SGP_CUDA	572	7.05e-5	12.20	63.3

4.2 Boundary effect correction

We show now the effectiveness of the boundary effect correction described in Section 2.3 on the RL, OSEM and SGP algorithms. The numerical experiments are designed according to the following procedure: we select a 256×256 HST image of the Crab nebula NGC 19521, and we build the blurred and noisy image by means of the same procedure (and the same parameters) adopted in the previous tests, but using the AO-corrected PSF[3] shown in Figure 5.

The parameters of this PSF (pixel size, diameter of the telescope, etc.) are not provided. However, it has approximately the same width as the ideal PSF described in Section 4.1. We apply RL and SGP first to the full image, and then to four 160×160 partly overlapping sub-domains with the addition of the boundary effect correction. The full deconvolved image is obtained as a mosaic of the central parts (see Fig. 6). The same comparison is performed in the multiple-image case

[3]Downloaded from `http://www.mathcs.emory.edu/~nagy/RestoreTools/index.html`

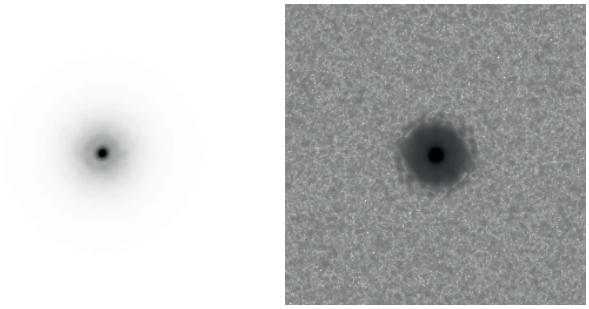

Fig. 5. The PSF used for the single deconvolution experiments with boundary effect correction (*left panel*) and the corresponding MTF (*right panel*). Both are represented in reversed gray scale.

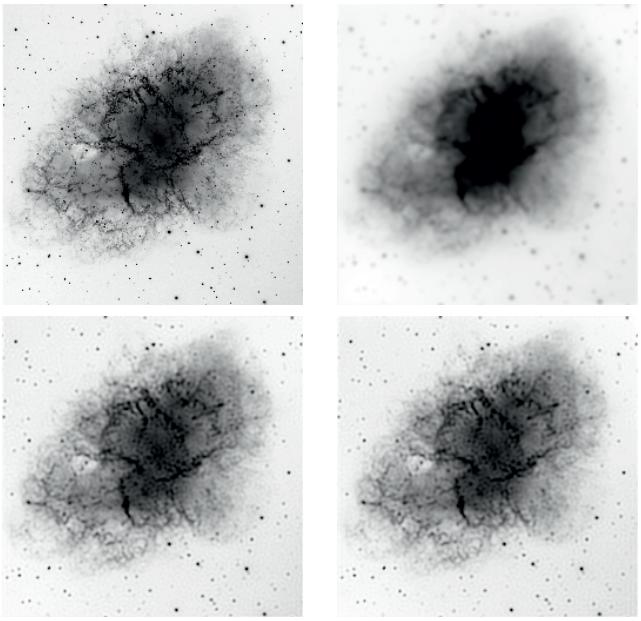

Fig. 6. Crab nebula test: the object (*top left*), its blurred and noisy image in the case $m = 10$ (*top right*), the reconstructions of the full image with SGP (*bottom left*) and as a mosaic of four reconstructions of partially overlapping subdomains, using SGP with boundary effect correction (*bottom right*).

by using three 512×512 images of the nebula NGC 7027 obtained by means of the LN PSFs described in the previous section (in this test, 320×320 sub-domains are extracted). In Tables 4 and 5 we report the serial and parallel performances of RL, OSEM (when multiple images were available) and SGP in both cases of

Table 4. Reconstruction of the 256×256 Crab object with a standard deconvolution and as a mosaic of the reconstructions of four subimages with boundary effect correction.

	Standard deconvolution				Boundary effects correction			
				$m = 10$				
Algorithm	It	Err	Sec	SpUp	It	Err	Sec	SpUp
RL	5353	0.128	419.8	-	4070	0.129	1146	-
RL_CUDA	5353	0.128	19.45	21.6	4070	0.129	61.55	18.6
SGP	151	0.129	14.28	-	129	0.129	46.42	-
SGP_CUDA	151	0.129	1.219	11.7	129	0.133	4.342	10.7
				$m = 12$				
Algorithm	It	Err	Sec	SpUp	It	Err	Sec	SpUp
RL	954	0.136	74.83	-	696	0.137	196.5	-
RL_CUDA	954	0.136	3.516	21.3	696	0.137	10.99	17.9
SGP	52	0.137	4.984	-	53	0.137	19.41	-
SGP_CUDA	52	0.137	0.406	12.3	53	0.137	1.922	10.1
				$m = 15$				
Algorithm	It	Err	Sec	SpUp	It	Err	Sec	SpUp
RL	128	0.172	10.09	-	99	0.172	28.08	-
RL_CUDA	128	0.172	0.483	20.9	99	0.172	1.704	16.5
SGP	10	0.172	1.093	-	9	0.172	3.859	-
SGP_CUDA	10	0.172	0.093	11.8	9	0.172	0.360	10.7

Table 5. Reconstruction of the nebula using three equispaced 512×512 images, in the cases of standard deconvolution and as a mosaic of four reconstructed subimages with boundary effect correction.

	Standard deconvolution				Boundary effects correction			
				$m = 10$				
Algorithm	It	Err	Sec	SpUp	It	Err	Sec	SpUp
RL	3401	0.032	4364	-	2899	0.034	13978	-
RL_CUDA	3401	0.032	48.00	90.9	2899	0.034	174.2	80.2
OSEM	1133	0.032	1602	-	950	0.034	5447	-
OSEM_CUDA	1133	0.032	18.59	86.2	950	0.034	64.03	85.1
SGP	144	0.033	220.7	-	160	0.034	873.3	-
SGP_CUDA	144	0.033	3.563	61.9	160	0.034	15.45	56.5
	Standard deconvolution				Boundary effects correction			
				$m = 15$				
Algorithm	It	Err	Sec	SpUp	It	Err	Sec	SpUp
RL	353	0.091	441.5	-	243	0.094	1174	-
RL_CUDA	353	0.091	4.937	89.4	243	0.094	15.28	76.8
OSEM	117	0.091	165.7	-	81	0.094	479.1	-
OSEM_CUDA	117	0.091	2.062	80.4	81	0.094	5.939	80.7
SGP	16	0.087	26.14	-	11	0.087	69.88	-
SGP_CUDA	16	0.087	0.546	47.9	11	0.086	1.532	45.6

full and splitted deconvolution. The computational times reported in the case of boundary effect correction refer to the reconstruction of all the four sub-domains.

4.3 Edge-preserving regularization

As an example of regularized reconstruction we consider the case of an edge-preserving prior, called hypersurface (HS) regularization (Charbonnier *et al.* 1997). It is defined by

$$f_1(x) = \sum_{j_1,j_2=1}^{n} \psi_\delta(D^2_{j_1,j_2}), \quad \delta \neq 0, \tag{4.9}$$

where

$$\psi_\delta(t) = \sqrt{t+\delta^2}, \quad D^2_{j_1,j_2} = (x_{j_1+1,j_2} - x_{j_1,j_2})^2 + (x_{j_1,j_2+1} - x_{j_1,j_2})^2. \tag{4.10}$$

For δ small this regularization is used as a smoothed approximation to total variation (TV) (see, for instance, Vogel 2002; Bardsley & Luttman 2009; Zanella *et al.* 2009; Defrise *et al.* 2011; Staglianò *et al.* 2011; for TV regularization, see Dey *et al.* 2006; Le *et al.* 2007; Brune *et al.* 2010; Setzer *et al.* 2010; Bonettini & Ruggiero 2011).

By computing the gradient of $f_1(x)$ one finds the following natural choice for the function $V_1(x)$ to be inserted in the scaling of the gradient of the complete objective function (see Eq. (3.5))

$$[V_1(x)]_{j_1,j_2} = [2\psi'_\delta(D^2_{j_1,j_2}) + \psi'_\delta(D^2_{j_1,j_2-1}) + \psi'_\delta(D^2_{j_1-1,j_2})], \tag{4.11}$$

where $\psi'_\delta(t)$ is the derivative of $\psi_\delta(t)$.

We consider as reference object the frequently used spacecraft image characterized by sharp details (Fig. 7). The size is 256×256 (in Fig. 7 we show only the central part), and the maximum value is 255; it is superimposed to a background $b = 1$. Moreover, for generating images with different noise levels, we consider three other versions with maximum values 2550, 25 500 and 25 5000, respectively (and backgrounds 10, 100, 1000), obtained by scaling the original object. Next, the four versions are convolved with a PSF and then perturbed with Poisson noise (we did not add Gaussian noise). The PSF used is the one already described in the previous section and shown in Figure 5. For each image we generate 25 different realizations of noise so that we have a total of 200 noisy images.

We first consider unregularized reconstructions. Early stopping of the iteration is based on two stopping rules. The first consists in computing at each iteration the relative r.m.s. error $\rho^{(k)}$, defined in Equation (4.2) and stopping the iteration when this parameter reaches its minimum value. The second consists in computing the discrepancy $D^{(k)}$, introduced by Bardsley & Goldes (2009), defined in Equation (4.3), and stopping the iteration when it crosses 1. Iteration is initialized with $x^{(0)} = y_{am} - b$ (where y_{am} is the arithmetic mean of the image values). In all cases, $D^{(0)} > 1$ and $D^{(k)}$ is decreasing for increasing k, providing a solution of the equation $D^{(k)} = 1$.

The results are given in Table 6 for the four images of the spacecraft with different noise levels. For each image we report average value and standard deviation both of the number of iterations and of the reconstruction error, computed using

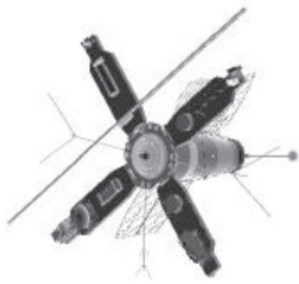

Fig. 7. The spacecraft image, represented in reversed gray scale.

the 25 realizations of noise. The reconstruction error is weakly dependent on the noise realization even if the number of iterations is strongly varying. Stopping based on Bardsley & Goldes criterion works better at the highest noise level.

Table 6. Unregularized reconstructions of the spacecraft: errors and iterations.

	Minimum error		**Discrepancy**	
	iter	**error (%)**	**iter**	**error (%)**
255	73 ± 19	40.1 ± 0.6	33 ± 14	43.9 ± 9.6
2550	186 ± 58	33.5 ± 0.4	117 ± 84	30.9 ± 11.5
25 500	465 ± 198	29.3 ± 0.3	593 ± 322	30.0 ± 1.1
255 000	1449 ± 376	26.9 ± 0.2	1788 ± 553	27.3 ± 0.5

In column (a) of Figure 8 we show the four images with different noise levels; in columns (b) and (c) the reconstructions corresponding to the minimum r.m.s. error and to the criterion of Bardsley & Goldes, respectively; finally, in the last column, we show the normalized residuals defined by

$$R^{(k)} = \frac{Hx^{(k)} + b - y}{\sqrt{Hx^{(k)} + b}}, \tag{4.12}$$

and computed in the case of the reconstructions of column (b). Artifacts are present at the lowest noise levels, due to the reconstruction method.

The previous numerical test is performed for investigating possible improvements of the reconstructions due to the use of edge-preserving regularization, as provided by the penalty function of Equation (4.9), with $\delta = 10^{-4}$, and we use the SGP algorithm with the scaling defined in terms of the function (4.11). This scaling has been already successfully used in the case of denoising of Poisson data (Zanella *et al.* 2009) and we use the same parameters of the algorithm described in that paper. For a given β, iteration is stopped when $|f_\beta(x^k; y) - f_\beta(x^{k-1}; y)| \leq 10^{-7} f_\beta(x^{k-1}; y)$. The choice of β is performed by computing x^*_β and using a secant-like method for satisfying the criterion of Bardsley & Goldes, with a tolerance of 10^{-3}. Next, the value of β providing the minimum r.m.s. error is obtained by

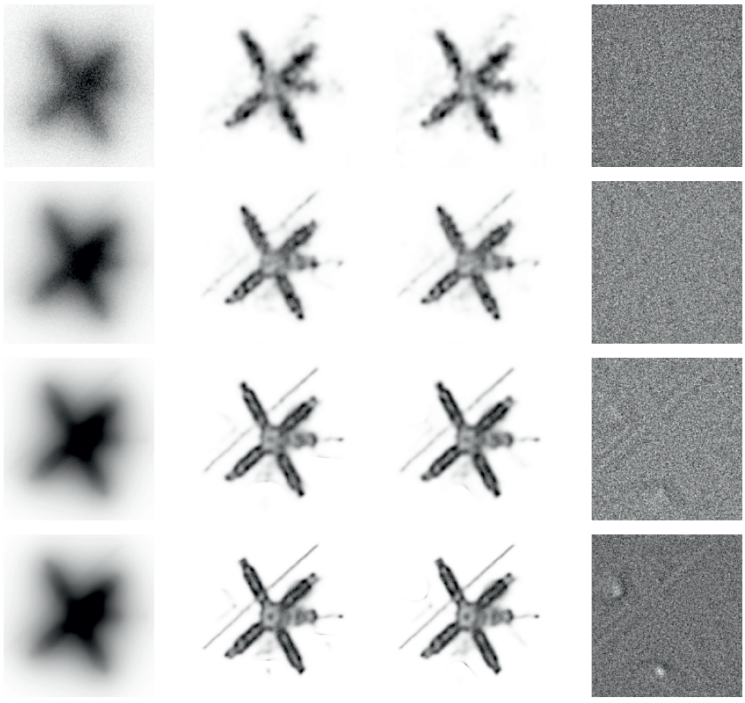

Fig. 8. Unregularized reconstructions of the spacecraft: (a) the blurred images; (b) the reconstructions with minimum r.m.s. error; (c) the reconstructions satisfying the criterion of Bardsey & Goldes; (d) the normalized residuals in the case of the reconstructions of column (b).

searching in an interval around the value provided by the discrepancy equation. Also in this experiment we considered 25 different realization of noise for each test image.

The reconstruction errors and the number of required iterations are reported in Table 7. The average reconstruction errors are smaller than those obtained in the unregularized case, with comparable standard deviations. As concerns the use the discrepancy criterion, it provides acceptable results except at the highest noise level. The reconstructions and the normalized residuals are shown in Figure 9. The residuals are still affected by strong artifacts, at least in the case of the lowest noise levels.

5 Concluding remarks and perspectives

We briefly discuss the main points of this paper by considering first the case of the ML problems.

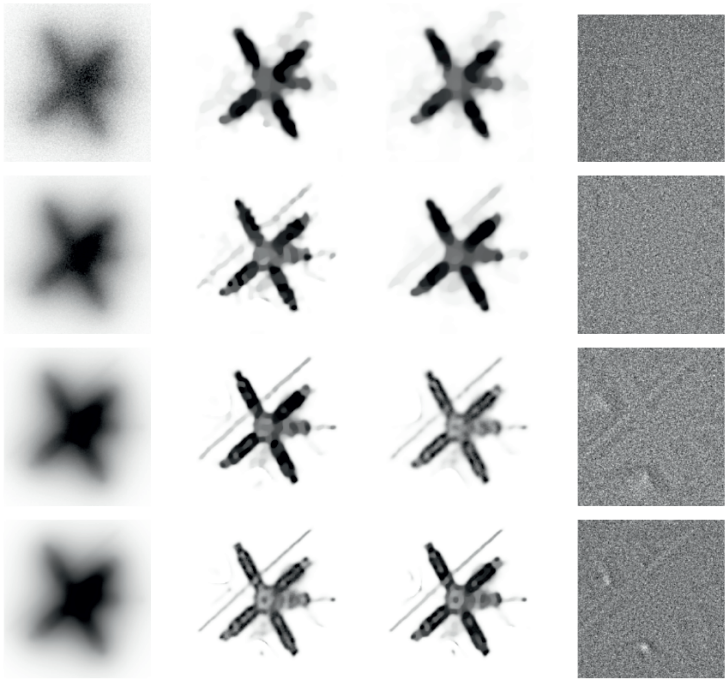

Fig. 9. Regularized reconstructions of the spacecraft: (a) the blurred images; (b) the reconstruction with the minimum r.m.s. error; (c) the reconstructions satisfying the criterion of Bardsley & Goldes; (d) the normalized residuals in the case of the reconstructions of column (b).

Table 7. Regularized reconstructions of the spacecraft: iterations and errors.

	Minimum error		**Discrepancy**	
	iter	**error (%)**	**iter**	**error (%)**
255	247 ± 54	36.4 ± 0.6	367 ± 198	40.7 ± 4.5
2550	458 ± 138	30.7 ± 0.3	462 ± 210	32.2 ± 1.0
25 500	1308 ± 124	26.1 ± 0.2	933 ± 221	26.9 ± 0.7
255 000	2190 ± 409	24.3 ± 0.8	1700 ± 462	24.9 ± 1.0

Both RL and SGP (with the scaling suggested by RL) converge to minimizers of the data fidelity function defined in terms of the generalized KL divergence, in particular to the unique minimizer if the function is strictly convex. In the case of the reconstruction of binaries or star clusters, the algorithms must be pushed to convergence and, of course, they provide the same result. However, as follows for instance from Table 4, the convergence of SGP is much faster than that of RL with a speed-up increasing from 4 to about 12 for the serial implementation, and from 3 to 9 for the parallel implementation, if the required accuracy is increased.

On the other hand, in the case of complex objects such as nebulae, galaxies or similar, it is well known that an early stopping of the iterations is required in the case of RL. Indeed the algorithm has the so-called *semi-convergence* property, in the sense that the iterations first approach the true object (we are talking about simulations) and then go away. Therefore it is interesting to remark that the iterations of SGP have a similar behaviour. The trajectories formed by the iterations of the two algorithms are different even when the starting point is the same, but the two points of minimal distance from the true object are very close (in general, visually indistinguishable), and SGP reaches the point with a number of steps much smaller than RL. The gain in computational time is considerable in spite of the fact that the cost of one SGP iteration is about 30% higher than that of one RL iteration (Bonettini *et al.* 2009). If we look at Tables 1 and 2, we find a speed-up ranging from 10 to 30 in the serial implementation, and from 6 to 20 in the parallel implementation. The speed-up depends on the specific object and, in general, it is higher when a higher number of iterations is required. We conclude these brief remarks by pointing out that, in the case of faint objects, SGP implemented on GPU is able to process a 2048×2048 image in a few seconds.

In the case of a Bayesian approach we do not still have estimates of the speed-up provided by SGP algorithms (with the scaling suggested by SGM) with respect to other algorithms and, in particular, SGM (with or without line-search in terms, for instance, of Armijo rule). In this paper we give only a few preliminary results obtained in the case of SGP deconvolution with edge-preserving regularization. The speed-up provided by GPU implementation of SGP edge-preserving denoising of Poisson data is estimated in Serafini *et al.* (2010) (see also Ruggiero *et al.* 2010, for GPU implementation of SGP deconvolution without regularization). A speed-up of the order of 20 is observed.

We expect that also in the case of regularized deconvolution SGP can provide very fast algorithms, reducing the computational time required for the estimation of the value of the regularization parameter with one of the methods described in Section 4 or other proposed methods. These topics are under investigation by our group. The goal is to provide a library of algorithms for different regularization functions.

We conclude by remarking that the SGP approach has been already applied to other problems, in particular to the computation of nonnegative least-square solutions (Benvenuto *et al.* 2010), to the nonnegative reconstruction of astronomical data from sparse Fourier data (Bonettini & Prato 2010) and to the least-squares problem with a sparsity regularization (Loris *et al.* 2009).

References

Anconelli, B., Bertero, M., Boccacci, P., Carbillet, M., & Lanteri, H., 2006, A&A, 448, 1217

Arcidiacono, C., Diolaiti, E., Tordi, M., Ragazzoni, R., Farinato, J., Vernet, E., & Marchetti, E., 2004, Appl. Opt., 43, 4288

Bardsley, J.M., & Goldes, J., 2009, Inverse Probl., 25, 095005

Bardsley, J.M., & Luttman, A., 2009, Adv. Comput. Math., 31, 35

Barrett, H.H., & Meyers, K.J., 2003, Foundations of Image Science (Wiley and Sons, New York), 1047

Barzilai, J., & Borwein, J.M., 1988, IMA J. Numer. Anal., 8, 141

Benvenuto, F., La Camera, A., Theys, C., Ferrari, A., Lantéri, H., & Bertero, M., 2008, Inverse Probl., 24, 035016

Benvenuto, F., La Camera, A., Theys, C., Ferrari, A., Lantéri, H., & Bertero, M., 2012, Inverse Probl., 28, 069502

Benvenuto, F., Zanella, R., Zanni, L., & Bertero, M., 2010, Inverse Probl., 26, 025004

Bertero, M., & Boccacci, P., 2000, A&AS, 144, 181

Bertero, M., & Boccacci, P., 2005, A&A, 437, 369

Bertero, M., Boccacci, P., Talenti, G., Zanella, R., & Zanni, L., 2010, Inverse Probl., 26, 10500

Bertero, M., Boccacci, P., La Camera, A., Olivieri, C., & Carbillet, M., 2011, Inverse Probl., 27, 113001

Birgin, E.G., Martínez, J.M., & Raydan, M., 2000, SIAM J. Optimiz., 10, 1196

Birgin, E.G., Martínez, J.M., & Raydan, M., 2003, IMA J. Numer. Anal., 23, 539

Bonettini, S., Zanella, R., & Zanni, L., 2009, Inverse Probl., 25, 015002

Bonettini, S., & Prato, M., 2010, Inverse Probl., 26, 095001

Bonettini, S., & Ruggiero, V., 2011, Inverse Probl., 27, 095001

Bonettini, S., Landi, G., Loli Piccolomini, E., & Zanni, L., 2012, Intern. J. Comp. Math., in press, DOI: 10.1080/00207160.2012.716513

Brune, C., Sawatzky, A., & Burger, M., 2010, J. Computer Vision, 92, 211

Charbonnier, P., Blanc-Féraud, L., Aubert, G., & Barlaud, A., 1997, IEEE T. Image Process., 6, 298

Defrise, M., Vanhove, C., & Liu, X., 2011, Inverse Probl., 27, 065002

Dey, N., Blanc-Féraud, L., Zimmer, C., *et al.*, 2006, Micros. Res. Techniq., 69, 260

Favati, P., Lotti, G., Menchi, O., & Romani, F., 2010, Inverse Probl., 26, 085013

Frassoldati, G., Zanni, L., & Zanghirati, G., 2008, J. Indust. Manag. Optim., 4, 299

Herbst, T.M., Ragazzoni, R., Andersen, D., *et al.*, 2003, Proc. SPIE, 4838, 456

Hudson, H.M., & Larkin, R.S., 1994, IEEE T. Med. Imaging, 13, 601

Lantéri, H., Roche, M., & Aime, C., 2002, Inverse Probl., 18, 1397

Le, T., Chartran, R., & Asaki, T.J., 2007, J. Math. Imaging Vis., 27, 257

Loris, I., Bertero, M., De Mol, C., Zanella, R., & Zanni, L., 2009, Appl. Comput. Harmon. A., 27, 247

Lucy, L.B., 1974, AJ, 79, 745

Lucy, L.B., & Hook, R.N., 1992, ASP Conf. Series, 25, 277

Prato, M., Cavicchioli, R., Zanni, L., Boccacci, P., & Bertero, M., 2012, A&A, 539, A133

Richardson, W.H., 1972, J. Opt. Soc. Am., 62, 55

Ruggiero, V., Serafini, T., Zanella, R., & Zanni, L., 2010, J. Global Optim., 48, 145

Serafini, T., Zanghirati, G., & Zanni, L., 2005, Optim. Method Softw., 20, 353

Serafini, T., Zanella, R., & Zanni, L., 2010, Adv. Parallel Comput., 19, 59

Setzer, S., Steidl, G., & Teuber, T., 2010, J. Vis. Commun. Image R., 21, 193

Shepp, L.A., & Vardi, Y., 1982, IEEE T. Med. Imaging, 1, 113
Snyder, D.L., Hammoud, A.M., & White, R.L., 1993, J. Opt. Soc. Am., A10, 1014
Staglianò, A., Boccacci, P., & Bertero, M., 2001, Inverse Probl., 27, 125003
Vogel, C.R., 2002, Computational Methods for Inverse Problems (SIAM, Philadelphia)
Zanella, R., Boccacci, P., Zanni, L., & Bertero, M., 2009, Inverse Probl., 25, 045010
Zhou, B., Gao, L., & Dai, Y.H., 2006, Comput. Optim. Appl., 35, 69