

# The study of an iterative method for the reconstruction of images corrupted by Poisson and Gaussian noise

F Benvenuto<sup>1</sup>, A La Camera<sup>2</sup>, C Theys<sup>3</sup>, A Ferrari<sup>3</sup>, H Lantéri<sup>3</sup> and M Bertero<sup>2</sup>

<sup>1</sup> DIMA, Università di Genova, Via Dodecaneso 35, I 16146 Genova, Italy

<sup>2</sup> DISI, Università di Genova, Via Dodecaneso 35, I 16146 Genova, Italy

<sup>3</sup> Laboratoire Hippolyte Fizeau, Université de Nice Sophia Antipolis, CNRS UMR6525, 06108 Nice Cedex 2, France

E-mail: [bertero@disi.unige.it](mailto:bertero@disi.unige.it)

Received 24 September 2007, in final form 20 February 2008

Published 29 April 2008

Online at [stacks.iop.org/IP/24/035016](http://stacks.iop.org/IP/24/035016)

## Abstract

In 1993, Snyder *et al* investigated the maximum-likelihood (ML) approach to the deconvolution of images acquired by a charge-coupled-device camera and proved that the iterative method proposed by Llacer and Nuñez in 1990 can be derived from the expectation-maximization method of Dempster *et al* for the solution of ML problems. The utility of the approach was shown on the reconstruction of images of the Hubble space Telescope. This problem deserves further investigation because it can be important in the deconvolution of images of faint objects provided by next-generation ground-based telescopes that will be characterized by large collecting areas and advanced adaptive optics. In this paper, we first prove the existence of solutions of the ML problem by investigating the properties of the negative log of the likelihood function. Next, we show that the iterative method proposed by the above-mentioned authors is a scaled gradient method for the constrained minimization of this function in the closed and convex cone of the non-negative vectors and that, if it is convergent, the limit is a solution of the constrained ML problem. Moreover, by looking for the asymptotic behavior in the regime of high numbers of photons, we find an approximation that, as proved by numerical experiments, works well for any number of photons, thus providing an efficient implementation of the algorithm. In the case of image deconvolution, we also extend the method to take into account boundary effects and multiple images of the same object. The approximation proposed in this paper is tested on a few numerical examples.

## 1. Introduction

The maximum likelihood (ML) approach is frequently used in image reconstruction/deconvolution. It is based on modeling the noise corrupting the detected image. In the

case of additive Gaussian noise, it is equivalent to minimize the least-squares (LS) functional. In the case of Poisson noise, as arises from photon counting, it is equivalent to minimize the Kullback–Leibler (KL) divergence (also called I-divergence) of the computed data from the detected data.

Charged-coupled-device (CCD) cameras are used for acquiring images both in microscopy and astronomy. They are preceded by optical elements (lenses or mirrors), which limit resolution and introduce aberrations; moreover, in the case of ground-based telescopes, additional significant aberrations are introduced by the atmospheric turbulence, even when its effect is partially corrected by adaptive optics (AO). As a consequence, the acquired images must be processed by deblurring methods. If a ML approach is adopted, a study of the noise introduced by a CCD camera is required. This problem was considered by researchers working on the restoration of the images of the Hubble Space Telescope at the beginning of the 1990s [16, 23, 25], and an accurate statistical model is described, for instance, in [24]. We sketch it here for the convenience of the reader.

We denote by  $x = \{x_j\}_{j \in R}$  the object to be imaged, where  $j$  is, in general, a multi-index, i.e. a pair of indices in the 2D case (astronomy) or a triple of indices in the 3D case (microscopy). Moreover, we denote by  $y = \{y_i\}_{i \in S}$  the detected image, where, again,  $i$  is, in general, a multi-index. The two sets of index values,  $R$  and  $S$ , may coincide but they may also be different. *Card*  $S$  and *card*  $R$  are, respectively, the number of data and unknowns of the image reconstruction problem.

Finally, we denote by  $H$  the matrix describing the effect of the optical components and, possibly, of the atmosphere. In most applications it can be a Toeplitz or a circulant matrix, i.e.  $H_{i,j} = H_{i-j}$ . We assume that it satisfies the following conditions:

$$H_{i,j} \geq 0; \quad \sum_{i \in S} H_{i,j} > 0, \quad \forall j \in R; \quad \sum_{j \in R} H_{i,j} > 0, \quad \forall i \in S. \quad (1)$$

In other words, we assume that each row or column contains at least one nonzero element.

Then the model of  $y_i$ , discussed in [24], is

$$y_i = y_i^{(\text{obj})} + y_i^{(\text{back})} + y_i^{(\text{ron})}, \quad (2)$$

where  $y_i^{(\text{obj})}$  is the number of photoelectrons arising from the object radiation and is described by a Poisson process with expected value  $(Hx)_i$ ;  $y_i^{(\text{back})}$  is the number of background photoelectrons (including external and internal background, dark current and bias) and is also described by a Poisson process with expected value  $b_i$ ; and  $y_i^{(\text{ron})}$  is the so-called amplifier read-out noise, described by an additive Gaussian process, with the expected value  $r$  and variance  $\sigma^2$ . For simplicity, we assume  $r = 0$  since it is always possible to shift the data by a suitable constant (without changing their statistics) to satisfy this condition.

According to the previous model, by assuming statistical independence between different pixels,  $y$  is a realization of a random vector  $Y$ , with probability density given by the convolution of a product of *card*  $S$  independent Poisson distributions, with expected values  $(Hx+b)_i$ ,  $i \in S$ , with a product of *card*  $S$  independent Gaussian processes, all with expected value 0 and variance  $\sigma^2$ . Thanks to the distributivity of convolution over separable variable functions, the probability density of  $Y$  is given by (the object  $x$  is treated as a set of unknown parameters while  $b$  and  $\sigma^2$  are assumed to be known)

$$p_Y(y; x) = \prod_{i \in S} \left( \sum_{n=0}^{+\infty} \frac{e^{-(Hx+b)_i} (Hx+b)_i^n}{n!} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(y_i-n)^2} \right). \quad (3)$$

Then, the likelihood is the function of  $x$  obtained by letting  $y$  be the detected image,  $L_y^Y(x) = p_Y(y; x)$ , and the ML approach maximizes this function of  $x$ . As usual, the

maximization problem can be equivalently stated as a minimization problem by considering the negative log (neglog) of the likelihood. More precisely, we set

$$\begin{aligned}
 J(x) &= -\ln L_y^Y(x) - \frac{\text{card } S}{2} \ln(2\pi\sigma^2) \\
 &= -\sum_{i \in S} \ln \left\{ \sum_{n=0}^{+\infty} \frac{e^{-(Hx+b)_i} (Hx+b)_i^n}{n!} e^{-\frac{1}{2\sigma^2}(n-y_i)^2} \right\} \\
 &= \sum_{i \in S} \left\{ (Hx+b)_i - \ln \sum_{n=0}^{+\infty} \frac{(Hx+b)_i^n}{n!} e^{-\frac{1}{2\sigma^2}(n-y_i)^2} \right\}, \quad (4)
 \end{aligned}$$

the domain of the functional being the convex and closed cone of the non-negative vectors.

An iterative algorithm for the solution of the problem is derived in [24] using the expectation-maximization (EM) method of Dempster *et al* [8] for the solution of ML problems. Regularized versions of the same algorithm are proposed and tested in [15].

As far as we know, the problem of the existence of solutions of this ML problem and the convergence of the iterative algorithm proposed in [16, 24] has not yet been investigated. This paper makes some progress in this direction. Section 2 proves the existence of solutions by investigating the properties of the functional  $J(x)$ . Section 3 shows that the iterative algorithm proposed in [16, 24] is a scaled gradient method for the constrained minimization of  $J(x)$  and that, if it is convergent, then the limit is a solution of the constrained ML problem. Moreover, we find an approximation that greatly reduces the computational cost of the algorithm and that, even if derived by means of asymptotic expansions, can be applied in all practical cases.

Section 4 considers the specific problem of image deconvolution. We modify the algorithm to compensate for possible boundary effects; moreover, we extend the method to the problem of multiple image deconvolution. Section 5 discusses the implementation of the method and its validation on a few numerical examples. Section 6 makes comments and concluding remarks.

## 2. The existence of solutions of the ML problem

We first prove the concavity of the likelihood function  $L_y^Y(x)$ , which is a direct consequence of the following simple lemma for which no reference could be found in the classical estimation theory literature.

**Lemma 1.** *Consider the data model  $Y = Z + E$ , where  $Z$  and  $E$  are two independent random vectors:  $Z$  denotes the ‘signal of interest’, which depends on the unknown parameters  $x$  and  $E$  denotes an additive noise with known parameters. If the likelihood function of the model  $Z$  is concave (resp. strictly concave), then the likelihood of the noisy model  $Y$  is also concave (resp. strictly concave).*

**Proof.** Define  $p_Z(z; x)$  and  $p_E(e)$  to be the probability density functions (pdf) of  $Z \in \mathcal{Z}$  (support of  $Z$ ) and  $E$ , respectively. Since the two vectors are independent, the pdf of  $Y$  is

$$p_Y(y; x) = \int p_Z(u; x) p_E(y - u) du. \quad (5)$$

Consequently, the likelihoods  $L_y^Y(x)$  and  $L_y^Z(x)$  of the ‘noisy’ and ‘noiseless’ model for a given observed vector  $y$  are related by

$$L_y^Y(x) = \int L_u^Z(x) p_E(y - u) du. \quad (6)$$

As long as  $p_E(e) \geq 0$ , it can be easily checked that if, for all  $u, x_1, x_2$  and  $t \in (0, 1)$ ,  $L_u^Z(tx_1 + (1-t)x_2) \geq tL_u^Z(x_1) + (1-t)L_u^Z(x_2)$ , equation (6) implies that the same result holds for  $L_y^Y(x)$ .

Assume now that, for all  $u$ ,  $L_u^Z(x)$  is strictly concave. If  $E$  is continuous, there necessarily exists a nonempty open subset  $\mathcal{U} \subset \mathcal{Z}$  where, for all  $u \in \mathcal{U}$ ,  $p_E(y - u) > 0$  and  $L_y^Y(x) = \int_{\mathcal{U}} L_u^Z(x) p_E(y - u) du$ . Reasoning analogous to that given above proves that the strict concavity of  $L_u^Z(x)$  implies the strict concavity of  $L_y^Y(x)$ .  $\square$

**Proposition 1.** *The function  $J(x)$ , defined in equation (4), is convex on the closed and convex set  $\mathcal{C}$  of the non-negative vectors. It is strictly convex if the equation  $Hx = 0$  has the unique solution  $x = 0$ .*

**Proof.** Lemma 1 relates the concavity of  $L_y^Y(x)$  to the concavity of  $L_y^Z(x)$ , or equivalently the convexity of  $J(x)$  to the convexity of the neglog likelihood of the signal of interest  $Z$ . The proof then follows from standard results of image reconstruction for Poisson noise, which state that the corresponding neglog likelihood is convex and strictly convex if the null space of  $H$  contains only the null element [22].  $\square$

**Proposition 2.** *The function  $J(x)$  is non-negative and locally bounded on  $\mathcal{C}$ . Therefore, it has minima and all minima are global. The minimum is unique if the equation  $Hx = 0$  has the unique solution  $x = 0$ .*

**Proof.** The non-negativity follows from the second expression in equation (4) since the series is bounded by 1. The local boundedness of the function follows from the convergence of the series for any  $x \in \mathcal{C}$ . Then, the existence of global minima follows from the previous proposition.  $\square$

The previous results demonstrate the existence of solutions of the ML-problem. It is possible to gain more insight into the properties of the functional  $J(x)$  and hence of the minimization algorithm investigated in the following section by introducing the following functions:

$$p(s; t) = \sum_{n=0}^{+\infty} \frac{s^n}{n!} e^{-\frac{1}{2\sigma^2}(n-t)^2} \tag{7}$$

$$q(s; t) = \sum_{n=1}^{+\infty} \frac{s^{n-1}}{(n-1)!} e^{-\frac{1}{2\sigma^2}(n-t)^2} = \sum_{n=0}^{+\infty} \frac{s^n}{n!} e^{-\frac{1}{2\sigma^2}(n+1-t)^2} \tag{8}$$

$$r(s; t) = \sum_{n=2}^{+\infty} \frac{s^{n-2}}{(n-2)!} e^{-\frac{1}{2\sigma^2}(n-t)^2} = \sum_{n=1}^{+\infty} \frac{s^{n-1}}{(n-1)!} e^{-\frac{1}{2\sigma^2}(n+1-t)^2}, \tag{9}$$

which are defined for any non-negative value of  $s$  and any real value of  $t$ . The functions  $q(s; t)$  and  $r(s; t)$  are, respectively, the first and second partial derivative of  $p(s; t)$  with respect to  $s$ .

If we introduce the vectors  $p(x)$ ,  $q(x)$  and  $r(x)$ , whose components are given by

$$\begin{aligned} p_i(x) &= p((Hx + b)_i, y_i), & q_i(x) &= q((Hx + b)_i, y_i), \\ r_i(x) &= r((Hx + b)_i, y_i); & i &\in \mathcal{S}, \end{aligned} \tag{10}$$

and the weighting vector  $h > 0$  with components

$$h_j = \sum_{i \in \mathcal{S}} H_{i,j}; \quad j \in \mathcal{R}, \tag{11}$$

then, the gradient and the Hessian of  $J(x)$  are given by

$$\begin{aligned} \nabla J(x) &= h - H^T \frac{q(x)}{p(x)}, & \nabla^2 J(x) &= H^T D(x) H, \\ D(x) &= \text{diag} \left\{ \frac{q^2(x) - p(x)r(x)}{p^2(x)} \right\}, \end{aligned} \quad (12)$$

where the product and quotient of two vectors are defined component by component (Hadamard product and quotient of two vectors).

The previous results on the convexity of  $J(x)$  imply that the quotient appearing in the expression of the Hessian is non-negative. In the following lemma we prove that it is always positive, thus proving that convexity and strict convexity depend only on the properties of the imaging matrix  $H$ .

**Lemma 2.** For any  $t$  and  $s > 0$ , we have

$$q^2(s; t) - p(s; t)r(s; t) > 0, \quad (13)$$

where the functions  $p, q, r$  are defined in equations (7)–(9).

**Proof.** We set

$$\xi_t(n) = \exp \left[ -\frac{1}{2\sigma^2} (n-t)^2 \right], \quad (14)$$

and observe that, for any  $t$ ,

$$\xi_t(n+1)\xi_t(m) - \xi_t(n)\xi_t(m+1) > 0, \quad m > n. \quad (15)$$

Then, if we write  $q^2(s; t)$  as the product of the two (equivalent) series given in equation (8), use the expression of  $p(s; t)$  given in equation (7), and express  $r(s; t)$  as the second series given in equation (9), we get

$$q^2(s; t) - p(s; t)r(s; t) = \sum_{m=0}^{+\infty} \sum_{n=0}^{+\infty} \frac{s^{m+n-1}}{m!n!} m [\xi_t(n+1)\xi_t(m) - \xi_t(n)\xi_t(m+1)]. \quad (16)$$

If we split the sum with respect to  $n$  into a sum from 0 to  $m$  and a sum from  $m$  to  $+\infty$  (note that the terms with  $n = m$  are zero), then, by exchanging indices in the second sum and collecting the two terms, we have

$$q^2(s; t) - p(s; t)r(s; t) = \sum_{m=0}^{+\infty} \sum_{n=0}^m \frac{s^{m+n-1}}{m!n!} (m-n) [\xi_t(n+1)\xi_t(m) - \xi_t(n)\xi_t(m+1)], \quad (17)$$

and therefore the series is positive, because of inequality (15).  $\square$

**Proposition 3.** The function  $J(x)$  is strictly convex if and only if the equation  $Hx = 0$  has the unique solution  $x = 0$ .

**Proof.** The proof follows from the previous lemma and equation (12).  $\square$

**Remark.** The existence of solutions of the ML problem does not imply that we have obtained sensible estimates of the unknown object. It should also be proved that these solutions are stable with respect to noise fluctuations. We have evidence that this property does not hold true, as follows from the analysis of two related noise models. It is well known that the least-square solutions, coinciding with the ML solutions in the case of additive Gaussian noise, are widely oscillating as a consequence of noise propagation. This effect is partially reduced, but not suppressed, if one introduces the additional constraints of non-negativity and flux

conservation (constraint on the 1-norm of the solution). These constraints are automatically satisfied by the solutions of the ML problem in the case of Poisson noise and zero background [22]. However, these solutions are affected by the so-called checkerboard effect which is a consequence of noise propagation [19]: they are zero in a large number of pixels and grow up in the others. The noise model considered in this paper is intermediate between the previous ones and therefore we expect that the ML solutions have a similar behavior. This conjecture is supported by numerical results obtained with the iterative method investigated in the following section. Since the ML problem is ill-posed, the quite natural remedy should be to introduce *regularization*, through penalties, Bayesian priors or other approaches (for a discussion, in the case of Poisson plus Gaussian noise, see [24]). Such an analysis is beyond the scope of this paper and is considered, for instance, in [15].

### 3. The iterative algorithm

In the model of image formation defined by equation (2), the values of an image are not necessarily positive; negative values can occur, as a consequence of the additive Gaussian noise when the background is zero or sufficiently small. However, the goal of image reconstruction is to produce a non-negative estimate of the unknown object. Therefore, as we already remarked, the problem is the minimization of  $J(x)$  on the closed and convex cone  $\mathcal{C}$  of the non-negative vectors. Thanks to the results of the previous section this problem has a solution. The minimum occurs at  $x = 0$  if the following condition is satisfied (see equation (12)):

$$H^T \frac{q(0)}{p(0)} < h. \quad (18)$$

However, non-trivial minima should occur if an object is present in the field of view of the imaging system.

If  $x^*$  is one of these minima, it satisfies the Karush–Kuhn–Tucker (KKT) conditions that, in this particular case, can be written as

$$x^* \left( h - H^T \frac{q(x^*)}{p(x^*)} \right) = 0 \quad (19)$$

$$x^* \geq 0, \quad h - H^T \frac{q(x^*)}{p(x^*)} \geq 0. \quad (20)$$

If we introduce the nonlinear operator

$$T(x) = \frac{x}{h} H^T \frac{q(x)}{p(x)}, \quad (21)$$

then, the first KKT-condition, equation (19), shows that each minimum point  $x^*$  is a fixed point of the operator  $T$ ,

$$x^* = T(x^*). \quad (22)$$

By applying the method of successive approximations, we obtain the algorithm proposed by Llacer and Nuñez [16] as well as by Snyder *et al* [24],

- choose  $x^{(0)} > 0$ ;
- given  $x^{(k)}$ , compute

$$x^{(k+1)} = \frac{x^{(k)}}{h} H^T \frac{q(x^{(k)})}{p(x^{(k)})}. \quad (23)$$

Since the operator  $T$  is not a contraction, the convergence of the algorithm cannot be deduced from general theorems of fixed point theory. As far as we know, the convergence of the method is an open problem.

The method elegantly handles the non-negativity constraint of the problem. More precisely, since the quotient  $q/p$  is strictly positive and the matrix  $H$  satisfies the conditions of equation (1), it is easy to prove, by induction, that, if  $x^{(0)} > 0$ , then each iterate  $x^{(k)}$  is strictly positive. Of course, this remark does not imply that the limit, if it exists, does not have zero components.

In addition, if we write the iteration in the following form:

$$x^{(k+1)} = x^{(k)} + \frac{x^{(k)}}{h} \left( H^T \frac{q(x^{(k)})}{p(x^{(k)})} - h \right), \quad (24)$$

it becomes evident that the method is just a scaled gradient method (with step one in the descent direction), the scale factor being the distance of each component to the non-negativity constraint. We point out that these properties are common to two other iterative methods used in image reconstruction: the iterative space reconstruction algorithm (ISRA) [7, 9] and the Richardson–Lucy method (RLM) [17, 21, 22], which apply to the constrained maximization of the likelihood function, respectively, in the case of Gaussian and Poisson noise.

The following partial result can be easily proved.

**Proposition 4.** *If the sequence of the iterates is convergent, then the limit satisfies the KKT-conditions and therefore is a constrained minimum of the functional  $J(x)$ .*

**Proof.** Let us denote by  $x^*$  the limit of the sequence  $\{x^{(k)}\}_{k=1}^{\infty}$ . Since the operator  $T$  is continuous, it is evident that the limit satisfies the first KKT-condition. Moreover, since all the iterates are strictly positive, as we already remarked, it also follows that the limit is non-negative. Therefore, we have only to prove that the third KKT-condition is satisfied. This follows from the previous result in the pixels where  $x_j^* > 0$ , so we must only consider the pixels where  $x_j^* = 0$ .

Let us assume that the condition is not satisfied in one of these pixels, i.e. we assume that, for a given  $j$ ,

$$x_j^* = 0, \quad \frac{1}{h_j} \left( H^T \frac{q(x^*)}{p(x^*)} \right)_j > 1. \quad (25)$$

It follows that there exists  $k_0$  such that, for any  $k \geq k_0$ ,

$$\frac{1}{h_j} \left( H^T \frac{q(x^{(k)})}{p(x^{(k)})} \right)_j > 1. \quad (26)$$

Then, since all the iterates are strictly positive, it also follows that, for all these values of  $k$ ,  $x_j^{(k+1)} > x_j^{(k)}$ , in contradiction with the assumption that the limit is zero.  $\square$

**Remark.** In all our numerical experiments we always found convergence of the algorithm, and, thanks to the previous result, we can conclude that the limit is a ML solution. However, we also know that these solutions are corrupted by noise propagation so that they are not reliable estimates of the unknown object. An important feature shared by several iterative methods converging to ML solutions corresponding to different noise models, such as Landweber, conjugate gradient, projected Landweber, ISRA, RLM (just to mention some of them) is the so-called semiconvergence phenomenon [11, 18]: the iterative method provides a satisfactory result after a certain number of steps but the iterates deteriorate when the iteration goes on. A number of examples of this behavior is discussed in [2]. In other words one obtains

regularization through early stopping of the iterations. Rigorous mathematical analysis in the case of Landweber iterations and conjugate gradient is given in [11]. Our numerical experience demonstrates that also the algorithm (23) has the semiconvergence property. Therefore, early stopping of the iterations can be an easy and simple way for obtaining ‘regularized’ solutions. This conclusion does not mean that it is not necessary to consider more refined regularization methods, as we discussed at the end of section 2.

In many cases, the best result can only be obtained with a large number of iterations. Since the cost of one iteration in equation (23) is high (one has to compute the quotient of two series), the computational burden can become excessive. For this reason, it is important to have an approximate but sufficiently accurate estimate of the quotient of equation (23). Two different approaches were proposed [23, 25]. The first consists in adding  $\sigma^2$  to the data, so that the new data are affected by additive Gaussian noise with expected value  $\sigma^2$ , and in approximating the Gaussian distribution with a Poisson distribution. The final result consists in modifying the data and background terms so that the iterative algorithm is basically the RLM with the above-mentioned modifications

$$x^{(k+1)} = \frac{x^{(k)}}{h} H^T \frac{y + \sigma^2}{Hx^{(k)} + b + \sigma^2}. \quad (27)$$

The second approach is based on a saddle-point approximation of the mixed Gauss–Poisson distribution. Its implementation is not as simple as the Poisson approximation, and it is also computationally more expensive. Moreover, a numerical validation shows that, for an increasing number of iterations, the two approximations provide essentially the same results [25].

We propose a different approximation that, in principle, applies only to the pixels where  $y_i$  is sufficiently large (for instance,  $y_i > 30$ ). It is based on approximating the Poisson distribution with a Gaussian one and, although its derivation is lengthy (see the appendix), the final result is quite simple. Moreover, a numerical implementation has demonstrated that it works well also for small and negative values of  $y_i$  and that it is more accurate than the approximation provided by equation (27). This point will be discussed in section 5.

The approximation is obtained from asymptotic approximations of the functions  $p(s; t)$ ,  $q(s; t)$ , defined in equations (7)–(8), for large values of  $t$ ,  $s$  satisfying the condition  $|s - t| \leq c\sqrt{t}$ ; these are derived in the appendix. From equations (A.17) and (A.21) we have

$$\frac{q(s; t)}{p(s; t)} = \exp\left[-\frac{1 + 2(s - t)}{2(s + \sigma^2)}\right] \left\{1 + O\left(\frac{1}{\sqrt{t}}\right)\right\}. \quad (28)$$

Then, from equation (10), in a pixel where  $y_i$  is large, we get

$$\frac{q_i(x)}{p_i(x)} = \exp\left[-\frac{1 + 2((Hx + b)_i - y_i)}{2((Hx + b)_i + \sigma^2)}\right] \left\{1 + O\left(\frac{1}{\sqrt{y_i}}\right)\right\}. \quad (29)$$

The unexpected and interesting result provided by our numerical experiments is that the approximation provided by this equation can be used everywhere, thus reducing the computational burden of the method in a significant way. Indeed, in such a case, one iteration has approximately the same computational cost of one RLM iteration, as defined in equation (27).

Let us also remark that, if  $(Hx + b)_i$  and  $y_i$  are large, then, by taking into account that the exponent in equation (29) is of the order of  $y_i^{-1/2}$ , the first-order Taylor expansion of the exponential provides

$$\exp\left[-\frac{1 + 2((Hx + b)_i - y_i)}{2((Hx + b)_i + \sigma^2)}\right] \simeq \frac{y_i + \sigma^2}{(Hx^{(k)} + b)_i + \sigma^2}, \quad (30)$$

and therefore the two approximations coincide.

#### 4. Application to image deconvolution

A quite natural application of the previous approach is the deconvolution of 3D images in microscopy or the deconvolution of 2D images in astronomy. In these applications, it is usually assumed that  $R = S$  and that the imaging matrix  $H$  is defined by a convolution product

$$Hx = K * x, \quad (31)$$

where  $K$  is the so-called point spread function (PSF) of the imaging system. If the object  $x$  is surrounded by a uniform background and is completely contained within the field of view (FoV) of the imaging system, then a useful approximation consists in extending image, PSF and object periodically outside the FoV (periodic boundary conditions), so that the convolution product can be easily and efficiently computed by means of the FFT algorithm. In such a case it is also assumed that the PSF is normalized in the following way:

$$\sum_{i \in S} K_i = 1, \quad (32)$$

so that we have  $h_i = 1$  (see equation (11)). This condition implies that

$$\sum_{i \in S} (Hx)_i = \sum_{j \in S} x_j. \quad (33)$$

The physical interpretation is that the total number of photons in the object (also called total flux) coincides with the total number of photons in the computed image.

Finally, we point out that one iteration of the algorithm defined by equation (23), with the quotient computed by means of the leading term of the asymptotic expansion of equation (29), requires the computation of four FFTs, and therefore has approximately the same computational cost of one RLM iteration.

##### 4.1. Boundary effect correction

If the object to be imaged extends outside the FoV, then the periodic boundary conditions cannot be used. Indeed they introduce discontinuities at the boundary that produce Gibbs artifacts (also called ripples) in the deconvolution process. These can propagate inside the image domain, completely degrading the quality of the reconstructed image.

To avoid this difficulty, the use of boundary conditions (Dirichlet, reflective, anti-reflective) has been proposed (for a recent paper, see [10]). This approach is equivalent to extending the image outside the FoV assuring, in some cases, continuity of the extrapolated image and of its gradient. In addition, it can lead to fast deconvolution algorithms. However, the extension of the image is, to a large extent, arbitrary; hence, we prefer an approach that has been proposed for RLM [4], but has wider applicability [1, 27] and consists of reconstructing the object in a domain broader than the FoV. A similar approach, derived from ideas proposed in [20], was proposed for the restoration of the images of the Hubble Space Telescope, and implemented in the *lucy task* of STSDAS [26, 28], even if a complete description of the implementation details is not provided in the refereed literature. Moreover, the modification of RLM implemented in STSDAS can also compensate for flat field and bad pixels by the use of a suitable mask.

If we wish to reconstruct the object in a domain  $R$  that is broader than  $S$ ,  $R \supset S$ , at first glance it is not possible to use the FFT for the computation of  $H$  and  $H^T$ , and therefore the computational cost is excessive. However, let us consider a domain  $D$  broader than  $R$ ,  $D \supset R \supset S$ , and such that  $Hx$  is negligible at its boundary. We extend  $x$  from  $R$  to  $D$  and  $y$  from  $S$  to  $D$  by zero padding. Moreover, we assume that the PSF is defined on  $D$  and normalized to 1 on  $D$ ,

$$\sum_{i \in D} K_i = 1. \quad (34)$$

This assumption is not unrealistic. The extension of the PSF can be obtained by zero padding from that defined on  $S$  or, in the case of astronomical images, it can be obtained from the image of a star in the FoV, by the process of extraction and extrapolation described in [13]. We remark that, as a consequence of the high dynamic range of astronomical images, their deconvolution requires an estimate of the small values of the PSF. In the case of adaptive optics, a Gaussian approximation for the tail of the PSF is not satisfactory.

Next, we assume a periodic extension of  $x$ ,  $y$  and  $K$  outside  $D$ , and we choose the size  $N'$  of  $D$  in such a way that the use of the FFT is possible, i.e.  $N'$  can be a power of 2 or 3 or 5. Then  $Hx$  and  $H^T y$  can be efficiently and correctly computed by means of the FFT. For instance, if  $S$  is  $N \times N$ , and  $N' = 2N$ , then the increase in the computational cost is about by a factor of 4 and therefore can be acceptable.

If we look at the iterative algorithm of equation (23), we see that we can compute efficiently the rhs. Indeed, we can extend the quotient from  $S$  to  $D$  by zero padding to get the following array:

$$C_i(x^{(k)}) = \begin{cases} \frac{q_i(x^{(k)})}{p_i(x^{(k)})}, & \text{if } i \in S, \\ 0, & \text{otherwise,} \end{cases} \quad (35)$$

and we can also assume a periodic extension of this array outside  $D$ . Then  $H^T C(x^{(k)})$  can be computed efficiently by means of the FFT.

The last point is the computation of the weighting array  $h$ . From equation (11) we have

$$h_j = \sum_{i \in S} K_{i-j} = (K^T * M_S)_j, \quad j \in D, \quad (36)$$

where  $M_S$  is the characteristic function of the set  $S$ , i.e. it is 1 over  $S$  and 0 outside. It follows that  $h_j$  can also be computed efficiently by means of the FFT. We remark that, thanks to the normalization condition (34) of the PSF,  $h_j$  is approximately 1 over  $S$  and decreases outside  $S$ . Therefore, if we introduce the windowing array  $w$  defined by

$$w_j = \begin{cases} \frac{1}{h_j}, & \text{if } j \in R, \\ 0, & \text{otherwise,} \end{cases} \quad (37)$$

we can write the algorithm of equation (23) in the following form:

$$x^{(k+1)} = w x^{(k)} H^T C(x^{(k)}), \quad (38)$$

and all the iterates are automatically 0 outside  $R$ .

We conclude by remarking that the reconstruction domain  $R$  can be defined as the set of index values corresponding to components of the weighting vector/array that are greater than some threshold  $\eta$ , i.e.  $R = \{j | h_j > \eta\}$ .

#### 4.2. Multiple-image deconvolution

The results obtained in the case of single-image deconvolution can be easily extended to the case of multiple-image deconvolution. The application we have in mind is to the reconstruction of the images that will be provided by a new interferometer, called LINC-NIRVANA (<http://www.mpia.de/LINC/index.html>), developed by a German–Italian collaboration for the Large Binocular Telescope (LBT). This instrument will provide different interferometric images of the same astronomical target and will require routine use of image reconstruction

methods for obtaining a single high-resolution image from the different raw images. A recent survey of the results obtained by our group is given in [5].

Assume we have  $M$  images of the same object  $x$ , and denote these images as well as the corresponding PSFs and backgrounds with an index  $m$  (written in roman font):  $y_m, K_m, b_m$ , ( $m = 1, \dots, M$ ), so that their components will be denoted by two indices  $y_{m,i}, K_{m,i}, b_{m,i}$ , with  $i \in S$ . As in the single-image case, the expected value of  $y_m$  is given by  $K_m * x + b_m$ . Moreover, we let  $J_m(x)$  denote the functional of equation (4) with  $y, H$  replaced by  $y_m, H_m$ , where  $H_m$  is the matrix corresponding to the PSF  $K_m$ . Since all the images are statistically independent, the functional will be given by

$$J(x) = \sum_{m=1}^M J_m(x). \quad (39)$$

The extension of the iterative algorithm of equation (23) is quite easy. Indeed, it is replaced by

$$x^{(k+1)} = \frac{x^{(k)}}{M} \sum_{m=1}^M H_m^T \frac{q_m(x^{(k)})}{p_m(x^{(k)})}, \quad (40)$$

where, for simplicity, we assumed that all the PSFs satisfy the normalization condition of equation (32).

The computational cost of this algorithm can be reduced by applying the acceleration method proposed by Hudson and Larkin [12] for emission tomography, based on ordered subsets of projections. In the application to astronomical imaging, the tomographic projections are replaced by the different images of the same target and, since their number is in general small, the subsets can consist of only one image. Therefore, as proposed in [3], one iteration of equation (40) is replaced by a cycle over the  $M$  images, so that the algorithm is as follows:

- given  $x^{(k)}$ , set  $h^{(0)} = x^{(k)}$  and, for  $m = 1, \dots, M$ , compute

$$h^{(m)} = h^{(m-1)} H_m^T \frac{q_{m-1}(h^{(m-1)})}{p_{m-1}(h^{(m-1)})}; \quad (41)$$

- set  $h^{(M)} = x^{(k+1)}$ .

Such an algorithm reduces the number of iterations required by about a factor of  $M$  with a computational cost per iteration approximately coinciding with that of equation (40). Therefore, the reduction in computation time is approximately by a factor of  $M$ . A more precise estimate is given in [3].

## 5. Computer implementation

The iterative algorithms discussed in section 3 have the following general structure:

$$x^{(k+1)} = \frac{x^{(k)}}{h} H^T C(x^{(k)}), \quad (42)$$

where  $C(x)$  is a positive vector/array that takes one of the following possible forms:

- 

$$C_i(x) = \frac{\sum_{n=0}^{+\infty} \frac{(Hx+b)_i^n}{n!} e^{-\frac{1}{2\sigma^2}(n+1-y_i)^2}}{\sum_{n=0}^{+\infty} \frac{(Hx+b)_i^n}{n!} e^{-\frac{1}{2\sigma^2}(n-y_i)^2}}; \quad (43)$$

- 

$$C_i^{(1)}(x) = \exp\left[-\frac{1 + 2((Hx+b)_i - y_i)}{2((Hx+b)_i + \sigma^2)}\right]; \quad (44)$$

•

$$C_i^{(2)}(x) = \frac{y_i + \sigma^2}{(Hx + b)_i + \sigma^2}. \quad (45)$$

with  $C_i^{(1)}(x)$  and  $C_i^{(2)}(x)$  being the two different approximations of  $C_i(x)$  that will be compared in this section. Concerning the implementation of  $C_i(x)$ , we take into account the strong decay of the Gaussian factor, so the two series are computed using a number of terms given by  $[y_i + 4\sigma]$ . The implementation of the two other algorithms is obvious. To compare the three algorithms, we select examples where the condition  $y_i \leq 30$  is satisfied in the vast majority of pixels.

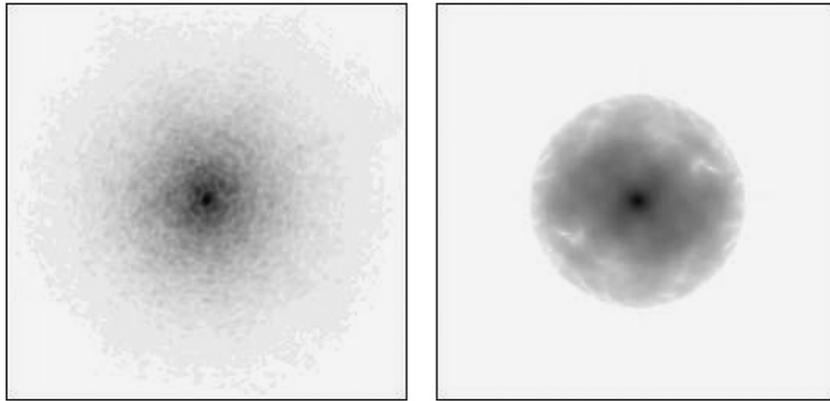
We performed our numerical experiments both in the case of single image deconvolution and in the case of multiple image deconvolution, the latter case being considered mainly in view of possible application to the deconvolution of the images from the LINC-NIRVANA interferometer of LBT. Since we obtained similar results in both cases we report only those obtained in the first one. We also considered several different astronomical objects and several different PSFs. We show only two among the many examples we have investigated. Indeed, the conclusions that can be derived from these two examples can be applied to all others. All our experiments are performed using the software package AIRY (Astronomical Image Restoration in interferometrY, <http://dirac disi.unige.it/>) [6], which has been developed for the simulation and successive deconvolution of LINC-NIRVANA images. Obviously, it can also be used for images of a single-mirror telescope.

First of all, we point out that our numerical experiments do not intend to simulate real astronomical observations but only to check and compare the numerical accuracy of the different approximations mentioned above. For this reason we investigate neither the effect of the background, which is relevant in infrared observations, nor the effect of the analog-to-digital conversion. As a consequence we consider images which can take negative values.

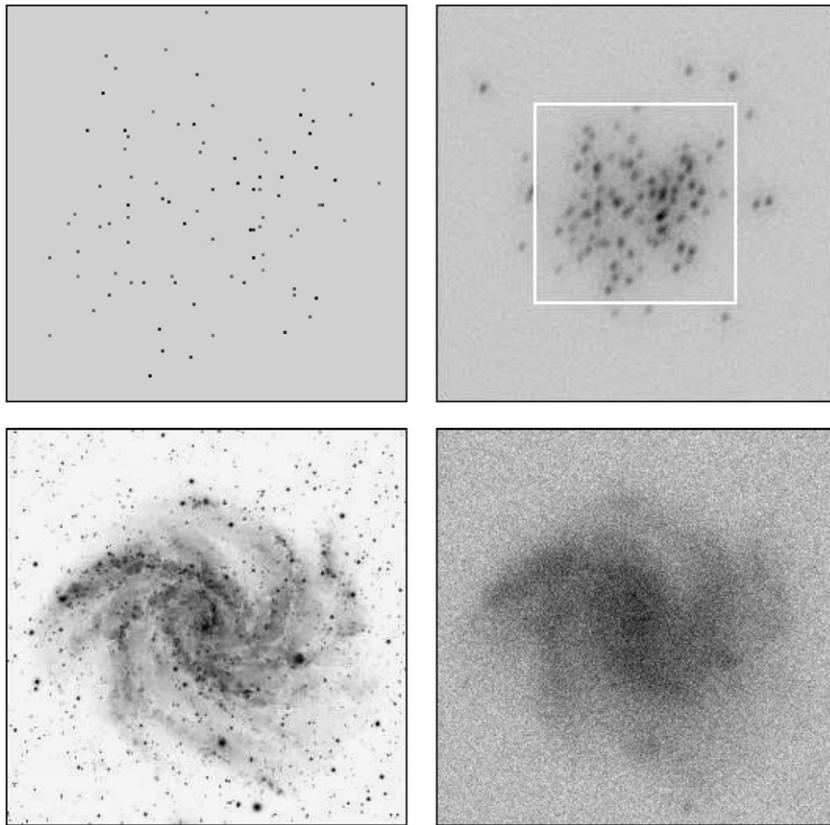
The PSF used in our experiments has been generated using the software package CAOS (code for adaptive optics systems, <http://www-luan.unice.fr/caos/>), and therefore is a simulation of an AO PSF, with a Strehl ratio of about 20%. The picture of this PSF,  $256 \times 256$ , is given in figure 1 (left panel) together with the corresponding modulation transfer function (MTF), i.e. the modulus of the Fourier transform of the PSF. Using a suitable gray scale, the circular bandwidth arising from the geometry of the mirror is visible, even if the high-frequency components are attenuated as a consequence of the low AO correction we are assuming.

The images are obtained by convolving the simulated object with this PSF. No background is added. Next, they are perturbed with Poisson noise (from photon counting) and additive Gaussian noise with  $\sigma = 10$ , a value that is reasonable for the read-out noise of existing scientific cameras; it could take higher values in real images if they are obtained by a co-adding of several frames.

We report the results obtained for two different kinds of astronomical objects: the first is a star cluster consisting of 100 stars, with an angular distribution, with respect to the centroid of the cluster, described by a Gaussian function; the second is the image of the galaxy NGC6946, as provided by the prime-focus camera of LBT ([http://medusa.as.arizona.edu/lbto/astronomical\\_images.htm](http://medusa.as.arizona.edu/lbto/astronomical_images.htm)), and reduced in size to a  $256 \times 256$  array. The integrated magnitude of the star cluster is set to 12.5, with magnitudes of the stars ranging approximately from 17 to 18. The numerical values of the object range from 0 to  $1.80 \times 10^4$ . The integrated magnitude of the galaxy is also set to 12.5, but the numerical values of the object range from 0 to 120.



**Figure 1.** Left panel: the PSF used in our simulation (log scale). Right panel: the corresponding MTF (0.25 power scale).



**Figure 2.** Upper panels: the  $128 \times 128$  central part of the star cluster (left) and the  $256 \times 256$  blurry image of the full cluster (right); the white square indicates the central part corresponding to the left panel. Lower panels: the galaxy NGC6946 (left) and the corresponding blurry image (right).

Figure 2 shows the two objects with the corresponding images. In the case of the star cluster, the image takes negative values in 31% of the pixels and values smaller than 30 in

**Table 1.** Mean value (MV) and standard deviation (SD) of the C-arrays, in the case of the star cluster, for different numbers of iterations.

Number of iterations	Parameters	$C$	$C^{(1)}$	$C^{(2)}$
100	MV	0.9903	0.9903	0.9885
	SD	0.0956	0.0956	0.0962
1000	MV	0.9947	0.9947	0.9947
	SD	0.0935	0.0935	0.0939
10 000	MV	0.9944	0.9943	0.9944
	SD	0.0931	0.0931	0.0935

**Table 2.** Mean value (MV) and standard deviation (SD) of the differences  $C - C^{(1)}$  and  $C - C^{(2)}$ , in the case of the star cluster, for different numbers of iterations.

Number of iterations	Parameters	$C - C^{(1)}$	$C - C^{(2)}$
100	MV	$-2.80 \times 10^{-5}$	$1.78 \times 10^{-3}$
	SD	$1.17 \times 10^{-3}$	$6.21 \times 10^{-3}$
1000	MV	$7.29 \times 10^{-7}$	$-4.05 \times 10^{-5}$
	SD	$7.23 \times 10^{-4}$	$5.89 \times 10^{-3}$
10 000	MV	$3.28 \times 10^{-6}$	$-2.32 \times 10^{-5}$
	SD	$6.95 \times 10^{-4}$	$5.85 \times 10^{-3}$

83% of the pixels. Moreover, the minimum and maximum values are  $-35.4$  (about  $3\sigma$ ) and  $457.0$ . For the galaxy, the corresponding percentages are 21% and 75%, respectively, and the minimum and maximum values are  $-34.5$  and  $113.7$ . Therefore, in the vast majority of the pixels, conditions for the validity of the asymptotic approximation derived in the appendix are not satisfied.

The algorithm behaves differently in the two cases; the convergence is slow in the case of the star cluster, while it is much faster in the case of the galaxy (a diffuse object), a behavior similar to that of the RLM algorithm. Indeed, the three different approximations of the array  $C(x)$  do not lead to different convergence behavior.

We first analyze and compare the behavior of the arrays of equations (43)–(45) as a function of the number of iterations. In the case of the cluster, we computed 10 000 iterations, and we checked the results at 100, 1000 and 10 000 iterations. Table 1 gives, for these iterations, the mean values and the standard deviations of the pixel values of each one of the three arrays. They take values close to 1, with deviations of the order of 10%. No significant difference is found between these parameters of the three arrays; also, the increase in the number of iterations does not significantly modify the situation.

Differences appear if we look at the arrays  $C - C^{(1)}$  and  $C - C^{(2)}$ , i.e. the deviations of the approximations with respect to the ‘correct’ array  $C$ . These results, reported in table 2, indicate that the approximation derived in this paper, even if it should be valid only for large values of  $y_i$ , provides more accurate values than the approximation obtained by replacing the Gaussian distribution of the read-out noise with a Poisson distribution; the mean value is smaller by a factor of about 100 for  $k = 100$  and 1000, while the standard deviation is smaller by a factor of about 10 for  $k = 1000$  and 10 000. Note that good reconstructions are already obtained for  $k = 2000$ .

**Table 3.** Mean value (MV) and standard deviation (SD) of the C-arrays, in the case of the galaxy, for different numbers of iterations.

Number of iterations	Parameters	$C$	$C^{(1)}$	$C^{(2)}$
50	MV	0.9957	0.9957	0.9936
	SD	0.0922	0.0924	0.0924
100	MV	0.9975	0.9975	0.9968
	SD	0.0920	0.0920	0.0921

**Table 4.** Mean value (MV) and standard deviation (SD) of the differences  $C - C^{(1)}$  and  $C - C^{(2)}$ , in the case of the galaxy, for different numbers of iterations.

Number iterations	Parameters	$C - C^{(1)}$	$C - C^{(2)}$
50	MV	$2.14 \times 10^{-6}$	$2.19 \times 10^{-3}$
	SD	$8.86 \times 10^{-4}$	$6.39 \times 10^{-3}$
100	MV	$1.23 \times 10^{-6}$	$7.49 \times 10^{-4}$
	SD	$8.54 \times 10^{-4}$	$5.57 \times 10^{-3}$

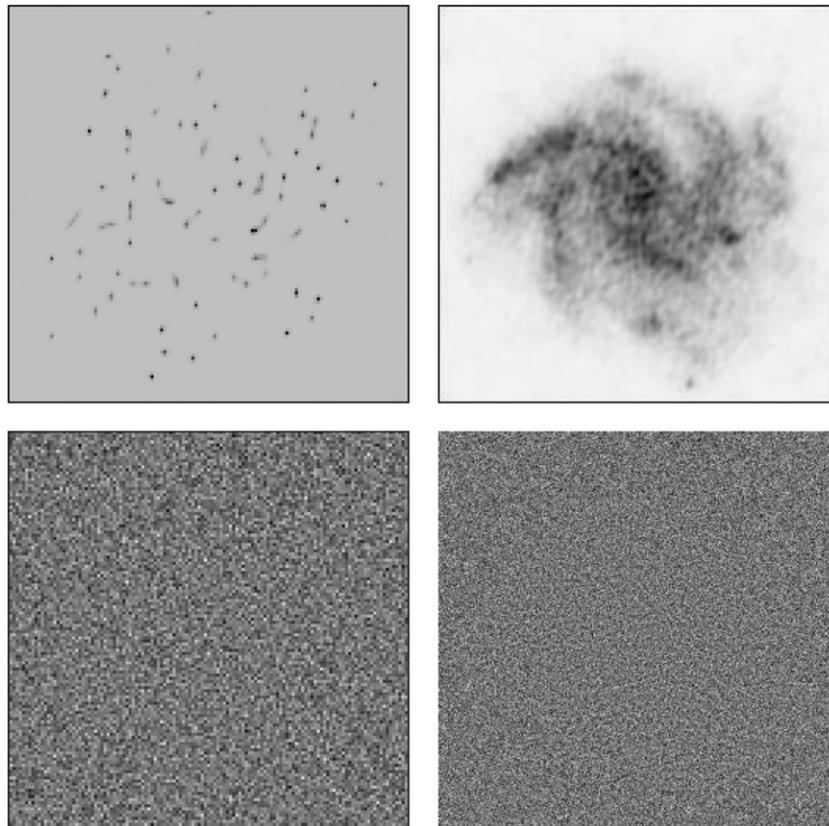
Similar results are obtained in the case of the galaxy, reported in tables 3 and 4. The best restoration is reached after a number of iterations ranging from 40 to 50; therefore, we checked the arrays only at 50 and 100 iterations (at 100 iterations the checkerboard effect is already evident). Again, the approximation proposed in this paper is better than the Poisson approximation.

In the case of the star cluster, we pushed the iterations up to 10 000 but, if we look at the magnitudes (fluxes) of the reconstructed stars, the results do not change after 2000 iterations. If we subtract the reconstruction obtained with the approximation  $C^{(1)}(x)$  from that obtained with  $C(x)$ , we find an array with values ranging from  $-69.9$  to  $42.3$ ; the mean value of the array is  $4.9 \times 10^{-4}$  and the standard deviation 1.4. On the other hand, if we subtract the reconstruction obtained with the approximation provided by  $C^{(2)}(x)$  we obtain an array with values ranging from  $-54.0$  to  $70.9$ ; the mean value is  $-3.4 \times 10^{-3}$  and the standard deviation 1.6. In conclusion, in such a case, the reconstruction provided by  $C^{(1)}$  is comparable with that provided by  $C^{(2)}$ .

The upper-left panel of figure 3 shows the reconstruction of the cluster (represented with a log scale) obtained, after 2000 iterations, with the algorithm implementing the computation of the series. If we compare this result with the left panel of figure 3, we recognize a few artifacts, but the estimation of the magnitudes of the stars is quite satisfactory except in those cases where two stars are too close to be resolved by the algorithm. Moreover, to check the statistical significance of our results, we compute the normalized residuals defined by

$$R_i(\bar{x}) = \frac{y_i - (H\bar{x})_i}{\sqrt{(H\bar{x})_i^2 + \sigma^2}}, \quad (46)$$

where  $\bar{x}$  denotes the reconstructed object. If the reconstructions are statistically correct these arrays should be free of artifacts and their values should have a Gaussian distribution with zero expected value and variance 1. All these conditions are satisfied: the map of the normalized residuals is shown in the lower-left panel of figure 3 (the results are quite similar for the three approximations); moreover, the expected values for the three reconstructions are about  $-0.055$  while the variances are about 0.995.



**Figure 3.** Left: the reconstruction of the central part of the cluster (up), and the map of the corresponding normalized residual (down). Right: the reconstruction of the galaxy (up) and the corresponding residual (down).

In the case of the galaxy, we also check the accuracy of the reconstruction by computing, at each iteration, both the relative r.m.s. error defined by

$$\rho^{(k)} = \frac{\|x - x^{(k)}\|_2}{\|x\|_2}, \quad (47)$$

where  $x$  is the true object and  $\|\cdot\|_2$  denotes the Euclidean norm, and the Kullback–Leibler (KL) divergence (or I-divergence) of  $x^{(k)}$  from  $x$ ,

$$\delta^{(k)} = \sum_{j \in S} \left\{ x_j \ln \frac{x_j}{x_j^{(k)}} + x_j^{(k)} - x_j \right\}. \quad (48)$$

In such a case the convergence is quite fast and, for this reason, we compute only 100 iterations. Indeed, the minimum value of  $\rho^{(k)}$  is reached after 45 iterations for the first two methods and after 51 for the third one. In all cases, the relative r.m.s. error is on the order of 35%. Moreover,  $\delta^{(k)}$  has also a minimum occurring for a bit larger value of  $k$ , ranging from 60 to 70 according to the algorithm used, and the minimum value is on the order of  $2.2 \times 10^5$ . The reconstructions look similar to those obtained by stopping the iterations at the minimum of  $\rho^{(k)}$ .

As in the case of the star cluster, if we subtract the reconstruction obtained with the approximation  $C^{(1)}(x)$  from that obtained with  $C(x)$ , we find an array with values ranging

from  $-0.95$  to  $0.45$ ; the mean value of the array is  $8.8 \times 10^{-4}$  and the standard deviation  $5.7 \times 10^{-2}$ . On the other hand, if we subtract the reconstruction obtained with the approximation provided by  $C^{(2)}(x)$ , we obtain an array with values ranging from  $-1.64$  to  $7.84$ ; the mean value is  $-0.24$  and the standard deviation is  $0.49$ . Therefore, in such a case, the approximation provided by  $C^{(1)}$  is more accurate than the other one.

The reconstruction provided by the first method is shown in the upper-right panel of figure 3. By comparing with the left panel of figure 3, we see that important details are lost due to the large noise affecting the data. Finally, we also checked the statistical accuracy of the results by looking at the normalized residuals. The expected value is around  $-0.043$  for the first two methods and around  $-0.066$  for the third one, while the variance is  $0.993$  for all of them. All the residuals are free of artifacts and the map of those provided by the first method is shown in the lower-right panel of figure 3.

## 6. Concluding remarks

As already remarked in [25], in the case of images corrupted by both Gaussian and Poisson noise, the more refined algorithm taking into account both kinds of noise does not provide much improvement with respect to the standard RLM that takes only Poisson statistics into account, if the background is sufficiently large to exclude negative values in the image. However, in the case of very faint objects and low background, it provides more accurate reconstructions for quantitative analysis. In addition, the more refined model should be used when the read-out noise introduces negative values in the image.

Besides proving some theoretical results that provide a more sound basis of the main algorithm, we also introduced an approximation that, in some cases, is more accurate than that based on approximating the Gaussian noise with a suitable Poisson noise, while preserving the same computational efficiency. Therefore, this approximation can be used in designing efficient algorithms for the minimization of the penalized log-likelihood function, using the method proposed in [14, 15]. The need of a suitable regularization is made evident, for instance, by the artifacts appearing in the reconstruction of the galaxy. Edge-preserving regularization may be the correct approach and work is in progress in this direction.

Although the numerical experiments reported in this paper apply to the case of a telescope consisting of a simple mirror, the extension to the case of multiple images, also presented in this paper, has been implemented in the software package AIRY and will be tested in the reconstruction of the images of the LINC-NIRVANA interferometer for LBT, especially in the case of very distant and faint objects.

## Acknowledgments

We thank the anonymous referees for their valuable contributions to the improvement of the paper. The work was partially supported by MUR (Italian Ministry for University and Research), grant no 2006018748 and by INAF (Italian Institute for Astronomy and Astrophysics).

## Appendix

This appendix derives the asymptotic behavior of the functions

$$\bar{p}(s; t) = e^{-s} p(s; t), \quad \bar{q}(s; t) = e^{-s} q(s; t), \quad (\text{A.1})$$

where  $p(s; t)$  and  $q(s; t)$  are defined in equations (7) and (8), respectively. Our results hold true for large values of  $t$ , with  $s$  satisfying the conditions

$$t - c\sqrt{t} \leq s \leq t + c\sqrt{t}, \tag{A.2}$$

where  $c$  is a given but arbitrary constant (for instance,  $c = 3$ ). When the functions of equation (A.1) are inserted into the reconstruction algorithm of equation (23), the value of  $t$  is given by the value  $y_i$  of the observed image in a given pixel, while the value of  $s$  is given by the model  $(Hx^{(k)} + b)_i$ ; approximating the data in that pixel at iteration  $k$ . Therefore condition (A.2) implies that the difference between model and data cannot be arbitrarily large.

We first investigate the function  $\bar{p}(s; t)$ ; the result can be easily extended to  $\bar{q}(s; t)$ . If we introduce the integers

$$N_1 = [t - c\sqrt{t}] - 1, \quad N_2 = [t + c\sqrt{t}] + 1 \tag{A.3}$$

(as usual,  $[u]$  means the integer part of  $u$ ), then the values of  $n$  corresponding to the maxima of the Gaussian and Poisson distributions are interior to the interval  $[N_1, N_2]$ . Therefore, we split the series defining  $\bar{p}(s; t)$  into three terms

$$\begin{aligned} \bar{p}(s; t) &= \bar{p}_1(s; t) + \bar{p}_2(s; t) + \bar{p}_3(s; t) \\ &= \left\{ \sum_{n=0}^{N_1-1} + \sum_{n=N_1}^{N_2} + \sum_{n=N_2+1}^{+\infty} \right\} \frac{e^{-s} s^n}{n!} e^{-\frac{1}{2\sigma^2}(n-t)^2}. \end{aligned} \tag{A.4}$$

Since the maximum value of the Gaussian factor is reached inside the interval  $[N_1, N_2]$ , the first and third term can be bounded by the values of the Gaussian, respectively, in  $n = N_1$  and  $n = N_2$ . We have

$$\bar{p}_1(s; t) \leq \exp\left[-\frac{(t - N_1)^2}{2\sigma^2}\right] \leq \alpha_1 \exp\left[-\frac{c^2 t}{2\sigma^2}\right], \tag{A.5}$$

$$\bar{p}_3(s; t) \leq \exp\left[-\frac{(t - N_2)^2}{2\sigma^2}\right] \leq \alpha_3 \exp\left[-\frac{c^2 t}{2\sigma^2}\right], \tag{A.6}$$

where  $\alpha_1, \alpha_3$  are suitable constants and therefore are negligible with respect to the second term, as we will show.

Indeed, the second term can be estimated using the Gaussian approximation of the Poisson distribution. More precisely, from the Stirling approximation of the factorial for large values of  $n$ ,

$$\ln n! = n \ln n - n + \ln\sqrt{2\pi n} + \lambda(n), \tag{A.7}$$

with  $\lambda(n) \leq (12n)^{-1}$ , we can write

$$\begin{aligned} \ln\left(\frac{e^{-s} s^n}{n!}\right) &= -s + n \ln s - \ln n! \\ &= -\ln\sqrt{2\pi s} - n \ln\frac{n}{s} + n - s + e_1(n, s), \end{aligned} \tag{A.8}$$

with

$$e_1(n, s) = -\frac{1}{2} \ln\frac{n}{s} - \lambda(n). \tag{A.9}$$

Then, for  $n \in [N_1, N_2]$  and  $s$  constrained by the condition of equation (A.2), we find that there exist  $\bar{t}$  and  $\alpha_2$  such that, for any  $t \geq \bar{t}$ , we have

$$|e_1(n, s)| \leq \frac{1}{2} \frac{|n - s|}{s} + \frac{1}{12n} \leq \frac{N_2 - N_1}{2N_1} + \frac{1}{12N_1} \leq \frac{\alpha_2}{\sqrt{t}}. \tag{A.10}$$

Next, from the second-order Taylor formula for the natural logarithm, we have

$$\ln \frac{n}{s} = \ln \left( 1 + \frac{n-s}{s} \right) = \frac{n-s}{s} - \frac{1}{2} \left( \frac{n-s}{s} \right)^2 + e_2(n, s), \quad (\text{A.11})$$

and, again, for  $n, s$  satisfying the previous conditions, there exist  $\bar{t}$  and  $\bar{\alpha}_4, \alpha_4$  such that, for  $t \geq \bar{t}$ , we get

$$|e_2(n, s)| \leq \bar{\alpha}_4 \left( \frac{|n-s|}{s} \right)^3 \leq \bar{\alpha}_4 \left( \frac{|N_2 - N_1|}{N_1} \right)^3 \leq \frac{\alpha_4}{t^{3/2}}. \quad (\text{A.12})$$

Then, if we insert equation (A.11) into equation (A.8) and take into account the inequalities (A.10) and (A.12), after some algebra, we obtain the estimate

$$\ln \left( \frac{e^{-s} s^n}{n!} \right) = -\ln \sqrt{2\pi s} - \frac{1}{2s} (n-s)^2 + O \left( \frac{1}{\sqrt{t}} \right),$$

$$n \in [N_1, N_2], \quad |s-t| \leq c\sqrt{t}. \quad (\text{A.13})$$

This equation provides the leading term of an asymptotic expansion of  $\bar{p}_2(s; t)$  for large values of  $t, s$  satisfying condition (A.2),

$$\bar{p}_2(s; t) = \frac{1}{\sqrt{2\pi s}} \sum_{n=N_1}^{N_2} e^{-\frac{1}{2} \left[ \frac{(n-s)^2}{s} + \frac{(n-t)^2}{\sigma^2} \right]} \left\{ 1 + O \left( \frac{1}{\sqrt{t}} \right) \right\}. \quad (\text{A.14})$$

In a similar way we obtain

$$\bar{q}_2(s; t) = \frac{1}{\sqrt{2\pi s}} \sum_{n=N_1}^{N_2} e^{-\frac{1}{2} \left[ \frac{(n-1-s)^2}{s} + \frac{(n-t)^2}{\sigma^2} \right]} \left\{ 1 + O \left( \frac{1}{\sqrt{t}} \right) \right\}. \quad (\text{A.15})$$

Thanks to the inequalities (A.5) and (A.6) (that hold true also for  $\bar{q}_1(s; t)$  and  $\bar{q}_3(s; t)$ , respectively), the leading terms of  $\bar{p}_2(s; t)$  and  $\bar{q}_2(s; t)$  are also the leading terms of the asymptotic expansions of  $\bar{p}(s; t)$  and  $\bar{q}(s; t)$ . These can be written in a more convenient form by introducing the following quantities  $\alpha, \alpha', \beta$  defined by

$$\alpha = s \frac{t + \sigma^2}{s + \sigma^2}, \quad \alpha' = \alpha + \frac{\sigma^2}{s + \sigma^2}, \quad \frac{1}{\beta} = \frac{1}{s} + \frac{1}{\sigma^2}, \quad (\text{A.16})$$

so that the final result is

$$\bar{p}(s; t) = e^{-\frac{(s-t)^2}{2(s+\sigma^2)}} \sum_{n=N_1}^{N_2} e^{-\frac{(n-\alpha)^2}{2\beta}} \left\{ 1 + O \left( \frac{1}{\sqrt{t}} \right) \right\} \quad (\text{A.17})$$

$$\bar{q}(s; t) = e^{-\frac{(s+1-t)^2}{2(s+\sigma^2)}} \sum_{n=N_1}^{N_2} e^{-\frac{(n-\alpha')^2}{2\beta}} \left\{ 1 + O \left( \frac{1}{\sqrt{t}} \right) \right\}. \quad (\text{A.18})$$

However, if we observe that

$$n - \alpha = n - s - \frac{s}{s + \sigma^2} (t - s) = O \left( \frac{1}{\sqrt{t}} \right) \quad (\text{A.19})$$

$$(n - \alpha')^2 = (n - \alpha)^2 + 2 \frac{\sigma^2}{s + \sigma^2} (n - \alpha) + \left( \frac{\sigma^2}{s + \sigma^2} \right)^2$$

$$= (n - \alpha)^2 + O \left( \frac{1}{\sqrt{t}} \right), \quad (\text{A.20})$$

we also have

$$\bar{q}(s; t) = e^{-\frac{(s+1-t)^2}{2(s+\sigma^2)}} \sum_{n=N_1}^{N_2} e^{-\frac{(n-\alpha)^2}{2\beta}} \left\{ 1 + O \left( \frac{1}{\sqrt{t}} \right) \right\}. \quad (\text{A.21})$$

## References

- [1] Anconelli B, Bertero M, Boccacci P, Carbillet M and Lanteri H 2006 Reduction of boundary effects in multiple image deconvolution with an application to LBT LINC-NIRVANA *Astron. Astrophys.* **448** 1217–24
- [2] Bertero M and Boccacci P 1998 *Introduction to Inverse Problems in Imaging* (Bristol: Institute of Physics Publishing)
- [3] Bertero M and Boccacci P 2000 Application of the OS-EM method to the restoration of LBT images *Astron. Astrophys. Suppl. Ser.* **144** 181–6
- [4] Bertero M and Boccacci P 2005 A simple method for the reduction of boundary effects in the Richardson–Lucy approach to image deconvolution *Astron. Astrophys.* **437** 369–74
- [5] Bertero M, Anconelli B, Boccacci P, Desiderà G, Carbillet M and Lanteri H 2006 High resolution image reconstruction: the case of the Large Binocular Telescope (LBT) *Astronomy with High Contrast Imaging III* ed M Carbillet, A Ferrari and C Aime (*EAS Publication Series* vol 22) (Les Ulis: EDP Sciences) pp 35–67
- [6] Correia S, Carbillet M, Boccacci P, Bertero M and Fini L 2002 Restoration of interferometric images-I. The software package AIRY *Astron. Astrophys.* **387** 733–43
- [7] Daube-Witherspoon M E and Muehlelehner G 1986 An iterative image space reconstruction algorithm suitable for volume ECT *IEEE Trans. Med. Imaging* **5** 61–6
- [8] Dempster A D, Laird H M and Rubin D B 1977 Maximum likelihood from incomplete data via the EM algorithm *J. R. Stat. Soc. B* **39** 1–37
- [9] De Pierro A R 1987 On the convergence of the iterative image space reconstruction algorithm for volume ECT *IEEE Trans. Med. Imaging* **6** 124–5
- [10] Donatelli M, Estatico C, Martinelli A and Serra-Capizzano S 2006 Improved image deblurring with anti-reflective boundary conditions and re-blurring *Inverse Problems* **22** 2035–53
- [11] Engl H W, Hanke M and Neubauer A 1996 *Regularization of Inverse Problems* (Dordrecht: Kluwer)
- [12] Hudson H M and Larkin R S 1994 Accelerated image reconstruction using ordered subsets of projection data *IEEE Trans. Med. Imaging* **13** 601–9
- [13] La Camera A, Desiderà G, Arcidiacono C, Boccacci P and Bertero M 2007 Advances in the reconstruction of LBT LINC-NIRVANA images *Astron. Astrophys.* **471** 1091–97
- [14] Lanteri H, Roche M and Aime C 2002 Penalized maximum likelihood image restoration with positivity constraints: multiplicative algorithms *Inverse Problems* **18** 1397–419
- [15] Lanteri H and Theys C 2005 Restoration of astrophysical images—the case of Poisson data with additive Gaussian noise *EURASIP J. Appl. Signal Proc.* **15** 2500–13
- [16] Llacer J and Nuñez J 1990 Iterative maximum likelihood and Bayesian algorithms for image reconstruction in astronomy *The Restoration of HST Images and Spectra* ed R L White and R J Allen (Baltimore: The Space Telescope Science Institute) pp 62–9
- [17] Lucy L B 1974 An iterative technique for the rectification of observed distributions *Astron. J.* **79** 745–54
- [18] Natterer F 2001 *The Mathematics of Computerized Tomography (Classics in Applied Mathematics* vol 32) (Philadelphia, PA: SIAM) p 89
- [19] Natterer F and Wübbeling F 2001 *Mathematical Methods in Image Reconstruction* (Philadelphia, PA: SIAM) p 122
- [20] Polite D G and Snyder D L 1991 Correction for accidental coincidences and attenuation in maximum-likelihood image reconstruction for positron-emission tomography *IEEE Trans. Med. Imaging* **10** 82–9
- [21] Richardson W H 1972 Bayesian based iterative method of image restoration *J. Opt. Soc. Am.* **62** 55–9
- [22] Shepp L A and Vardi Y 1982 Maximum likelihood reconstruction for emission tomography *IEEE Trans. Med. Imaging* **1** 113–22
- [23] Snyder D L 1990 Modifications of the Lucy-Richardson iteration for restoring Hubble Space-Telescope imagery *The Restoration of HST Images and Spectra* ed R L White and R J Allen (Baltimore: The Space Telescope Science Institute) pp 56–61
- [24] Snyder D L, Hammoud A M and White R L 1993 Image recovery from data acquired with a charge-coupled-device camera *J. Opt. Soc. Am. A* **10** 1014–23
- [25] Snyder D L, Helstrom C W, Lanterman A D, Faisal M and White R L 1995 Compensation for read-out noise in CCD images *J. Opt. Soc. Am. A* **12** 272–83
- [26] Stobie E B, Hanish R J and White R L 1994 Implementation of the Richardson–Lucy Algorithm in STSDAS *ASP Conf. Ser.* **61** 296–99
- [27] Vio R, Bardsley J, Donatelli M and Wamsteker W 2005 Dealing with edge effects in least-squares image deconvolution problems *Astron. Astrophys.* **442** 397–403
- [28] White R L 1993 Enhancements to the Richardson–Lucy algorithm *Newslett. STScI's Image Restor. Proj.* **1** 11–23