JOURNAL OF COMPUTATIONAL PHYSICS 103, 169–183 (1992)

An Algorithm for Deconvolution by the Maximum Entropy Method with Astronomical Applications

JOHANN REITER

Mathematisches Institut der Technischen Universität München, Arcisstrasse 21, D-8000 Munich 2, Germany

Received August 21, 1990; revised October 11, 1991

DEDICATED TO PROFESSOR J. PFLEIDERER ON THE OCCASION OF HIS 60TH BIRTHDAY

The solution of a Fredholm integral equation of the first kind which is, in general, an ill-posed problem, can be regularized by the maximum entropy method (MEM). With this method the solution is reformulated as a nonlinear optimization problem with one or two nonlinear constraints. In real life applications, however, this optimization problem is a large-scale one with up to 10^6 unknowns to be determined. For the solution of such problems we present a numerical algorithm which is made to work most efficiently on modern multiprocessor, vector computers. The usefulness of the algorithm is illustrated by deconvolving optical pictures of the sky taken with astronomical telescopes. © 1992 Academic Press, Inc.

1. INTRODUCTION

The mathematical modelling of physical processes very often leads to the problem of solving a Fredholm integral equation of the first kind, which can be represented in its discretized form as

$$\sum_{i=1}^{N} h_{\mu i}\, g_i = d_\mu - \varepsilon_\mu, \qquad \mu = 1, ..., M. \tag{1.1}$$

Here, the "original" ("true") object (source distribution, image, map) g_i is blurred by the operation of the point-spread function (PSF) $h_{\mu i}$ of the measuring device (e.g., spectrometer, tomograph, telescope), giving rise to a data set d_μ which is disturbed by the superposition of random noise fluctuations ε_μ. For notational brevity the suffixes in Eq. (1.1) are written as one-dimensional, even in the multi-dimensional case; Greek indices are used for data space and Latin ones for image space. In practice, the individual values ε_μ cannot be measured and are, therefore, unknown. Only some statistical properties of the underlying distribution from which the ε_μ are drawn can be given. If the PSF depends on the distance between pixels μ and i only, Eq. (1.1) reduces to an integral equation of the convolution type.

Due to the smoothing action of the PSF, the solution of Eq. (1.1) for the g_i is, in general, an ill-posed problem [1]. This means that small perturbations of the (noisy) right-hand side can cause unacceptably different values for the g_i. A straightforward inversion of Eq. (1.1), which is in any case only possible for $N = M$, will lead to excessively oscillating solutions violating, in general, problem-inherent constraints (e.g., positivity of the solution if the g_i represent a density or an intensity distribution, respectively). Obviously, such an approach is meaningless.

In the general case with $N \geqslant M$, important examples of which are aperture synthesis in radio astronomy [2] or medical tomography [3], there exists, in principle, an infinite number of solutions of Eq. (1.1), each of which may be described by a distribution f_i in image space. Associated with each such distribution f_i are simulated data (model values, "mock" data)

$$m_\mu = \sum_i h_{\mu i} f_i, \qquad \mu = 1, ..., M, \tag{1.2}$$

which reproduce the right-hand side of Eq. (1.1) within the noise level. From a purely mathematical point of view, out of the infinite number of possible solutions, none is to be preferred, but as a matter of practical procedure, exactly one satisfactory solution has to be selected. It seems reasonable to choose that which is the "best" in some plausible (albeit arbitrary) sense and which does not violate any a priori knowledge about the true solution. That is, the solution of Eq. (1.1) has to be reformulated as an optimization problem which is mathematically equivalent to stabilizing the solution of Eq. (1.1) via an optimized functional.

Since the pioneering works of Larkin [4] and Frieden [5] during the last two decades, the maximum entropy method (MEM) has proven to be a very powerful and

0021-9991/92 $5.00

reliable technique for deconvolution and image reconstruction and has found widespread applications. The most attractive features of MEM are (i) that the entropy functional (1.3a) can be derived from simple heuristic arguments [6], (ii) that it is data-adaptive in that it uses all the information known about the data and noise, and (iii) that noise is suppressed very efficiently in the reconstructed map or, in other words, that only that structure is deconvolved for which there is sufficient evidence in the data. According to the basic principles of MEM, which, for example, are discussed in Jaynes [7–10] and [11], Eq. (1.1) can be restated as an optimization problem in the following way: Maximize

$$S(\mathbf{f}) = S(f_1, ..., f_N) = -\sum_i \frac{f_i}{F_0}\left(\log\left(N\frac{f_i}{F_0}\right) - 1\right) = \max, \qquad (1.3a)$$

subject to the constraints

$$g_k(\mathbf{r}) = g_k(r_1, ..., r_M) := G_k(\mathbf{r}) - G_k^{\text{aim}} = 0, \qquad k = 1, ..., K. \qquad (1.3b)$$

Here, $S(\mathbf{f})$ is the entropy functional which is modified in order to ensure that the total flux F_0 of the reconstructed map is $\sum_i f_i = F_0$. This can be seen by an unconstrained optimization of S which yields the default values $f_i = F_0/N$, $i = 1, ..., N$, that is, the "flat" map. If the data d_μ and the f_i are given as fluxes per pixel, then it follows immediately that $F_0 = \sum_\mu d_\mu$, except for boundary effects. Flux from outside may be scattered into the map d_μ, or flux within the map may be partly scattered out of the map, and not contribute to $\sum_\mu d_\mu$. However, the deconvolved map is not very sensitive to the precise value of F_0 [12]. For interferometer measurements which do not give the total flux, a zero-spacing observation must be added which directly yields F_0.

The error statistics $G_k(\mathbf{r})$ take care (i) that the reconstructed map f_i represents the actually measured (noisy) data within the noise level σ_μ of datum d_μ via the normalized residuals

$$r_\mu = (m_\mu - d_\mu)/\sigma_\mu, \qquad \mu = 1, ..., M, \qquad (1.4)$$

whereby m_μ is given in Eq. (1.2), and (ii) that the statistical distribution of the r_μ coincides sufficiently well with that known for the ε_μ. In general, a small number ($K \leqslant 2$) of constraints suffices in order to fulfill these two requirements [6].

The parameter G_k^{aim}, the value of which depends on the special form of $G_k(\mathbf{r})$ and can be derived from statistical considerations, sets some (upper) bound to the value which plausibly is allowed for the error statistic G_k. In the case of a Gaussian error law for the ε_μ, mostly a single error statistic ($K = 1$), usually χ^2, is used [13]; that is,

$$G_1(\mathbf{r}) = \sum_\mu r_\mu^2, \qquad (1.5a)$$

and the expectation value $\langle\chi^2\rangle_{\text{exp}}$ or the most probable value $\langle\chi^2\rangle_{\text{prob}}$ of the χ^2-statistic can be used for G_1^{aim}. The smaller a value of G_1^{aim} is chosen the better the model values m_μ must fit the data d_μ. Consequently, more structure must show up in the deconvolved map but at the risk of being nonrealistic and overinterpreting the (noisy) data. Therefore, in practice, different values of G_1^{aim} should always be tried.

Using χ^2 as error statistic, the distribution of the fit residuals r_μ becomes markedly non-Gaussian in certain applications, and it may be preferable to replace χ^2 by the so-called E^2-statistic [14] which is defined by

$$G_1(\mathbf{r}) = \sum_\mu (r_{(\mu)} - v_\mu)^2. \qquad (1.5b)$$

Here, $r_{(\mu)}$ is the μth ordered residual, and v_μ is given by

$$\int_{-\infty}^{v_\mu} e^{-t^2/2}\, dt = \sqrt{2\pi}\,(\mu - 0.5)/M. \qquad (1.6)$$

Again, the expectation value $\langle E^2\rangle_{\text{exp}}$ or the most probable value $\langle E^2\rangle_{\text{prob}}$ of the E^2-statistic can be used for G_1^{aim}. Fit formulas for computing $\langle E^2\rangle_{\text{exp}}$ and $\langle E^2\rangle_{\text{prob}}$, respectively, are given in [15].

Due to the smoothing action of the entropy functional (1.3a), there results a bias of the reconstructed map f_i towards the flat map, giving rise to a strong correlation of the residuals r_μ with the model values m_μ. As was shown in [6] this difficulty can be overcome by introducing an additional constraint ($K = 2$), namely the so-called T-statistic which is defined by

$$G_2(\mathbf{r}) = \sum_\mu r_\mu m_\mu/\sigma_\mu. \qquad (1.7)$$

By applying these two constraints, not only the overall shape of the residual distribution is checked (by χ^2 or E^2, respectively) but also the spatial distribution over the reconstructed map (by T). In practical applications, we found values of G_2^{aim} in the range $-5 \leqslant G_2^{\text{aim}}/M \leqslant +5$ to be suitable.

Due to the close connection of the entropy functional with the information content (structure) of the reconstructed map, the approach (1.3) for solving a Fredholm integral equation of the first kind seeks to find a compromise between a maximally smooth map and an optimal fit of the model values to the data set. It should be noted that

the comparison of the reconstructed map and the actually measured data set is made entirely in the data space via the constraints $g_k(\mathbf{r})$.

A numerical algorithm for solving problem (1.3) with $K=2$ which is made to work most efficiently on modern vector-processing computers is given in Section 2. Examples are given in Section 3, followed by some conclusions in Section 4.

2. A NUMERICAL ALGORITHM

The maximum entropy approach (1.3) for solving the Fredholm integral equation of the first kind (1.1) requires the solution of a nonlinear optimization problem with nonlinear constraints. The problem is also a large-scale one, since the reconstructed map (the f_i's) may contain up to about $N=10^6$ pixels that must be determined. The number of data points M may be similarly large. It follows immediately from this large-scale nature of the problem that ordinary methods for solving constrained optimization problems, such as modified Newton methods or sequential quadratic programming algorithms, cannot be applied, due to the required storage and inversion of $N \times N$ matrices. The handling of such material is beyond the scope of even very large modern computers. However, Newton methods have been successfully employed for solving small-scale maximum entropy problems [5, 16]. Zeroth-order methods which require the computation of the functional S and the constraints g_k only, are also not applicable because their rate of convergence is generally too slow. Similar considerations hold for first-order methods which involve, beside the computation of S and g_k, the corresponding gradients too. It seems that only those methods are feasible which use transformations iteratively from the image space into the data space and vice versa [17]. Mathematically well-grounded algorithms for solving large-scale maximum entropy problems do exist in the literature [18, 19], but these are only for the cases where the constraints can be taken as linear. Clearly, these algorithms cannot be applied to our current problem.

In order to overcome the above-mentioned difficulties associated with the high dimensionality of the problem, a new algorithm based upon the variable-metric method of Wilson [20] was developed. As it turned out this algorithm works most efficiently on vector-processing computers, since many vector operations in image-space and in data-space, together with image–data transformations and vice versa have to be performed in each iteration step of the algorithm. Of course, this implies that the image–data as well as the data–image transformations must be coded efficiently. In the case where Eq. (1.1) is of the convolution type, the transformations are simple convolutions which can be computed in the most efficient way via the fast Fourier transform (FFT). This is true to a high degree for vector-processing machines.

The method of Wilson defines, in each iteration step, a subproblem to minimize a quadratic approximation of the Lagrangian function subject to the linearized constraints. Therefore, problem (1.3) can be transformed into the sequential solution of the subproblems

$$\tfrac{1}{2}\mathbf{d}^{\mathrm{T}} H \mathbf{d} + \nabla S(\mathbf{f})^{\mathrm{T}}\, \mathbf{d} = \min \tag{2.1a}$$

subject to

$$\begin{aligned} \nabla g_1(\mathbf{r})^{\mathrm{T}}\, \mathbf{d} + g_1(\mathbf{r}) &= 0 \\ \nabla g_2(\mathbf{r})^{\mathrm{T}}\, \mathbf{d} + g_2(\mathbf{r}) &= 0, \end{aligned} \tag{2.1b}$$

where we have assumed that two constraints are in operation ($K=2$). Here, the entropy functional (1.3a) is replaced by $-S$ in order to have a minimization instead of a maximization problem. The actual metric in Eq. (2.1a) is defined by

$$H = \nabla\nabla S(\mathbf{f}) - p_1\, \nabla\nabla g_1(\mathbf{r}) - p_2\, \nabla\nabla g_2(\mathbf{r}) \tag{2.2}$$

which is the Hessian of the Lagrangian function

$$L(\mathbf{f}) = S(\mathbf{f}) - p_1\, g_1(\mathbf{r}) - p_2\, g_2(\mathbf{r}), \tag{2.3}$$

and the current iterates are denoted by $\mathbf{f}$ and $\mathbf{r}$, whereby $\mathbf{r}$ is a function of $\mathbf{f}$ which is given in Eq. (1.4). The symbol ∇ means differentiation with respect to f_i; that is,

$$\nabla S(\mathbf{f})^{\mathrm{T}} = \left(\frac{\partial S}{\partial f_1}, \ldots, \frac{\partial S}{\partial f_N}\right)^{\mathrm{T}},$$

$$\nabla g_k(\mathbf{r})^{\mathrm{T}} = \left(\frac{\partial g_k}{\partial f_1}, \ldots, \frac{\partial g_k}{\partial f_N}\right)^{\mathrm{T}}.$$

The optimal Lagrange multipliers p_1, p_2 of subproblem (2.1) are used as guesses for defining the Hessian H in the following iteration step. A line search with a step size parameter $\lambda > 0$ is used to find a new vector

$$\mathbf{f}^* = \mathbf{f} + \lambda \mathbf{d}, \tag{2.4}$$

such that a penalty function $P(\lambda)$ will have a lower function value at the new iterate $\mathbf{f}^*$. For more theoretical details see [21, 22].

Because of the high dimensionality of maximum entropy reconstructions ($\mathbf{d}$ is an N-component vector, H is an $N \times N$ matrix!) it is evident that the subproblem (2.1) is unsolvable, since the inverse of the Hessian H cannot be determined. Therefore, we have tried: (i) to represent

the increment vector $\mathbf{d}$ as a linear combination of a small number n_s of suitable chosen "search directions" $\mathbf{e}_l$, that is,

$$\mathbf{d} = \sum_{l=1}^{n_s} \alpha_l \mathbf{e}_l, \tag{2.5}$$

where each $\mathbf{e}_l$ is an N-component vector belonging to image space; and (ii) to replace the customary line search by constraining the length of $\mathbf{d}$, for example, by the condition [17]

$$\mathbf{d}^{\mathrm{T}}(\nabla\nabla S)\,\mathbf{d} \leqslant R/F_0, \tag{2.6}$$

where $R>0$ is a free parameter of the algorithm. Condition (2.6) is equivalent to defining a "trust" region in N-dimensional space within which the next iterate of the map $\mathbf{f}$ has to be found.

The value of the parameter R can be estimated in the following manner. Inserting the flat map $f_i^{(0)} = F_0/N$, $i = 1, ..., N$, into the cost function (1.3a) we find that $S(\mathbf{f}^{(0)}) = 1$. Due to the logarithmic term involved, $S(\mathbf{f})$ is a slowly varying function of its argument $\mathbf{f}$. Therefore, we have $S(\mathbf{f}) \sim \mathcal{O}(1)$ for any other map $\mathbf{f}$. Having this in mind and expanding S into a Taylor series,

$$S(\mathbf{f}+\mathbf{d}) = S(\mathbf{f}) + \nabla S^{\mathrm{T}}\mathbf{d} + \tfrac{1}{2}\mathbf{d}^{\mathrm{T}}(\nabla\nabla S)\,\mathbf{d} + \cdots,$$

we see immediately that a reasonable trust region around $\mathbf{f}$ could be defined by $\delta := \mathbf{d}^{\mathrm{T}}(\nabla\nabla S)\,\mathbf{d}$ being sufficiently smaller than unity, say $\delta \sim 0.1$. Comparing this estimate with Eq. (2.6), we find that R should be of the order of $\delta \cdot F_0$. This result is confirmed through numerical experiments, where we have found values of R in the range (0.1–0.3) F_0 to be adequate.

The main advantage of using the Hessian $\nabla\nabla S$ as the metric for calculating the length of $\mathbf{d}$ in Eq. (2.6) is that, because of

$$\mathbf{d}^{\mathrm{T}}(\nabla\nabla S)\,\mathbf{d} = \frac{1}{F_0}\sum_{i,j=1}^{N} d_j \frac{\delta_{ij}}{f_i} d_i = \frac{1}{F_0}\sum_{i=1}^{N}\frac{d_i^2}{f_i}, \tag{2.7}$$

large pixel values f_i are allowed to change more than low ones [23]. This is important because high values of f_i presumably represent the most interesting parts of the reconstructed map. In addition, it helps to accelerate the convergence rate of the algorithm.

It is evident that the search directions $\mathbf{e}_l$ cannot be derived on purely mathematical grounds. The following considerations, however, show how to choose them in a heuristic (albeit arbitrary) way. Since

$$F_0(\nabla\nabla S) = \mathrm{diag}(f_1^{-1}, ..., f_N^{-1}) =: D^{-1}$$

is a diagonal matrix, we have for the Hessian (2.2)

$$H = \frac{1}{F_0}D^{-1} - p_1\nabla\nabla g_1 - p_2\nabla\nabla g_2$$

$$= \frac{1}{F_0}D^{-1}(I_{N\times N} - F_0 p_1 D(\nabla\nabla g_1) - F_0 p_2 D(\nabla\nabla g_2)),$$

where $I_{N\times N}$ is the identity matrix of order N. Thus, the inverse Hessian H^{-1} is given by

$$H^{-1} = F_0(I_{N\times N} - F_0 p_1 D(\nabla\nabla g_1) - F_0 p_2 D(\nabla\nabla g_2))^{-1} D. \tag{2.8a}$$

As mentioned in the Introduction, the job of the error statistic g_1 is to force the reconstructed map to represent the actually measured data. Consequently, g_1 is responsible for the deconvolution of basic structures inherent in the map. To the contrary, the statistic g_2 was introduced into the problem in order to force the residuals r_μ to be more or less uncorrelated with the model values m_μ. Consequently, the application of g_2 is expected to result: (i) in an enlargement of peaks, (ii) in an enhancement of structure in general, and (iii) in a smoothing of the background. However, through the use of g_2, new major peaks or new basic structure are not expected to show up in the deconvolved map. Therefore, it can safely be assumed that constraint g_2 will give only small corrections to the solution of subproblem (2.1), in comparison with that of the first constraint g_1.

In retrospect, this assumption can be checked after the solution of problem (1.3) by inspection of the Lagrange multipliers. Let p_1 and p_2 denote the Lagrange multipliers relative to constraint g_1 and g_2, respectively, at the optimal point. Since a Lagrange multiplier indicates the relative

TABLE I

M87: Numerical Results for the Various Versions of MEM with $N = M = 128 \times 128$

	Version of MEM		
	χ^2	E^2	E^2+T
G_1^{aim}	16682	3.52	3.52
G_2^{aim}	—	—	0
S	−0.6539	−0.6171	−0.5694
p_1	-4.18×10^{-6}	-1.31×10^{-6}	-7.00×10^{-4}
p_2	—	—	$+1.06\times10^{-6}$
Iteration number	11	20	27
CPU-time (s)	4.90	10.28	16.19

Note. S is the value of the modified entropy functional at the final map, p_1 and p_2 are the optimal Lagrange multipliers. In order to calculate S the scaling factor was chosen to be $F_0 = 1$. G_1^{aim} and G_2^{aim} are the final values of the error statistics.

TABLE II

NGC 1144: Numerical Results for the Various Versions of MEM with $N = M = 256 \times 256$

	Version of MEM			
	χ^2	E^2	E^2+T	E^2+T
G_1^{aim}	65536	3.68	3.68	3.68
G_2^{aim}	—	—	$4M$	$2M$
S	−0.99789	−0.99272	−0.99250	−0.99198
p_1	-4.26×10^{-8}	-2.98×10^{-5}	-1.03×10^{-3}	-5.13×10^{-5}
p_2	—	—	-3.22×10^{-7}	-1.26×10^{-8}
Iteration number	9	27	53	52
CPU-time (s)	15.79	48.84	113.3	111.5

Note. S is the value of the modified entropy functional at the final map, p_1 and p_2 are the optimal Lagrange multipliers. In order to calculate S the scaling factor was chosen to be $F_0 = 1$. G_1^{aim} and G_2^{aim} are the final values of the error statistics.

importance of the corresponding constraint [24], we would expect that $|p_2| \ll |p_1|$. As will be demonstrated by the examples, this expectation proves true (see Tables I and II). On these grounds, it seems justified to neglect in Eq. (2.8a) the term involving g_2. Then, a formal Taylor expansion of Eq. (2.8a) yields, for the inverse of the Hessian, approximately

$$H^{-1} \approx F_0(D + F_0 p_1 D(\nabla\nabla g_1) D + F_0^2 p_1^2 D(\nabla\nabla g_1) D(\nabla\nabla g_1) D). \quad (2.8b)$$

Inserting Eq. (2.8b) in Eq. (2.1) shows immediately that the vector $\mathbf{d}$ can be represented as a linear combination of the six vectors

$$\begin{aligned} &\mathbf{e}_1 = D\nabla S, \quad \mathbf{e}_3 = D(\nabla\nabla g_1)\,\mathbf{e}_1, \quad \mathbf{e}_5 = D(\nabla\nabla g_1)\,\mathbf{e}_3, \\ &\mathbf{e}_2 = D\nabla g_1, \quad \mathbf{e}_4 = D(\nabla\nabla g_1)\,\mathbf{e}_2, \quad \mathbf{e}_6 = D(\nabla\nabla g_1)\,\mathbf{e}_4, \end{aligned} \quad (2.9)$$

where $D = \text{diag}(f_1, ..., f_N)$. It seems advantageous to identify the search directions in the N-dimensional space with the six vectors $\mathbf{e}_l$ given in Eq. (2.9); that is, we have $n_s = 6$. From a conceptually quite different approach ("entropy metric") Skilling and Bryan [17] arrived at a similar result.

The family (2.9) of search directions has proven to be sufficiently powerful to solve all test problems so far. This again justifies the omission of constraint g_2 when inverting the Hessian H. It is rather remarkable that such a small subspace can capture enough of the structure of the large-scale optimization problem (1.3) [17]. However, in very difficult problems it might be helpful to have some information about g_2 present in the subspace spanned by the base vectors $\mathbf{e}_l$. Otherwise, the algorithm might converge very slowly. In any case, if convergence is reached, the final deconvolved map does not depend at all on the actual choice of the individual subspaces during the iterations. In view of the computational expense of the algorithm, however, as few search directions as possible should be invoked.

It is worth mentioning that for the calculation of the n_s search directions $\mathbf{e}_l$, defined in Eq. (2.9), no matrix-vector operations are necessary which would be prohibited due to the high dimensionality of the vectors and matrices involved. The main reasons for this are: (i) that D is a diagonal matrix, and (ii) that the Hessian $\nabla\nabla g_1$ is of special structure. For example, let g_1 be given by

$$g_1(\mathbf{r}) = \sum_{\mu=1}^{M} r_\mu^2 - G_1^{\text{aim}}. \quad (2.10)$$

The Hessian of it is

$$(\nabla\nabla g_1)_{ij} = 2 \sum_{\mu=1}^{M} \frac{h_{\mu i} h_{\mu j}}{\sigma_\mu^2}. \quad (2.11)$$

For the ith component of $D(\nabla\nabla g_1)\mathbf{a}$ (whereby $\mathbf{a}$ is an arbitrary N-component vector), we have

$$(D(\nabla\nabla g_1)\mathbf{a})_i = 2 f_i \sum_{\mu=1}^{M} \frac{h_{\mu i}}{\sigma_\mu^2} \left(\sum_{j=1}^{N} h_{\mu j} a_j \right). \quad (2.12)$$

As can be seen from Eq. (2.12) only one data–image and one image–data transform has to be performed in order to calculate the vector $D(\nabla\nabla g_1)\mathbf{a}$. One should note that in Eq. (2.12) the summation in data-space is on the first index of $h_{\mu i}$, which is equivalent to using the transpose $h_{i\mu}^{\mathrm{T}}$ of the PSF. Similar considerations hold for the calculation of the search directions in connection with the other error statistics defined in Section 1.

By means of the definition of $\mathbf{d}$ given in Eq. (2.5) the subproblem (2.1) is transformed into the new subproblem

$$\begin{aligned} &\tfrac{1}{2}\mathbf{y}^{\mathrm{T}} H^* \mathbf{y} + \mathbf{s}^{\mathrm{T}}\mathbf{y} = \min \\ &\mathbf{a}_1^{\mathrm{T}}\mathbf{y} + g_1 = 0 \\ &\mathbf{a}_2^{\mathrm{T}}\mathbf{y} + g_2 = 0, \end{aligned} \quad (2.13a)$$

where we have to determine the unknown vector $\mathbf{y}$ belonging to the subspace Σ of dimension n_s. Thus, the N-dimensional problem (2.1) has been transformed into the n_s-dimensional one (2.13a). Here, we have set

$$\begin{aligned} (\mathbf{s})_l &= \nabla S^{\mathrm{T}} \mathbf{e}_l, \\ (\mathbf{a}_1)_l &= \nabla g_1^{\mathrm{T}} \mathbf{e}_l, \\ (\mathbf{a}_2)_l &= \nabla g_2^{\mathrm{T}} \mathbf{e}_l, \\ (H^*)_{ml} &= \mathbf{e}_m^{\mathrm{T}}(\nabla\nabla S)\,\mathbf{e}_l - p_1 \mathbf{e}_m^{\mathrm{T}}(\nabla\nabla g_1)\,\mathbf{e}_l - p_2 \mathbf{e}_m^{\mathrm{T}}(\nabla\nabla g_2)\,\mathbf{e}_l \\ &=: (H^{(3)})_{ml} - p_1 (H^{(1)})_{ml} - p_2 (H^{(2)})_{ml}, \end{aligned}$$

with $l, m = 1, ..., 6$. Correspondingly, the inequality constraint (2.6) transforms into

$$\mathbf{y}^{\mathrm{T}} H^{(3)} \mathbf{y} \leqslant R/F_0. \tag{2.13b}$$

In order to simplify the solution of the new subproblem (2.13) we diagonalize matrix H^* by virtue of the transformation

$$\mathbf{y} = EF\mathbf{x}, \qquad \mathbf{x}, \mathbf{y} \in \Sigma, \tag{2.14}$$

by which (2.13) reduces to

$$\begin{aligned} \tfrac{1}{2}\mathbf{x}^{\mathrm{T}} \operatorname{diag}(\delta_1, ..., \delta_6)\, \mathbf{x} + \mathbf{b}^{\mathrm{T}}\mathbf{x} &= \min \\ \mathbf{c}_1^{\mathrm{T}}\mathbf{x} + g_1 &= 0 \\ \mathbf{c}_2^{\mathrm{T}}\mathbf{x} + g_2 &= 0 \\ R/F_0 - \mathbf{x}^{\mathrm{T}}\mathbf{x} &\geqslant 0 \end{aligned} \tag{2.15}$$

with $\mathbf{b} = (EF)^{\mathrm{T}}\mathbf{s}$, $\mathbf{c}_1 = (EF)^{\mathrm{T}}\mathbf{a}_1$, and $\mathbf{c}_2 = (EF)^{\mathrm{T}}\mathbf{a}_2$. The transformation matrices E and F, respectively, are defined by the equations

$$\begin{aligned} E&: \begin{cases} E^{\mathrm{T}} H^{(3)} E = I_{6\times 6}, \\ E^{\mathrm{T}} H^{(1)} E = \operatorname{diag}(\eta_1, ..., \eta_6), \end{cases} \\ F&: \begin{cases} F^{\mathrm{T}} F = I_{6\times 6}, \\ F^{\mathrm{T}} \tilde{H} F = \operatorname{diag}(\delta_1, ..., \delta_6), \end{cases} \end{aligned} \tag{2.16a}$$

where we have set

$$\tilde{H} = I_{6\times 6} - p_1 E^{\mathrm{T}} H^{(1)} E - p_2 E^{\mathrm{T}} H^{(2)} E, \tag{2.16b}$$

and $I_{6\times 6}$ is the unity matrix of order 6. The transformation matrices E and F, respectively, actually do exist, since the matrices $H^{(1)}$, $H^{(2)}$, $H^{(3)}$ are symmetric, and $H^{(3)}$ is positive-definite (see Eq. (2.7)). However, it may happen that the search directions $\mathbf{e}_l$ are linearly dependent, giving rise to a singular or nearly singular matrix $H^{(3)}$. This can be taken into account in the course of the simultaneous diagonalization of matrices $H^{(1)}$ and $H^{(3)}$ by discarding "small" eigenvalues of $H^{(3)}$, which is equivalent to reducing the dimension n_s of subspace Σ. Of course, such a reduction of n_s by no means affects the existence of matrices E and F.

Coupling the inequality constraint in Eq. (2.15) to the cost functional via a Lagrange multiplier ρ, we finally arrive at the subproblem

$$\begin{aligned} \tfrac{1}{2}\mathbf{x}^{\mathrm{T}} \operatorname{diag}(\delta_1 + 2\rho, ..., \delta_6 + 2\rho)\, \mathbf{x} + \mathbf{b}^{\mathrm{T}}\mathbf{x} &= \min \\ \mathbf{c}_1^{\mathrm{T}}\mathbf{x} + g_1 &= 0 \\ \mathbf{c}_2^{\mathrm{T}}\mathbf{x} + g_2 &= 0 \end{aligned} \tag{2.17a}$$

which has to be solved in each iteration step of the algorithm. Hereby, the Lagrange multiplier ρ has to be determined in such a way that the condition

$$R/F_0 - \mathbf{x}^{\mathrm{T}}\mathbf{x} \geqslant 0 \tag{2.17b}$$

is met. Subproblem (2.17) is a simple least-squares problem with two linear equality constraints and one quadratic inequality constraint. The solution of (2.17) is a crucial step in finding a solution of the optimization problem (1.3). Unfortunately, (2.17) may not have a solution because no feasible point of problem (2.17a) may satisfy the inequality constraint (2.17b). Then a slight modification of (2.17a) is necessary. Details of the solution of subproblem (2.17) are given in the Appendix.

Summarizing the individual steps which we have discussed above, we propose the following algorithm for solving large-scale maximum entropy problems which can be cast in the form of Eq. (1.3):

(A) Start with the "flat" map: $f_i = F_0/N$, $i = 1, ..., N$;
Set Lagrange multipliers: $p_1 = p_2 = 0$;

(B) Initiate iteration step:
  (1) Compute S, g_1, g_2, and the gradients ∇S, ∇g_1, ∇g_2;
  (2) Compute search directions $\mathbf{e}_l$, $l = 1, ..., 6$;
  (3) Compute matrices $H^{(k)}$, $k = 1, 2, 3$, and vectors $\mathbf{s}$, $\mathbf{a}_1$, $\mathbf{a}_2$ of subproblem (2.13);
  (4) Compute matrices E, F, and $\operatorname{diag}(\delta_1, ..., \delta_{n_s})$;
  (5) Compute vectors $\mathbf{b}$, $\mathbf{c}_1$, and $\mathbf{c}_2$ to set up subproblem (2.17);

(C) Solve subproblem (2.17);
determine optimal Lagrange multipliers p_k, $k = 1, 2$.

(D) Correction step:
  (1) Compute $\mathbf{y} = EF\mathbf{x}$, and the increment vector $\mathbf{d} = \sum_{l=1}^{n_s} y_l \mathbf{e}_l$;
  (2) Update map $\mathbf{f}^* = \mathbf{f} + \mathbf{d}$, and Lagrange multipliers $p_k^* = p_k$, $k = 1, 2$.

(E) Test of convergence:
If $\sqrt{g_1^2 + g_2^2} \leqslant \varepsilon_1$ and $\|p_1^*\mathbf{c}_1 + p_2^*\mathbf{c}_2 - \mathbf{b}\| \leqslant \varepsilon_2$ STOP, or go to (B).

Here, ε_1 and ε_2 are free parameters of the algorithm which control the accuracy to which the optimal point has to be reached. We have found the values $\varepsilon_1 = 10^{-2}$ and $\varepsilon_2 = 5 \times 10^{-2}$ to be useful in practice. The reconstructed maps, however, vary only a little, even when these tolerance criteria are substantially relaxed.

We have coded the algorithm described above with approximately 2300 statements in Fortran 77. Three different versions of MEM have been implemented. In the first version only one constraint is active, and χ^2 (see Eq. (1.5a)) serves as the single error statistic ($K = 1$, "χ^2-version"). The

same is true for the second version, apart from χ^2 being replaced by E^2 which is defined in (1.5b) ($K=1$, "E^2-version"). Finally, the third version uses both E^2 and T (see Eq. (1.7)) as error statistics ($K=2$, "(E^2+T)-version"). Each iteration step of the algorithm requires the computation of seven image–data and six data–image transforms in order to compute the constraints, the search directions, and the parameters of the subspace models. In addition, a considerable number of dot products each of length N or M have to be calculated. For example, the χ^2- or E^2-version needs 49 dot products and for the (E^2+T)-version some 98 dot products have to be computed in each iteration step. Moreover, if E^2 is used as the error-statistic, the M residuals (1.4) must be sorted in increasing order. Fortunately, evaluation of the dot products as well as sorting of the residuals can be vectorized excellently and even parallelized to a great extent if a multiprocessor machine is used. The major computing expenses of the algorithm are, however, the transforms. In all applications which we have carried out so far about 50 to 60% of the total CPU-time is necessary to perform the required transforms. This implies that they must be coded exceedingly efficiently. The number of iterations required to reach convergence to the optimal point depends both on the signal to noise ratio of the data and on the version of MEM being used. In general, 10–50 iterations are sufficient for most problems. Interestingly, this number does not depend noticeably on the numbers N and M.

The resulting program has proved highly reliable and powerful in a number of applications [15]. Difficulties have occurred only if the maximum entropy problem was improperly posed, which may happen predominantly in the following circumstances:

(a) If the baseline of the map has been estimated too high, there may exist too many data points with $d_\mu < 0$. In this case, no solution of the optimization problem (1.3) may exist, since the fulfillment of the constraints (1.3b) may not be possible on grounds of the positivity requirement of the entropy functional. If, on the other hand, the baseline has been determined too low, the positivity requirement may become irrelevant.

(b) The noise estimate in an image is crucial because it enters into whatever error statistic is used. If, for example, the values of the errors σ_μ of the data d_μ are estimated too small, the model values m_μ are forced to fit the d_μ too close, and noise on the data may be interpreted as true signal. An extremely noisy map, however, is in contradiction to the smoothness requirement of the entropy functional and, consequently, the algorithm may not converge under these circumstances.

(c) If occasional erroneous data points are included in the data set d_μ, no solution of problem (1.3) may exist, due to the smoothing action of the entropy functional. However, if the E^2-statistic is used as constraint, such corrupted points can easily be picked out as outliers when the residuals are sorted and can be inspected individually [14].

In particular, items (a) and (b) are important in the context of radio astronomical observations, where the baseline of the map as well as the noise σ_μ has to be determined from the measured data set d_μ itself.

3. EXAMPLES

The following examples were all performed at the Leibniz–Rechenzentrum in Munich, using the CRAY Y-MP4/432 computer system, running under the operating system UNICOS at level 5.1.

3.1. *Photography of M87*

The elliptical galaxy M87 (NGC 4486) in the Virgo cluster was photographed on fine grain IIIa-J emulsion by H. Arp, using the 5 m Hale telescope on Mt. Palomar at a seeing of about 1 in. and digitized in a raster of 120×120 points. The digital plate scans were obtained with a two-axis PDS microphotometer at eight bits per pixel [25]. The discretized image, a contour plot of which is shown in Fig. 1, was deconvolved by a Wiener-filter method [25], MEM-deconvolved by Bryan and Skilling [14] and Reiter and Pfleiderer [6], and deconvolved by Pfleiderer using a mini-

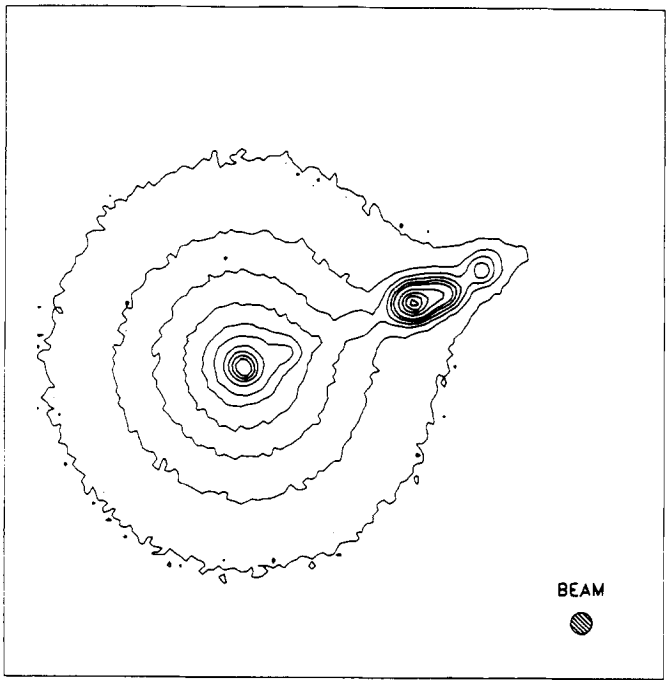

FIG. 1. Raw data map (roughly 1 × 1 ft). The contour levels are 10, 20, 30, 40, 50, 60, 80, 100, 120, 140, 180, 220, 260, and 300. North is up, East is left.

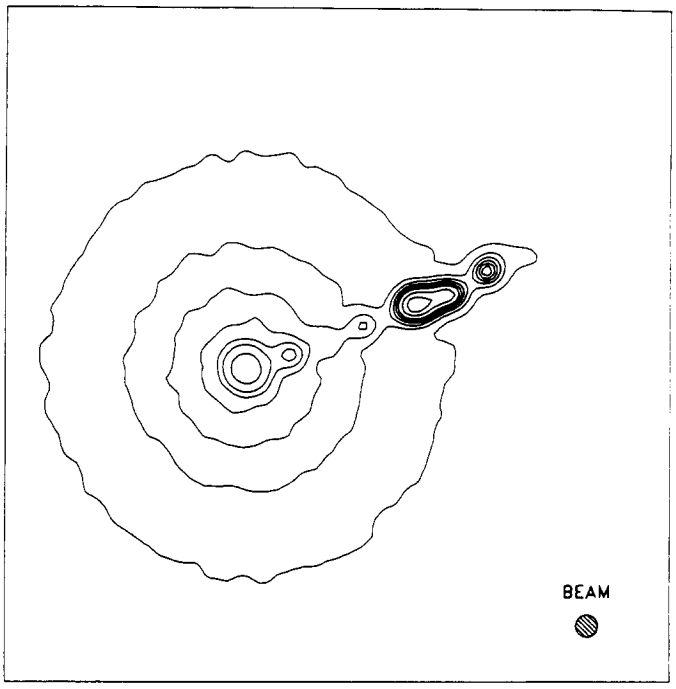

FIG. 2. Deconvolved source map of M87, χ^2-version with $G_1^{\text{aim}} = 16682$. Contour levels as in Fig. 1. The resolution is increased even if the high peaks are somewhat lowered as compared to the raw data.

mum information (MIM) approach [26]. The relevance of M87 to the various methods of deconvolution lies in the following facts:

(a) the jet emerging from the nucleus of M87 consists of several distinct knots which serve as a good test for the resolution achieved in the deconvolved map.

(b) Noise σ_μ and data d_μ are significantly correlated, as is typical for photographic plates. In MEM-deconvolutions this correlation generally leads to a distribution of the residuals r_μ (1.4), contradicting that of the assumed noise model if only one constraint (1.5a) or (1.5b), respectively, is used. In this case, therefore, the additional application of the T-statistic (1.7) is mandatory in order to force the residuals to have their correct statistical distribution [6]. In Wiener-filter deconvolutions, the noise–data correlation may yield unrealistic dark haloes around point-like sources due to filter cutoff at high spatial frequencies [25].

For the purpose of getting a code which is executable as fast as possible, the image–data and data–image transformations are both taken as circular, i.e., each row and column of the matrix to be convoluted is continued periodically. This is possible because the raw data map is sufficiently smooth near the boundaries (Fig. 1). The fast Fourier transform (FFT) can be used to great advantage for computing the convolutions because the matrices are periodic. For a further speedup the data map is extrapolated to $128 \times 128 = 2^{14}$ pixels in order to make the FFT work most

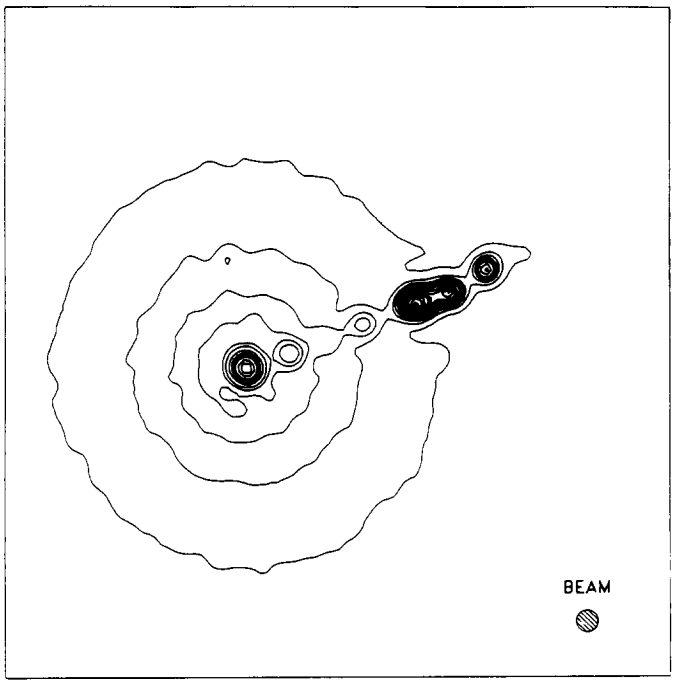

FIG. 3. Deconvolved source map of M87, E^2-version with $G_1^{\text{aim}} = 3.52$. Contour levels as in Fig. 1. The resolution and peak height is increased.

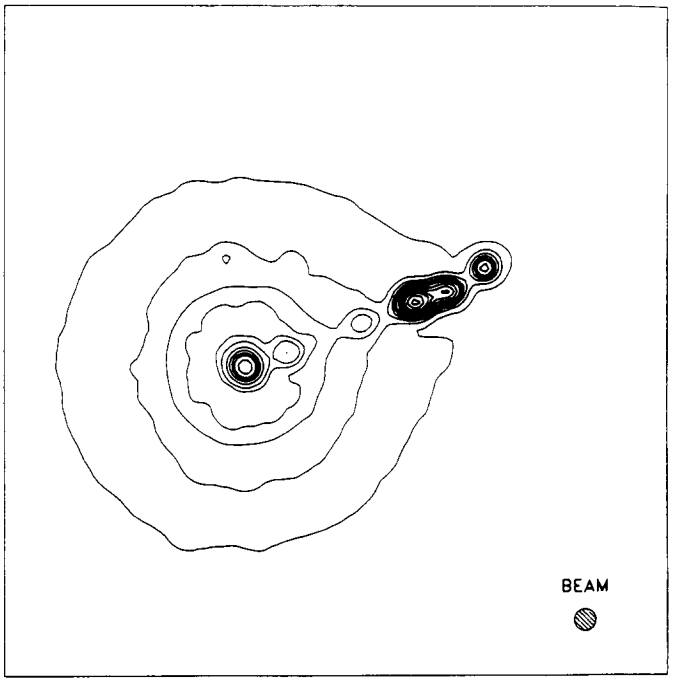

FIG. 4. Deconvolved source map of M87, $(E^2 + T)$-version with $G_1^{\text{aim}} = 3.52$ and $G_2^{\text{aim}} = 0$. Contour levels as in Fig. 1. As compared to Fig. 3, no increase in resolution but enhancement of peaks.

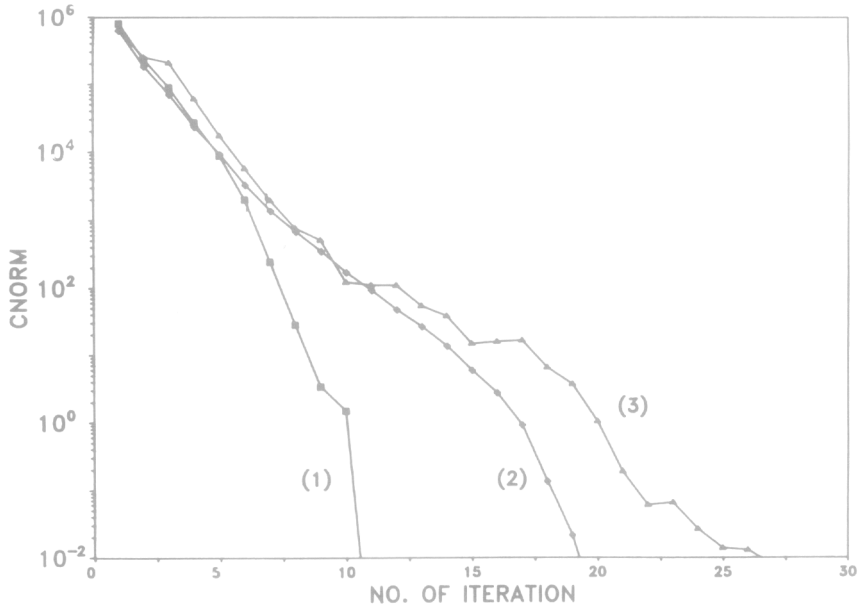

FIG. 5. M87: March of CNORM $= \sqrt{g_1^2 + g_2^2}$ in dependence of the iteration step. The χ^2-version is labeled (1), the E^2-version is labeled (2), and the $(E^2 + T)$-version is labeled (3).

efficiently. The resulting code can perform a single transform in about 24 ms of CPU-time.

The results of the deconvolutions obtained by the different versions of MEM are given in Figs. 2 to 4. A more detailed discussion of these results can be found in [6, 15]. The main results concerning the numerical algorithm are given in Table I, and in Fig. 5 the rate of convergence for the various versions of MEM is shown.

As can be seen in Tables I and II, the optimal Lagrange multiplier p_2 is about two to three orders of magnitude smaller than the optimal value of p_1. This can be taken as an a posteriori vindication for the assumption made in Section 2 that the second constraint need not be taken into account for the inversion of the Hessian matrix when computing the search directions $\mathbf{e}_l$. The smallness of the Lagrange multipliers p_1 and p_2 indicates that the optimal value of S and, hence, the final reconstructed map, does not depend critically on the values chosen for G_1^{aim} and G_2^{aim} [24].

Exploiting the parallelism inherent in the algorithm, it is possible to save 16 and 27% of the elapsed CPU-time (given in Table I) for the χ^2- or E^2-version and the $(E^2 + T)$-version, respectively, if four processors of the CRAY Y-MP/432 are invoked. The total CPU-time, however, is much larger, due to the overhead which is necessary in order to synchronize the processors.

3.2. *CCD frame of NGC 1143/1144 (Arp 118)*

The data of the interacting pair of galaxies NGC 1143/44 were kindly supplied to us by Mr. Roberto Vio of the Osservatorio Astronomico di Padova. The CCD had 512×291 pixels of size 0.32 in. squared. For our purpose, however, we actually use only a smaller "submap" of size $256 \times 256 = 2^{16}$ pixels which has been cut out from the larger raw data map. The most interesting features are already contained in this smaller map, plots of which are shown in Figs. 6 and 7, respectively. Clearly, six sources A to F can be identified, the

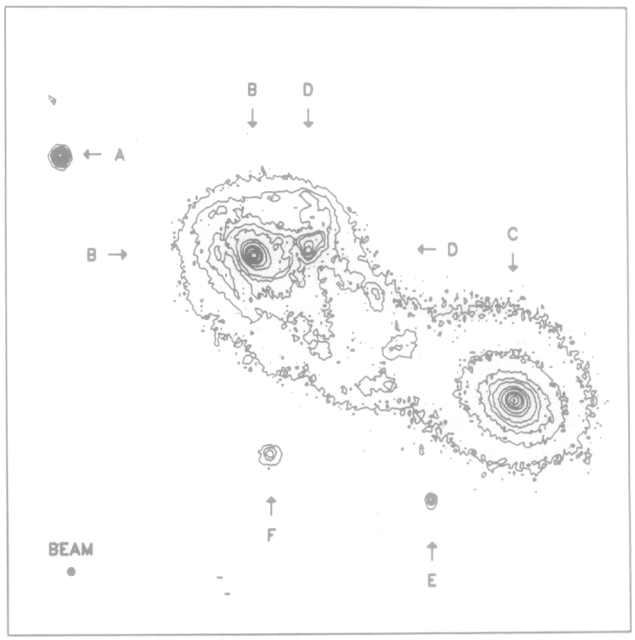

FIG. 6. Contour plot of the raw data map of NGC 1144. The contour levels are 1, 2, 3, 4, 5, 7.5, 10, 15, 20, 30, 40, 50, and 70. The maximum of the map is 85.41 at source A, the minimum is -0.78. In all, 14,752 pixels have negative values. South is at top, East is left. The distance of the two peaks B and C is 40 in.

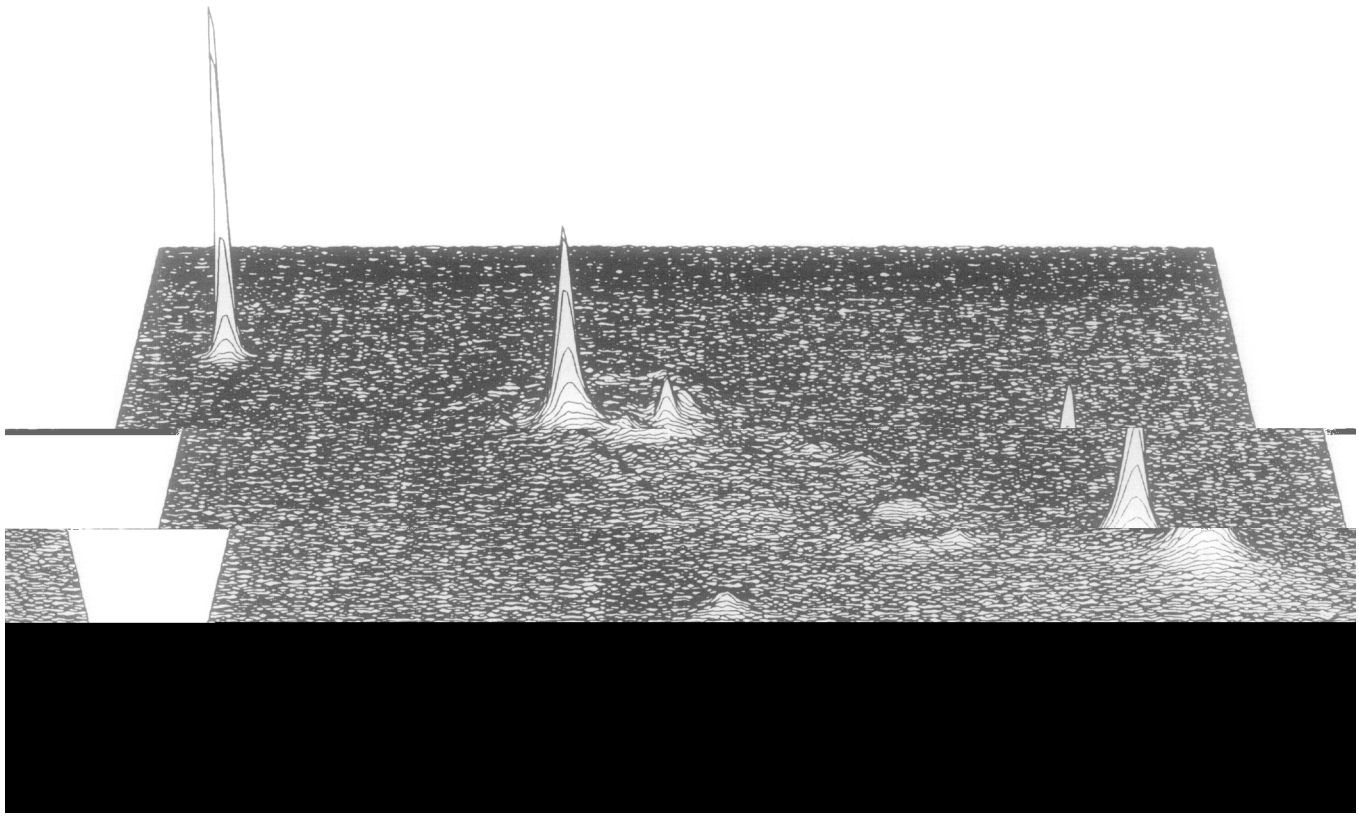

FIG. 7. Raw data map of NGC 1144.

heights of which are given in Table III. Source C is NGC 1143, source B is NGC 1144. The projected separation of the galaxies is 40 in. ($\approx 17\,kpc$) [27]. Sources A, E, and F are stars.

The following model is adopted for the error σ_μ^* of datum d_μ:

$$\sigma_\mu^* = \begin{cases} 0.2 & \text{if } d_\mu \leqslant 0 \\ 0.1 d_\mu + 0.2 & \text{if } 0 < d_\mu \leqslant 6 \\ 0.03333 d_\mu + 0.6 & \text{if } 6 < d_\mu \leqslant 12 \\ 0.01316 d_\mu + 0.8421 & \text{if } 12 < d_\mu \leqslant 50 \\ 0.01429 d_\mu + 0.7857 & \text{if } 50 < d_\mu \leqslant 86, \end{cases} \quad (3.1a)$$

where $\mu = 1, ..., M$.

Since source A is evidently the image of an isolated star, the central 25×25 pixels around it are taken as the point-spread function. This raw map of the PSF is slightly smoothed in order to make the resulting PSF rotationally symmetric. The PSF generated in this way has a full width at half maximum (FWHM) of about 2.8 pixels (see insert in Fig. 6).

Exactly as in the M87 example, the convolutions are performed circularly. Using the FFT, one transform takes about 91 ms of CPU-time. This is only 3.79 times the time which is required for one transform in the previous M87 example, where only one-fourth of the entries had to be transformed. This relative speedup is due to the fact that, in this example, vectors which are twice as long (256 as compared to 128) are to be transformed which is advantageous whenever a vector processor is used.

In order to make the algorithm convergent, a constant background $u = 10$ has to be added to each datum d_μ. Mainly two reasons are responsible for doing so. First,

TABLE III

NGC 1144: Source Heights for the Data and the Deconvolved Maps Using Various Versions of MEM

		Version of MEM						
				$E^2 + T$ with $T/M =$				
Source	Data	χ^2	E^2	5.0	4.0	3.0	2.0	1.0
A	85.41	5.17	157.23	173.08	171.48	171.48	164.35	169.64
B	50.22	10.19	88.44	98.80	101.96	101.79	100.29	103.28
C	41.47	9.59	65.47	73.10	75.16	74.93	74.28	76.07
D	13.34	4.35	17.43	19.51	19.49	19.65	18.80	18.65
E	4.96	2.60	4.60	5.02	5.63	5.69	5.62	5.96
F	3.74	2.59	2.20	3.06	3.40	3.16	3.10	3.54

Note. In each case, the artificial background is subtracted. The locations of the sources are given in Figs. 6, 8, 10, and 12.

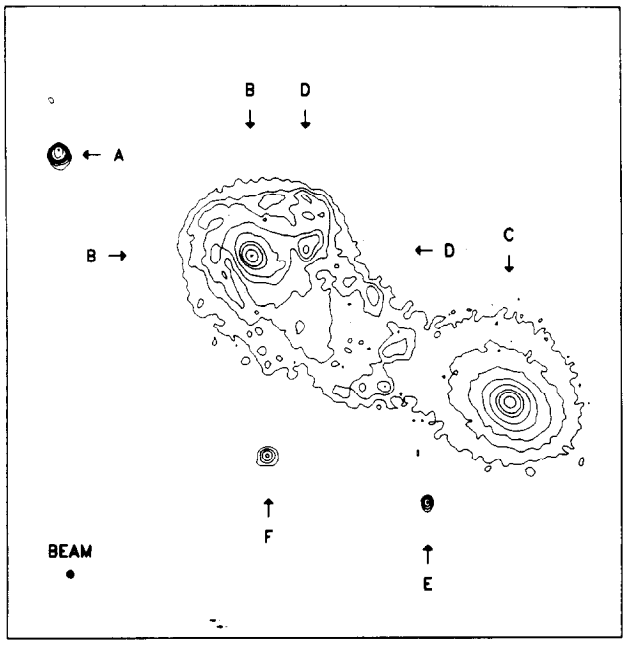

FIG. 8. Source map of NGC 1144 as deconvolved using the χ^2-version with $G_1^{\text{aim}} = M$. The contour levels are 1, 1.5, 2, 2.5, 3, 4, 5, 7.5, 10, 15, 20, 30, 40, 50, 70, 90, 110, and 160.

about 25% of the data have negative values. As already mentioned above, no solution of the equation [illegible] (1.3) may exist on grounds of the positivity requirement of the entropy functional if too many data are negative. Since the data minimum is -0.78, a constant background of $u = 0.75$ or so would, however, suffice. The second, much more important reason is the overshoot towards negative values of the deconvolved map near point-like sources. This will become clear later on when the results of the deconvolutions are discussed in more detail. In order to allow for the background u, the noise σ_μ^* of the original data map is increased by 10%, that is,

$$\sigma_\mu = 1.1\sigma_\mu^* \quad \text{for} \quad \mu = 1, ..., M, \tag{3.1b}$$

where σ_μ^* is given in (3.1a).

On theoretical grounds an expectation value $\langle E^2 \rangle_{\text{exp}} = 3.68$ of the E^2-statistic results from $M = 2^{16}$ data points [6]. However, this value can only be attained if it is possible to fit the map with no statistically significant residuals. Otherwise the value of the E^2-statistic which actually can be attained is several times greater than $\langle E^2 \rangle_{\text{exp}}$. In our case, it was necessary to smooth and symmetrize source A in accordance with the symmetric PSF.

The outcome of the deconvolutions obtained by the different versions of MEM are presented in Figs. 7 to 13. The main results concerning the numerical algorithm are given in Table II. In comparing Tables I and II it is well demonstrated that the number of iterations required to reach convergence is practically independent of the numbers N and M. This is true at least for the χ^2- and E^2-versions. The rather slow rate of convergence of the $(E^2 + T)$-version, however, is mainly due to the presence of the point source A. This indicates that, in general, the deconvolution of point-like structures is much more difficult for the $(E^2 + T)$-[illegible] numerical expense per iteration of the algorithm is roughly proportional to N and M.

The use of the χ^2-version gives only a poorly deconvolved map. The reason is probably that the (estimated) errors for high intensities are much too large. The χ^2 constraint allows a small number of residuals to be exceptionally large while the entropy gains markedly if sharp features are drastically

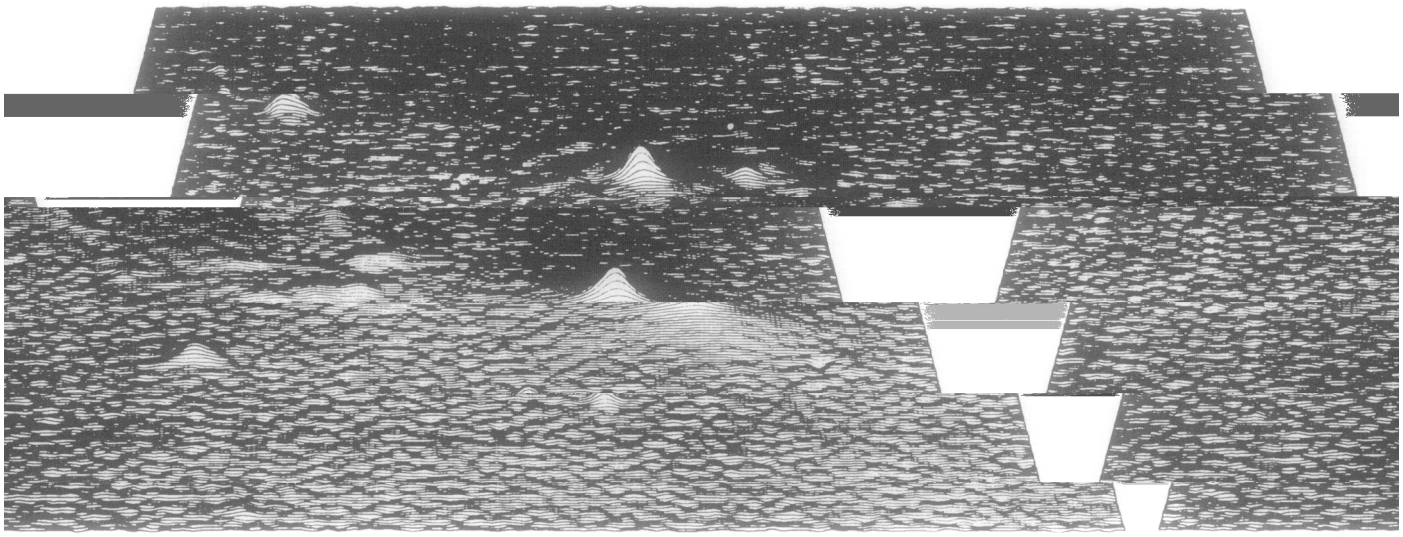

FIG. 9. Deconvolved source map of NGC 1144, χ^2-version with $G_1^{\text{aim}} = M$.

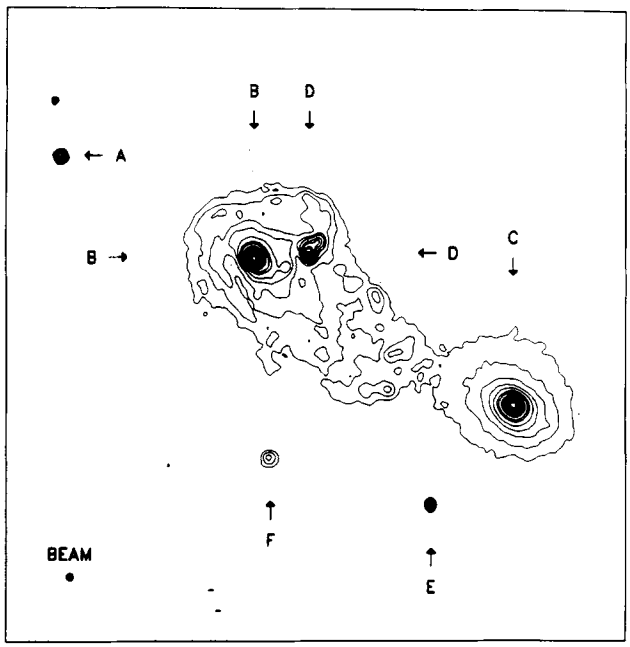

FIG. 10. Source map of NGC 1144 as deconvolved using the E^2-version with $G_1^{\text{aim}} = 3.68$. Contour levels as in Fig. 8.

reduced. The low-intensity details (spiral arms around B) are seen as well as in the other deconvolved maps. The noisy background can be explained by the fact that slight overfitting is advantageous in the χ^2-method in order to reduce χ^2 so that the residuals in the spikes can be correspondingly larger. The example demonstrates very well that not only a good choice of the errors but also a good error statistic should be used. As was the case in the above-given M87 example, the residuals r_μ are distributed markedly non-Gaussian. There are more than 100 residuals beyond eight standard deviations, the greatest residual in magnitude being $r_\mu = -25.61$. On statistical grounds, however, only four residuals beyond four standard deviations would be expected.

Contrarily, the E^2-version clearly increases in magnitude the major peaks A to D, and the background is smoothed. Sources E and F (faint stars), however, are somewhat lowered. A source just off the right to source D which is only marginally visible in the data map, is resolved more clearly in the deconvolved map. Directly above source A a point-like source shows up which is only marginally visible in the data map. Source C seems to be elongated in the direction towards source B. Spiral arms around the central source B

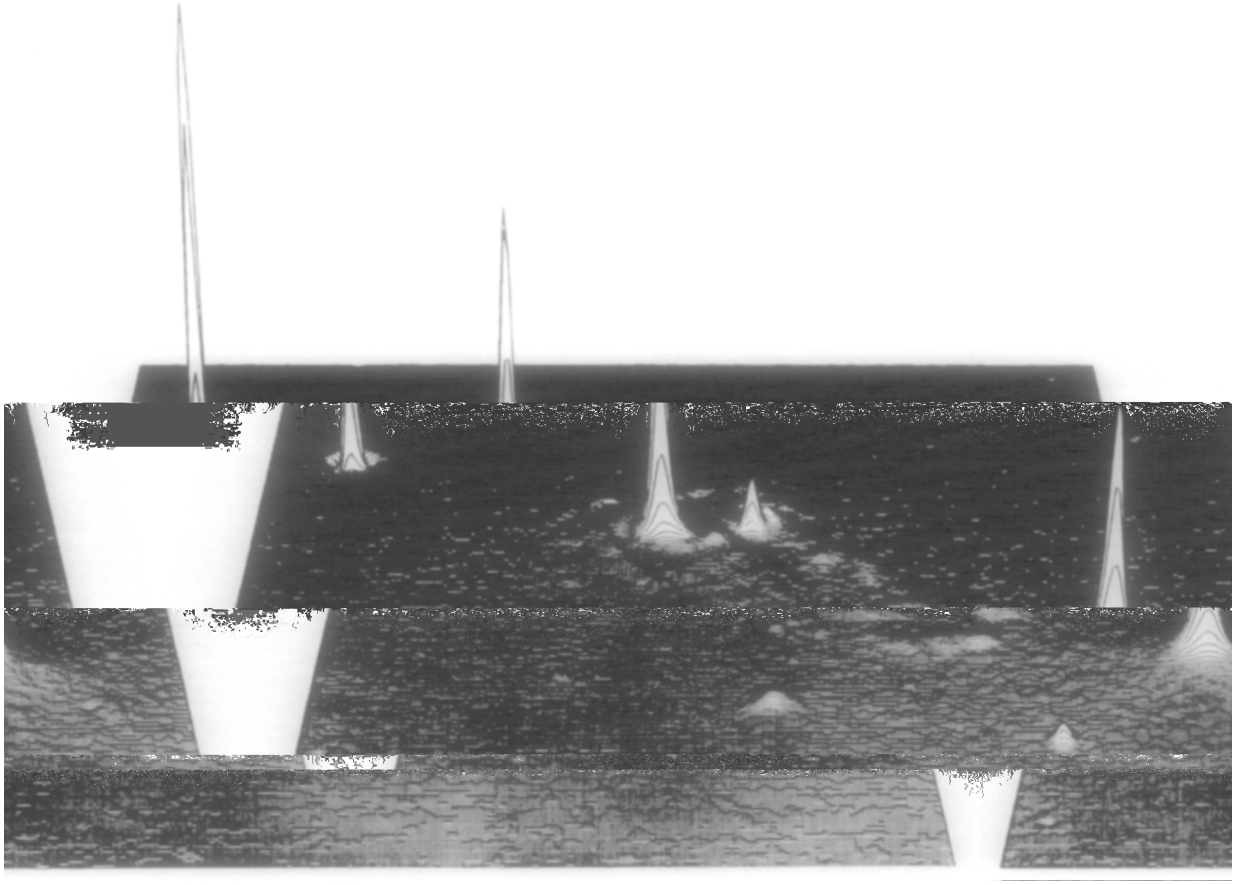

FIG. 11. Deconvolved source map of NGC 1144, E^2-version with $G_1^{\text{aim}} = 3.68$. The source A is cut at a height of 125.

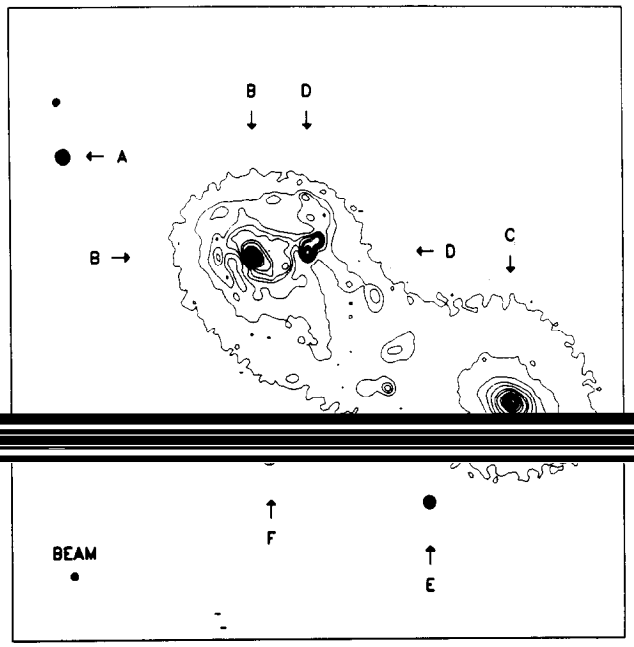

FIG. 12. Source map of NGC 1144 as deconvolved using the (E^2+T)-version with $G_1^{\text{aim}}=3.68$ and $G_2^{\text{aim}}=+4M$. Contour levels as in Fig. 8.

are clearly visible. Source A is now spread over 4×4 pixels only; that is, the size at its base is of the same order of magnitude as the FWHM of the PSF. This seems to be a very satisfactory result. However, about 60 pixels surrounding source A have values which are significantly smaller than the average level of the deconvolved map (≈ 10.47). That is, had the background u not been added to the original data, negative values would result in order to deconvolve the point source A. Evidently, this would be impossible due to the strict requirement of positivity of the entropy functional. Therefore, the addition of a constant background is mandatory in this example. To our knowledge, this is the first time where such a Gibbs-like

volutions. Constraining E only, a value $T=0.09M$ arises for the final map.

On the whole, the (E^2+T)-version yields the same outcome as the E^2-version. In some important details, however, the additional application of the T-statistic gives rise to superior results. Setting $T=4M$ we found the following improvements: First, all major peaks in the map are much more enlarged in magnitude (see Table III). Second, the final χ^2 of the map is somewhat smaller

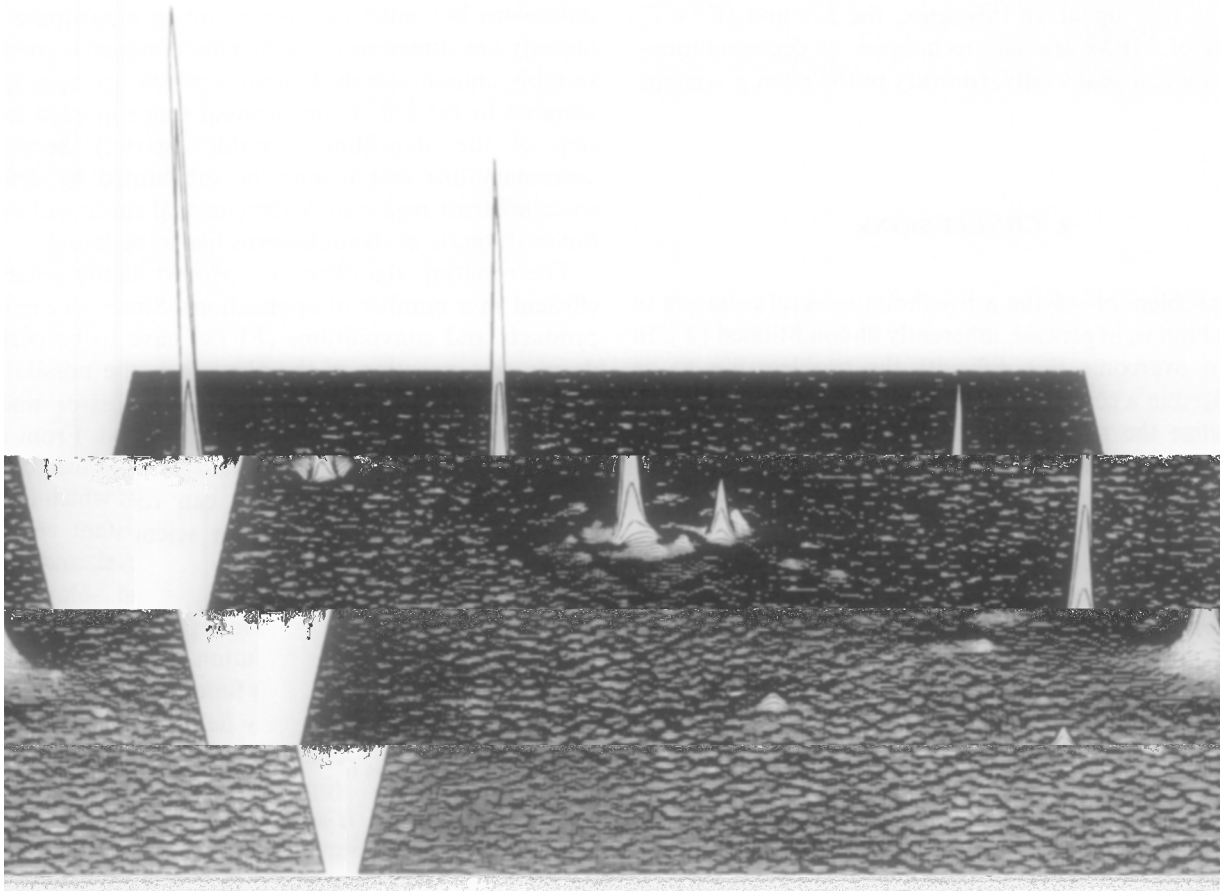

FIG. 13. Deconvolved source map of NGC 1144, (E^2+T)-version with $G_1^{\text{aim}}=3.68$ and $G_2^{\text{aim}}=+4M$. The source A is cut at a height of 125.

(6.54×10^4 as compared to 6.56×10^4) which indicates that the T-statistic forces the model values m_μ to a better fit of the data. Third, the correlation of the residuals with the model values, which is expected to be zero for a good fit, is significantly smaller (-0.225 as compared to -0.313). Altogether, this yields a deconvolved map with an improved resolution as compared to the E^2-version. As a result, a source amidst peaks B and D, which is only marginally visible in the data map, becomes clearly visible. This is also true for the spiral arms around the central source B. As can be seen in Table III, a further reduction of T does improve the deconvolved map only marginally.

Interestingly, the E^2-version, as well as the $(E^2 + T)$-version, gives rise to a value of $\chi^2 \approx M$, that is, the same value to which χ^2 was constrained in the χ^2-version. The peaks, however, are much more enlarged in magnitude in the E^2- and $(E^2 + T)$-versions, respectively, as compared to the χ^2-version. Therefore, the much better resolution and enhancement of the peaks is not due to a significantly smaller value of χ^2 as might be expected, but, rather, they are due to forcing the residuals to have their correct statistical distribution. Of course, constraining χ^2 to a smaller value than M could also make the peaks match the data more closely, but this will lead to spurious resolution elsewhere in the map, with noise on the data being interpreted as true signal. In this sense, the E^2- and $(E^2 + T)$-versions of MEM are safe techniques of deconvolutions, enhancing real peaks only, contrary to the plain χ^2-version.

4. CONCLUSIONS

The problem of solving a Fredholm integral equation of the first kind is, in general, inherently ill-conditioned [1]. In order to overcome this difficulty the problem has to be regularized in a certain respect. An elegant way to do this is to stabilize the solution via a functional which is to be optimized and to introduce constraints into the problem by virtue of which it is ensured: (i) that the reconstructed map is consistent with the actually measured data within the noise level, and (ii) that all a priori knowledge given about the true solution, e.g., positivity, is not violated. Even though many functionals can be used in order to regularize the solution [26, 28], during the last two decades the entropy functional (1.3a) has found widespread application in many fields of science. At least two reasons seem to be responsible for the great popularity of MEM. First of all the entropy functional (1.3a) automatically guarantees a positive solution everywhere in the reconstructed map. If, for example, positivity of the map is mandatory, this implies that the number of constraints can be exorbitantly reduced and, clearly, the computational expense of the method is diminished by the same extent. For most practical applications of MEM, one or two constraints suffice in order to make the reconstructed map statistically consistent with the measured data set [6]. Second, MEM seems to be unsurpassed by other methods which are currently in use as concerns noise suppression and smoothness of the deconvolved map. A serious drawback of MEM, however, is that in real life applications, e.g., two-dimensional deconvolutions, the resulting constraint optimization problem is a large-scale one with up to $N = 10^6$ unknowns to be determined. Of course, conventional algorithms for solving nonlinear constraint optimization problems cannot be applied because the handling and inversion of matrices with N^2 elements would then actually be required which is, obviously, beyond the possibilities even of very large modern computer systems.

In order to surmount these difficulties, therefore, we propose a new numerical algorithm which is based on so-called variable metric methods. This class of methods has proved very efficient in solving small-scale nonlinear constraint problems. Due to the large-scale nature of the optimization problem encountered in MEM, however, these methods cannot be applied directly but, rather, must be modified in two respects. First, the optimization process cannot be performed in the full N-dimensional space of the unknowns but must be carried out in a subspace of sufficiently low dimension $n_s \ll N$. This subspace is spanned by suitably chosen search directions which are appropriately adapted to the full N-dimensional space in each iteration step of the algorithm (variable metric!). Second, the customary line search must be substituted by defining a so-called trust region in N-dimensional space within which the next iterate of the unknowns has to be found.

The resulting algorithm has proved highly reliable and efficient in a number of applications. Since very many dot products and convolutions (FFTs) have to be performed during the execution of the algorithm, the possibilities of modern vector computers and multiprocessor machines, e.g., a CRAY Y-MP, can be fully exploited. From a computational point of view, therefore, deconvolution no longer seems to be a time consuming task but can routinely be applied in solving inverse problems in science and engineering.

APPENDIX

A solution of subproblem (2.17) may be found as follows:

(1) Determine $\rho_{\min}$ such that $\delta_i + 2\rho_{\min} > 0$ for $i = 1, \ldots, 6$. This step turns (2.17) into a meaningful problem, since occasionally it may happen that some of the δ_i are negative.

(2) Solve problem

$$\begin{aligned} \tfrac{1}{2}\mathbf{x}^{\mathrm{T}}\,\mathrm{diag}(\delta_1+2\rho_{\min},\ldots,\delta_6+2\rho_{\min})\,\mathbf{x}+\mathbf{b}^{\mathrm{T}}\mathbf{x}&=\min,\\ \mathbf{c}_1^{\mathrm{T}}\mathbf{x}+g_1&=0,\\ \mathbf{c}_2^{\mathrm{T}}\mathbf{x}+g_2&=0. \end{aligned} \tag{A.1}$$

If the solution $\mathbf{x}$ of problem (A.1) satisfies (2.17b) then proceed with step 5.

(3) Solve problem

$$\begin{aligned} \mathbf{x}^{\mathrm{T}}\mathbf{x}&=\min,\\ \mathbf{c}_1^{\mathrm{T}}\mathbf{x}+g_1&=0,\\ \mathbf{c}_2^{\mathrm{T}}\mathbf{x}+g_2&=0. \end{aligned} \tag{A.2}$$

Problem (A.2) results from (2.17a) for $\rho\to\infty$. Consequently, a solution of (2.17) can be found with $\rho\in[\rho_{\min},\infty)$, provided that problem (A.2) has a solution $\mathbf{x}$ which satisfies (2.17b). Having this value of ρ determined, solve problem (2.17a) and proceed with step 5. On the other hand, if the solution of (A.2) does not fulfill (2.17b), then no feasible point of (2.17a) satisfies (2.17b), and the algorithm proceeds with the following step.

(4) Solve problem

$$\left.\begin{aligned} &\mathbf{x}^{\mathrm{T}}\mathbf{x}=\min\\ &\mathbf{c}_1^{\mathrm{T}}\mathbf{x}+(1-\xi_1)\,g_1=0\\ &\mathbf{c}_2^{\mathrm{T}}\mathbf{x}+(1-\xi_2)\,g_2=0 \end{aligned}\right\}\quad\text{such that}\quad\begin{cases}\xi_1^2+\xi_2^2=\min\\ R/F_0-\mathbf{x}^{\mathrm{T}}\mathbf{x}=0.\end{cases} \tag{A.3}$$

Problem (A.3) is equivalent to problem (A.2) apart from the modification of constraints g_1 and g_2 such that $R/F_0-\mathbf{x}^{\mathrm{T}}\mathbf{x}=0$ can be obtained.

(5) End of the solution of subproblem (2.17); proceed with the next major step of the algorithm.

Although this solution procedure for subproblem (2.17) involves substantial programming, the computation time required for its execution is negligible in comparison to the other parts of the algorithm, above all to the required image–data and data–image transformations. In all test cases solved so far, the procedure turned out to be of utmost reliability.

ACKNOWLEDGMENTS

I thank Professor R. Bulirsch for making possible this work and for continuous support, Professor J. Pfleiderer for several useful discussions and for meticulously reading the whole manuscript, Professor K. Schittkowski for introducing me into the variable metric methods, Professor D. Smith for carefully reading parts of the paper, an anonymous reviewer for improving the manuscript, Dr. L. Walsh for checking the English, and the Leibniz–Rechenzentrum in Munich for granting the computer time. Dr. J. Skilling kindly provided me with the M87 data, and Mr. R. Vio provided me with the NGC 1143/44 data. This research was funded by the Deutsche Forschungsgemeinschaft (DFG). Preliminary work for it, which was done at the University of Innsbruck, was financed by the Austrian Fonds zur Förderung der wissenschaftlichen Forschung.

REFERENCES

1. A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-posed Problems* (Winston, Washington, DC, 1977).
2. S. F. Gull and G. J. Daniell, *Nature* **272**, 686 (1978).
3. M. C. Kemp, *Med. Radionuclide Imaging* **1**, 313 (1980).
4. F. M. Larkin, *BIT* **9**, 30 (1969).
5. B. R. Frieden, *J. Opt. Soc. Am.* **62**, 511 (1972).
6. J. Reiter and J. Pfleiderer, *Astron. Astrophys.* **166**, 381 (1986).
7. E. T. Jaynes, *Phys. Rev.* **106**, 620 (1957).
8. E. T. Jaynes, *IEEE Trans. Syst. Sci. Cyb.* **SSC-4**, 227 (1968).
9. E. T. Jaynes, in *The Maximum Entropy Formalism*, edited by R. D. Levine and M. Tribus (MIT Press, Cambridge, MA, 1979), p. 15.
10. E. T. Jaynes, *Proc. IEEE* **70**, 939 (1982).
11. J. Skilling, in *Proceedings, Fourth Maximum Entropy Workshop, Calgary, 1984*, edited by J. H. Justice (Cambridge Univ. Press, Cambridge, UK, 1986), p. 156.
12. S. F. Burch, S. F. Gull, and J. Skilling, *Comput. Vision Graphics Image Process.* **23**, 113 (1983).
13. J. G. Ables, *Astron. Astrophys. Suppl.* **15**, 383 (1974).
14. R. K. Bryan and J. Skilling, *Mon. Not. R. Astron. Soc.* **191**, 69 (1980).
15. J. Reiter, Ph.D. thesis, University of Innsbruck (1985).
16. B. R. Frieden and D. C. Wells, *J. Opt. Soc. Am.* **68**, 93 (1978).
17. J. Skilling and R. K. Bryan, *Mon. Not. R. Astron. Soc.* **211**, 111 (1984).
18. N. Agmon, Y. Alhassid, and R. D. Levine, *J. Comput. Phys.* **30**, 250 (1979).
19. J. Eriksson, *Math. Programming* **18**, 146 (1980).
20. R. B. Wilson, Ph.D. thesis, Harvard University, Boston (1963).
21. M. J. D. Powell, in *Proceedings, Biennial Conference, Dundee, 1977*, edited by G. A. Watson, Lecture Notes in Mathematics, Vol. 630 (Springer-Verlag, Berlin, 1978), p. 144.
22. K. Schittkowski, *Numer. Math.* **38**, 83 (1981).
23. J. Skilling, Algorithms and applications, in *Workshop on Maximum Entropy Estimation and Data Analysis, University of Wyoming, 1981* (unpublished).
24. G. R. Walsh, *Methods of Optimization* (Wiley, London, 1975).
25. H. Arp and J. Lorre, *Ap. J.* **210**, 58 (1976).
26. J. Pfleiderer, *Astron. Astrophys.* **194**, 344 (1988).
27. M. Joy and F. D. Ghigo, *Ap. J.* **332**, 179 (1988).
28. R. Nityananda and R. Narayan, *J. Astrophys. Astron.* **3**, 419 (1982).