Mathematical Programming 8 (1975) 1–19.
North-Holland Publishing Company

A RELAXATION METHOD FOR RECONSTRUCTING OBJECTS FROM NOISY X-RAYS*,**

G.T. HERMAN
State University of New York at Buffalo, Amherst. New York, U.S.A.

Received 19 December 1973

An algorithm is presented for estimating the density distribution in a cross section of an object from X-ray data, which in practice is unavoidably noisy. The data give rise to a large sparse system of inconsistent equations, not untypically 10^5 equations with 10^4 unknowns, with only about 1% of the coefficients non-zero. Using the physical interpretation of the equations, each equality can in principle be replaced by a pair of inequalities, giving us the limits within which we believe the sum must lie. An algorithm is proposed for solving this set of inequalities. The algorithm is basically a relaxation method. A finite convergence result is proved. In spite of the large size of the system, in the application area of interest practical solution on a computer is possible because of the simple geometry of the problem and the redundancy of equations obtained from nearby X-rays. The algorithm has been implemented, and is demonstrated by actual reconstructions.

1. Introduction

The problem of reconstructing a three-dimensional distribution from its two-dimensional projections (shadowgraphs) has arisen in a large number of scientific and medical areas. The basic mathematical problem was posed as early as 1917 by Radon [19] who has actually provided a closed form solution assuming ideal mathematical conditions. In 1956 Bracewell [1] considered this problem in relation to strip integration in radio astronomy. In 1963 Cormack [3] pointed out the importance of the problem in radiology, and in 1968 DeRosier and Klug [4] attacked it from the point of view of electron microscopy. Many different methods have been proposed in these and other fields; for a survey, see [10].

* A preliminary version of this paper has been presented at the VIII International Symposium on Mathematical Programming, Stanford University, Stanford, California.

** This research has been supported by N.S.F. Grant G.J. 998.

In this paper we propose a method which has been designed with special attention to the fact that in practical applications the data supplied to a reconstruction method is invariably noisy. The paper has three parts. In the first we show how the noisy data gives rise to a system of inequalities regarding the densities of the object. We propose a relaxation method (an iterative algebraic reconstruction technique) to solve the system of inequalities. We prove that, provided there is a solution to the system at all, our method will find such a solution in a *finite* number of iterations. In the second part we describe a computer program based on the method introduced in the first part. In the final part we report on the performance of our algorithm and compare it to three previously proposed algorithms, the ART2 algorithm of [14], the SIRT algorithm of [7], and the convolution method of [20]. We find that the algorithm proposed in this paper is often advantageous to the other methods in case of noisy data. A reconstruction using this method of a canine left ventricle from X-ray data is also shown.

2. Mathematical theory

Although this is by no means necessary for our method, for the sake of simplicity of explanation, we shall assume that the reconstruction of a three-dimensional object is to be done by a series of reconstructions of two-dimensional slices. In other words, we assume that in the reconstruction process the data has been obtained from X-rays which have been collimated to lie in one plane. From this data the density distribution in that plane is estimated, and the density distribution of the object is obtained by a sequence of such reconstructions for parallel planes. (Although we shall use the word density throughout this paper, in the X-ray application it is the distribution of the coefficient of X-ray absorption that is being calculated. In other applications of the reconstruction problem it is still other physical entities which take the place of "density".)

Consider Fig. 1. If we send an X-ray through the object, from the relative intensity of the X-ray at the source and at the detector we can estimate the total X-ray absorption along the path of the ray. A statement of this type is correct (with small variations) whether or not we assume the X-ray to be infinitely thin.

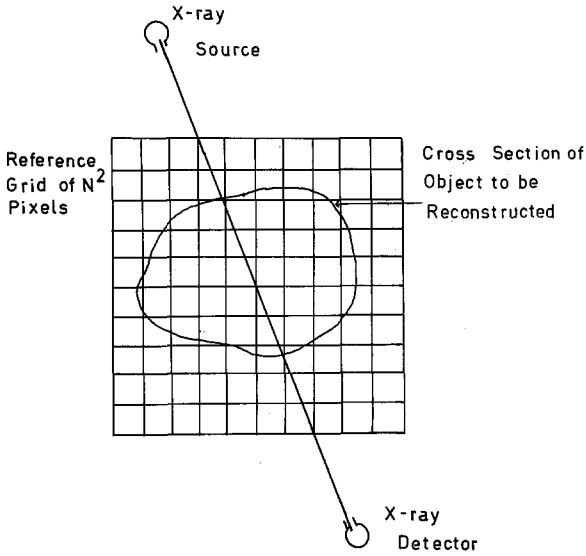

Fig. 1. Geometry for the derivations of the equations (1) which govern the reconstruction of a cross section from its X-ray projection.

Let us assume for computational purposes that the cross section to be reconstructed is enclosed in a square which is subdivided into N^2 little squares (pixels). Let us also assume (and this is an important assumption) that the density within each pixel is constant. Then we may write

$$\sum_{j=1}^{N^2} a_{i,j}x_j = b_i, \qquad 1 \leqslant i \leqslant M, \tag{1}$$

where x_j is the (unknown) density of the j^{th} pixel, $a_{i,j}$ is a constant of proportionality of the contribution of the j^{th} pixel to the i^{th} ray (for example, it can be taken to be proportional to the length of the intersection of the i^{th} ray with the j^{th} pixel), and b_i is calculated on the basis of the estimated total X-ray absorption along the i^{th} ray. For the sake of simplicity of the mathematical discussion we assume that all the equations in (1) are normalized, i.e., that

$$\sum_{j=1}^{N^2} (a_{i,j})^2 = 1, \qquad 1 \leqslant i \leqslant M. \tag{2}$$

(Such normalization is not necessary. Our algorithms are valid, with small alternations, without this assumption.)

Since the $a_{i,j}$ can be calculated geometrically and the b_i can be calculated on the basis of our experimental measurements, it appears that equations (1) may now be simply solved by the techniques of linear algebra. This is however not so. Apart from the size of the problem (quite possibly $N = 100$ and $M = 100\,000$) we are faced with the problem that the equations are noisy, and hence more than likely inconsistent. There are three different types of noise involved:

(i) the b_i are based on physical measurements which are invariably and unavoidably noisy,

(ii) our assumption that the density in each pixel is uniform is unlikely to be perfectly valid,

(iii) the calculation of the $a_{i,j}$ usually involves some idealization of the nature of X-rays (e.g., we usually ignore scattering).

It appears therefore that a precise solution to equations (1) is generally impossible. Even if such a solution existed, it does not necessarily represent the original distribution, due to (i)–(iii) above. Rather than trying to satisfy equations (1), it appears more reasonable to try to find a solution to the following set of inequalities:

$$b_i - \epsilon_i \leqslant \sum_{j=1}^{N^2} a_{i,j} x_j \leqslant b_i + \epsilon_i, \qquad 1 \leqslant i \leqslant M. \tag{3}$$

Here we have taken care of the noise by requiring the equations to be satisfied only within a certain tolerance ϵ_i. The ϵ_i can be estimated either based on our knowledge of the method of data collection, or experimentally. For example, in X-ray reconstruction, repeated measurements for the same geometric ray can help us to estimate the noise due to (i). In practical applications the choice of the ϵ_i is most important. If they are chosen too small, the original distribution might not satisfy the inequalities (3), or even worse, there may even be no solutions at all. If the ϵ_i are chosen too big, the inequalities provide us with very little information about the object to be reconstructed.

There is also another practical problem. If there is one solution to the inequalities (3), then there are usually many solutions, and the question arises which one to choose. We defer the discussion of this problem until the next section. In this section we restrict our attention to finding

an algorithm which, given a set of inequalities (as in (3)), will find a solution, if there is one, to this set of inequalities.

There are already algorithms in the published literature to do exactly this. In fact our method is based on a finitely convergent relaxation procedure of Motzkin and Schoenberg [18]. An explanation why we prefer our procedure will be given after the algorithm is described and our main theorem is proved.

In our description of the algorithm and in the proof we use standard mathematical notation. In particular, given two N^2-dimensional vectors $a = (a_1, \ldots, a_{N^2})$ and $x = (x_1, \ldots, x_{N^2})$, we use (a, x) to denote $\Sigma_{j=1}^{N^2} a_j x_j$, and we use $\|x\|$ to denote $\sqrt{(x, x)}$. Our algorithm produces a sequence $x^0, x^1, x^2, \ldots$ of N^2-dimensional vectors. For the purpose of the mathematical discussion, x^0 is *arbitrary*.

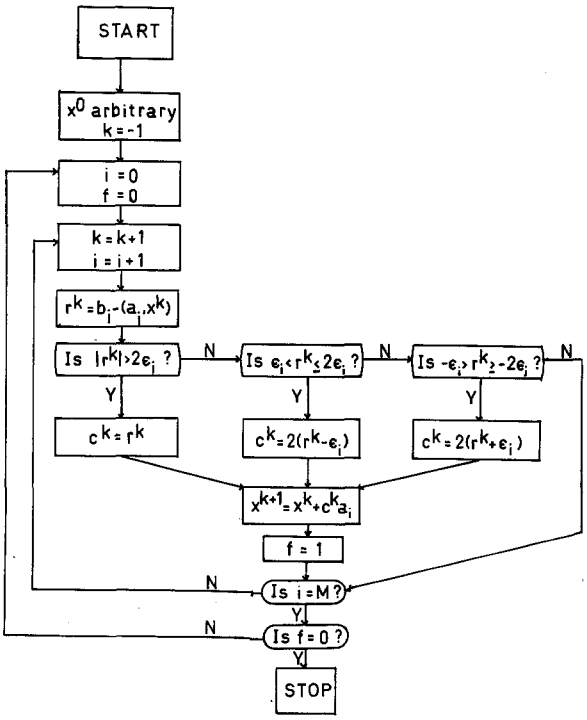

Fig. 2. Flowchart of the mathematical algorithm for the reconstruction of objects from their projections.

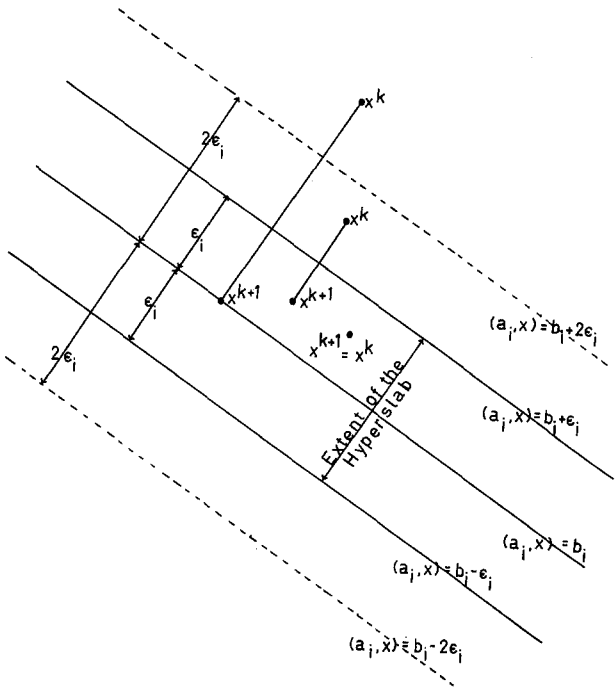

Fig. 3. Geometrical interpretation of the mathematical algorithm described in Fig. 2.

The algorithm to find a solution of (3), assuming the normalization condition (2), is described by the flowchart in Fig. 2.

What is described in this flowchart has a simple geometric interpretation. Each of the equations in (1) determines a hyperplane in an N^2-dimensional space. A point x is in the i^{th} hyperplane if and only if it satisfies the i^{th} equation. Similarly, each of the inequalities in (3) determines a "hyperslab" of thickness $2\epsilon_i$ (see Fig. 3).

The vectors x^0, x^1, x^2, ..., x^k, ... are produced as follows. x^0 is arbitrary. Then one-by-one we consider the inequalities in (3) for $1 \leqslant i \leqslant M$. The variable f is a flag whose purpose will become clear below. In producing x^{k+1} from x^k, we distinguish between three cases (see Fig. 3).

(i) x^k is more than $2\epsilon_i$ away from the i^{th} hyperplane. In this case, x^{k+1} is obtained by dropping a perpendicular from x^k onto the i^{th} hyperplane.

(ii) x^k lies outside the i^{th} hyperslab, but not more than $2\epsilon_i$ from the central hyperplane. In this case, x^{k+1} is obtained by reflecting x^k in the nearer face of the hyperslab.

(iii) x^k lies within the hyperslab. In this case, $x^{k+1} = x^k$.

In the first two cases we set the flag f equal to 1, to indicate that a change has been made. We repeat this process for all the inequalities in (3). When we are done, $f = 0$ if and only if only (iii) has been used, which happens if and only if all the inequalities are satisfied. In this case we stop, for we have found a solution to (3). Otherwise, we set f equal to 0 again, and we start the same procedure with the first inequality. Our principal result is that this procedure is guaranteed to terminate with a solution to (3), provided only that the ϵ_i are large enough.

Theorem. *If the set K of all points which satisfy the inequalities* (3) *is full dimensional (i.e., not a subset of any hyperplane), then the algorithm described above converges in a finite number of steps to an element of K.*

Proof. If the algorithm stops, the x^k at that time is clearly an element of K. Otherwise, the algorithm provides us with an infinite sequence $x^0, x^1, x^2, \ldots$ of points. We shall now show that the latter case is impossible, since it leads to a contradiction.

It is easy to prove (see Fig. 3) that for any point z in the i^{th} hyperslab it is the case that

$$\|x^{k+1} - z\| \leqslant \|x^k - z\|. \tag{4}$$

(Here we assume that for the k^{th} step in the algorithm the i^{th} hyperslab is used.) Since K is a subset of all the hyperslabs, the inequality in (4) holds for all points z in K and for all k. From this fact one can prove (see, e.g., [18, Lemma 1, Case 1] that if the sequence x^k is infinite, then it converges to a point x^*. Clearly, x^* is in K.

Consider a spherical neighborhood N of x^*, which is such that among all the hyperplanes of the form $(a_i, x) = b_i \pm c\epsilon_i$ $(1 \leqslant i \leqslant M, c \in \{-1, 0, 1\})$ only those intersect N which go through x^*. Since x^k converges to x^*, there exists a t such that if $k \geqslant t$, then $x^k \in N$.

Assume $k \geqslant t$ and $x^{k+1} \neq x^k$. By the choice of N and t, x^{k+1} must be a reflection of x^k in some plane through x^*, and so $\|x^{k+1} - x^*\| = \|x^k - x^*\|$. It follows that for all $k \geqslant t$, $\|x^k - x^*\| = \|x^t - x^*\|$. Since the x^k converges to x^*, this implies that $x^k = x^*$ for all $k \geqslant t$, contradicting the assumption that the algorithm does not halt.

This completes the proof of the Theorem. We wish to complete this section with a brief discussion of the reasons for proposing the algorithm described above. After all, it is known (see, e.g., [18]) that if we used reflection in both cases (i) and (ii) above our theorem would remain valid.

A general reason for not using reflection at every step is that Goffin [8, p. 115] has found that for rapid convergence it is a good strategy to use projections when we are far away from the hyperslab and reflections when we are near it.

More specifically, there are good intuitive reasons in the case of reconstruction not to use reflections all the time. Each step in the algorithm is a readjustment of the densities in those pixels which are intersected by the ray under consideration. If the density along the ray of the distribution proposed after the k^{th} step is higher than the measured density along that ray, it is perfectly reasonable to lower the densities in the pixels, but it makes no sense to lower them so much that the density along the ray after the $(k+1)^{\text{st}}$ iteration would actually become lower than the measured density. Intuition indicates and experience confirms that this would lead to distributions which will look "noisy" with a salt and pepper type of noise. We shall return to this point in the next section.

Finally, we wish to point out why our algorithm does not use projections onto the hyperplanes in every step. Such an algorithm has been proposed by us [9, see also 14]. It is clear that in the case when the equations (1) are inconsistent, such an algorithm cannot possibly converge. However, if we look only at every M^{th} x^k (i.e., the sequence x^M, x^{2M}, x^{3M}, ...), this sequence does converge [14, Theorem 2U]. The problem is, as was pointed out by Gilbert [7] and confirmed by us [14], that the x^* it converges to is usually a bad reconstruction in case of noisy data. In [14], it is demonstrated that early values of x^k are reasonably good reconstructions, and a method for selecting an appropriate k is proposed. However, as we shall see in the last section, for noisy data the algorithm proposed in this paper is consistently better than the algorithm proposed in [14].

3. Computer implementation

The algorithm we have implemented on the computer is based on but

is not identical to the mathematical algorithm described in the last section. There are a number of practical reasons for this.

First of all, it was found that in general the algorithm converges very rapidly. In particular, even though the finite convergence may have taken as many as $20M-30M$ steps, there was insignificant difference between the final result and the result obtained after the first $3M-4M$ steps. It makes therefore no sense computationally to run the algorithm until finite convergence is achieved, it should be stopped much earlier.

In trying to insure that our final result is a good reconstruction, our computer algorithm incorporates the following features:

(i) an appropriate choice of x^0,

(ii) variable tolerance,

(iii) constraining.

We now discuss these features one by one.

It is clear that the choice of x^0 is important. It is intuitively reasonable and easy to show that, in general, the nearer x^0 is being to a solution of (3), the faster the algorithm converges. Further, the choice of x^0 influences the final result. This brings us back to the question: which of the many solutions of the inequalities (3) we should be aiming at?

It has been proposed in [6] and further explained in [14] that it is reasonable to look for the solution with smallest variance, since such a solution is least likely to contain artifacts not forced on us by the data. It is shown in the Appendix of [14] that mathematically this is equivalent to looking for the minimum norm solution, i.e., the x for which $\|x\|$ is as small as possible. Although our algorithm does not actually find the minimum norm solution, we can insure that the solution we find is nearly minimum norm, by starting with an x^0 which has a small norm, and approaching our solution by making small changes in each step. This is a reason why reflection is not used in every step of the algorithm.

It is also a reason for using variable tolerance. In order to insure that the changes in the individual steps are small, it is reasonable to have a high tolerance in the early steps of the execution of the algorithm and reduce the tolerance as we go along. It is sufficient to consult Fig. 3 to see that this will have the desired effect.

The third feature we wish to discuss is the making use of the fact that in nearly all application areas the density values x_j cannot be negative. In many cases it is also known that x_j cannot be greater than a certain

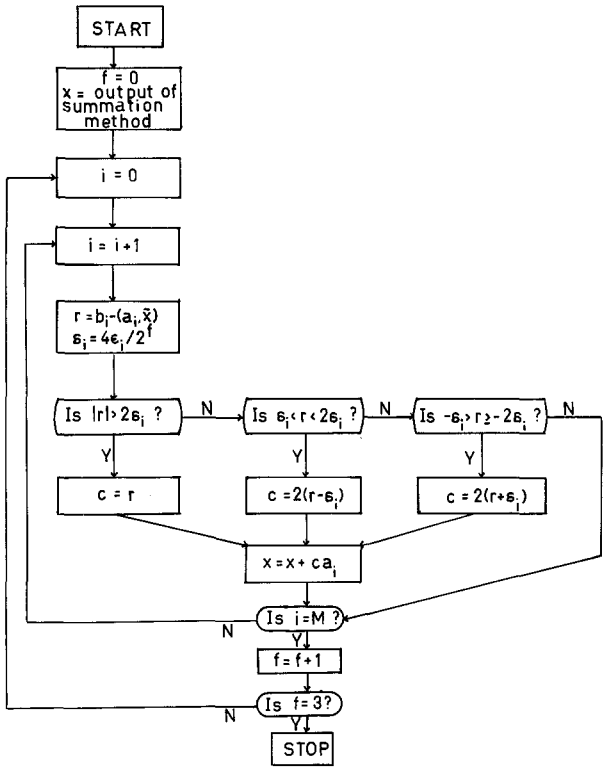

Fig. 4. Flowchart of the computer implemented algorithm for the reconstruction of objects from their projections.

positive number. Such information gives rise to additional inequalities which can be made use of during the execution of the algorithm in various ways. For example, one may simply set x_j equal to 0 whenever its value becomes negative. However, there are some better ways, see for example the ART2 algorithm in [14].

There are a number of ways of incorporating these principles into an algorithm. The flowchart in Fig. 4 describes the method used in our tests and comparisons described in the next section. In this implementation, x^0 is the output of the so-called summation method, as described and demonstrated in [12]. This is a not very accurate but extremely fast reconstruction method which produces very smooth reconstructions (see [12]). Hence it is a cheaply obtainable and good starting point according to the criteria given above.

We also make use of the notation that if $x = (x_1, ..., x_{N^2})$, then $\tilde{x} = (\tilde{x}_1, ..., \tilde{x}_{N^2})$, where for $1 \leqslant j \leqslant N^2$,

$$\tilde{x}_j = \begin{cases} 0 & \text{if } x_j < 0, \\ x_j & \text{if } 0 \leqslant x_j \leqslant U, \\ U & \text{if } x_j > U, \end{cases} \tag{5}$$

where U is the maximum possible value of the density in the object. Our estimate of the distribution is not the vector x at the time the algorithm stops, but rather the corresponding vector $\tilde{x}$. We also drop the superscript k in the description of the algorithm. δ_i denotes the variable tolerance.

In this implementation the algorithm (starting with the output of the summation method) runs for $3M$ steps. The tolerance δ_i reduces from $4\epsilon_i$ in the first M steps to $2\epsilon_i$ in the second M steps, to ϵ_i in the last M steps. Constraining is taken care of by using $\tilde{x}$ instead of x at the appropriate point (see [14] for a justification).

In spite of the very large size of the systems which one may have to deal with in many of the application areas, the computer algorithm described in Fig. 4 can obtain solutions surprisingly cheaply. The most time consuming part in the algorithm is the taking of the inner product $(a_i, \tilde{x})$, which is in the inner loop of the flowchart. However, due to the sparseness of the a_i, this is not really very expensive. In fact, in our implementation the a_i are not stored at all, but its nonzero components are recalculated every time from the simple geometry which determines the a_i (see [9, Appendix A] for a similar case). In the X-ray reconstruction application we are further helped by the fact that nearby X-rays give approximately the same information, and so by an appropriate ordering of the equations one need not even do as many as $3M$ steps. Thus the reconstruction shown in Fig. 5, which involved over 4000 unknowns ($N = 64$) and over 10 000 equations, can be carried out for a cost which is of the order of one dollar.

Finally, we wish to remark that the principles described in this paper have also been made use of in another computer algorithm which was especially designed to deal with binary distributions. Such distributions are encountered in electron microscopy and some areas of medicine. The performance of that algorithm has been reported on in [13].

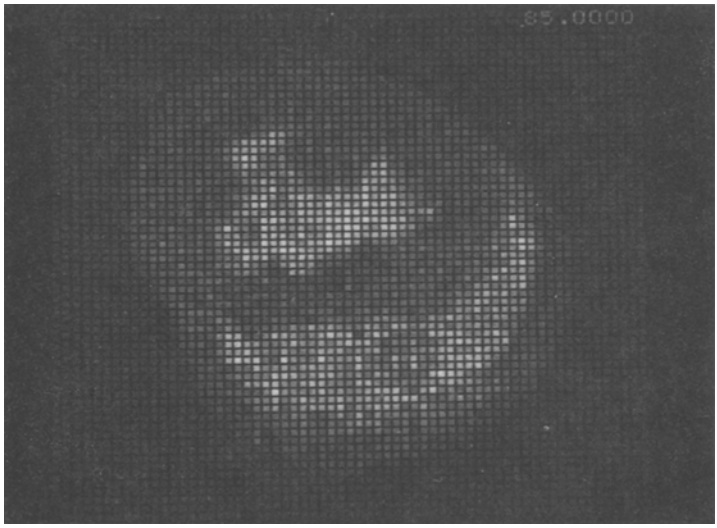

Fig. 5. Reconstruction of a cross section of a canine heart filled with barium paste. The ring inside one of the ventricles is an inserted catheter. The output has 64^2 pixels. The X-ray data has been collected using divergent rays from a source which has been moved around a circle in 35 equal steps which altogether spanned 183.6 degrees. Each position of the source gave rise to 300 measurements (M = 10 500), and the reconstruction shown has been obtained after 8 400 steps, i.e., before all the data has been used. The redundancy of nearby X-rays has been made use of by estimating the (reducing) tolerance on the basis of local variations in the projection data.

4. Results

The method described in this paper has been implemented and used for the reconstruction from X-rays of cross sections of a beating heart (see [16], [17] or [21] for more detailed reports of this application), as well as for reconstructions from ultrasound projections [11]. A reconstruction like the one shown in Fig. 5 gives one some idea of the performance of the method, especially since the original heart can be sliced and the reconstruction may be compared to the original (see [16, 17, 21]). However, for a precise evaluation of the algorithm, it appears to be more appropriate to study its performance on test patterns and computer generated data, provided that the data generation is done carefully to reflect the type of situation which may arise in applications.

We have therefore chosen 4 test patterns, computer printer produced

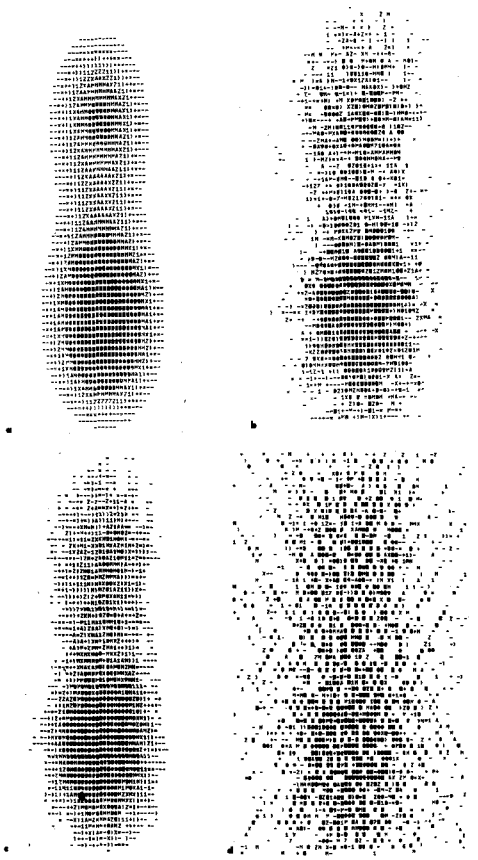

Fig. 6. A 64 × 64 reconstruction of the heat distribution described in equation (6). All reconstructions are from the same data set: 16 projections in a full range with parallel rays and Gaussian white noise with 10% standard deviation.

(a) Original distribution.
(b) Reconstruction using ART2.
(c) Reconstruction using SIRT.
(d) Reconstruction using convolution.

half-tone displays of which are shown in Fig. 6a, and Fig. 8a, b, c. These test patterns have been used by us before to compare algorithms [15]. All but the last (Fig. 8c) have been originally proposed by other authors. For example, the pattern of Fig. 6a has been used by Sweeney [22] in his holographic interferometric reconstruction studies as an example of a likely distribution of heat in a cross-section of a liquid which is being

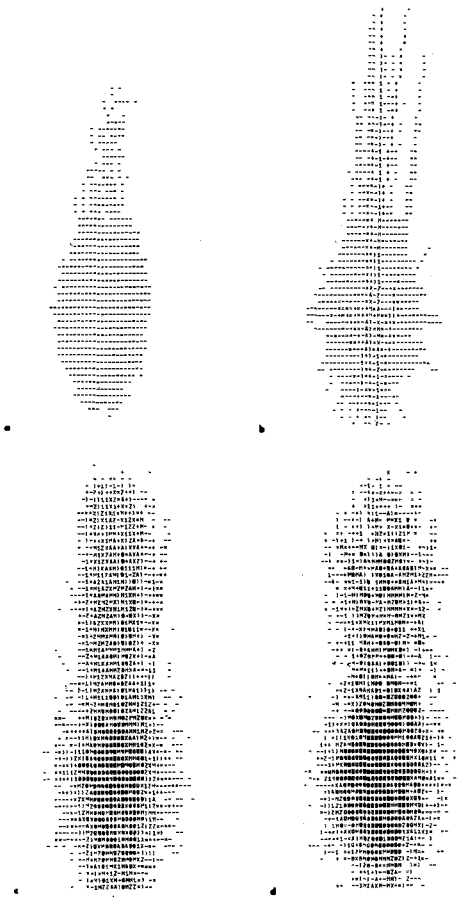

Fig. 7. Reconstruction of the heat distribution of equation (6) using ART3 from the same data as in Fig. 6. Four stages of the reconstruction process are shown: at $k = 0, M, 2M$ and $3M$ (final result). The actual mean density is the same at all four stages but, due to its rather low value and the fact that below a certain positive threshold the computer printer simulated half-tone is blank, the earlier stages appear to have a lower mean density.

heated by two sources. The underlying equation is of the form

$$f(x, y) = 0.9 \exp\{-0.5[x^2 + (y - 0.5)^2]\} + 0.6 \exp\{-0.5[x^2 + (y + 0.5)^2]\} \quad (6)$$

As opposed to the heart reconstructions, where the X-rays were diverging from a point source and were assumed to be infinitely thin, our simulated data collection was done by parallel rays of thickness approx-

imately the size of the sides of the pixels. Such data collection (strip integration) is appropriate for the applications in electronmicroscopy, radio-astronomy, etc.

The data collection was done by numerically integrating the given distribution within the strips. (For the purpose of numerical integration a very fine digitization was used, in which each pixel was further subdivided into $11^2 = 121$ smaller pixels.) After this the data was made noisy, by multiplying by a random number from a normal distribution with mean 1 and standard deviation s, where s was either 0.03 or 0.1.

In all our reconstructions we have chosen

$$\epsilon_i = (s + 0.01)b_i, \tag{7}$$

i.e., the tolerance was slightly larger than one standard deviation of the noise. The algorithm used was precisely the one given in Fig. 4. We shall from now on refer to this algorithm as ART3, since it is clearly a variant of the Algebraic Reconstruction Techniques proposed in [9]. The working of ART3 is demonstrated in Fig. 7. This is a reconstruction of Sweeney's pattern (Fig. 6a) from 16 sets of parallel rays (i.e., 16 projections) equally spaced between 0° and 180°. Since there are 64^2 pixels and a projection contains 64 to 127 rays, there are less than half as many equations than unknowns. In addition, the data was noisy with standard deviation 10% (i.e., $s = 0.1$). In spite of this, a reasonable reconstruction was achieved, as can be seen from Fig. 7, which shows four stages of the reconstruction process, at $k = 0, M, 2M$ and $3M$ (the final result).

In comparison, Fig. 6b, c, d shows the reconstructions from the same data using the ART2 method of [14], the SIRT method of [7] and the convolution method of [20], respectively. (All three methods are described as we implemented them in [15].) In considering the rather bad quality of the reconstruction in the convolution method, one should bear in mind that the result may be improved by filtering the noisy data, a device which we have not used. Ten iterations of the SIRT algorithm were used (see [7]).

Clearly, ART3 gives in this particular situation a better reconstruction than either ART2 or the convolution method, but not as good a reconstruction as SIRT. It is in the nature of the SIRT algorithm to handle white noise very effectively (see [15]). However, the SIRT algorithm

Table 1
Normalized root mean square distance of reconstructions from original patterns

Pattern	# views	Range	Noise	ART3	ART2	SIRT	CONV
Fig. 6a	16	full	10	0.34*	0.52*	0.24*	1.32*
		$-22.5°, 22.5°$	10	0.46	0.77	0.42	3.09
Fig. 8a	10	full	3	0.42	0.43	0.43	0.64
		full	10	0.61	0.70	0.54	1.19
	25	full	3	0.29*	0.44	0.37	0.26
		full	10	0.60	0.91	0.45	0.66
Fig. 8b	6	full	10	0.22*	0.30	0.27	0.86
	12	full	10	0.24	0.41	0.24	0.67
Fig. 8c	8	$-35°, +35°$	3	0.31	0.34	0.45	0.47
			10	0.33*	0.33	0.46	0.59
Best				4	0.5	4.5	1
Worst				0	2	0	8
Difference				4	−1.5	4.5	−7

costs over twice as much to run. For this reconstruction, the ART3 algorithm was not only better, but also somewhat cheaper than the ART2 algorithm. The convolution method is the cheapest of the four (less than half the cost of ART3), but in this case the convolution method gave such a bad reconstruction that the saving in cost is immaterial.

Since the relative timings described above are typical for all experiments, we shall report only on the accuracy of reconstruction in the 10 experiments we have carried out. In Table 1 we tabulated the values of

$$\delta = \left| \frac{\sum_{i=1}^{N^2} (x_i - \hat{x}_i)^2}{\sum_{i=1}^{N^2} (x_i - \bar{x})^2} \right|^{1/2} \tag{8}$$

where x_i and $\hat{x}_i$ are the average densities in the i^{th} pixel of the test pattern and reconstruction, respectively, and

$$\bar{x} = \frac{1}{N^2} \sum_{i=1}^{N^2} x_i$$

is the average density of the pattern. δ is the normalized root mean

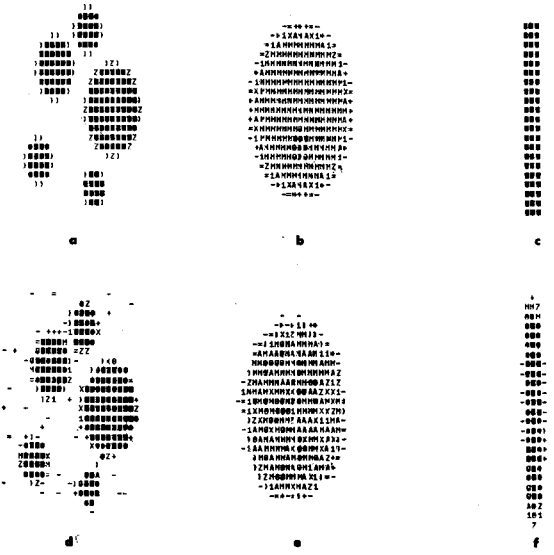

Fig. 8. Some reconstructions using ART3. In all patterns 25 × 25 pixels are used.
(a) A test pattern proposed in [7].
(b) A test pattern proposed in [8].
(c) A test pattern proposed in [6].
(d) Reconstruction of (a) from 25 projections in a full range with parallel rays and 3% noise.
(e) Reconstruction of (b) from 6 projections in a full range with parallel rays and 10% noise.
(f) Reconstruction of (c) from 8 projections in a 70% range with parallel rays and 10% noise.

square distance of the reconstruction to the original. The appropriateness of such a measure has been justified in [5]. In all our experiments the angle of the views were equally spaced in the range specified. Full range means that the views are equally spaced between 0° and 180°. The noise is the value of s as explained above (i.e., 3% noise means s = 0.03).

At the bottom of the table it is indicated in how many of ten experiments a particular method was the best or the worst, and the difference between these two numbers is also given. Although the exact values of these figures are not to be taken too seriously, the general conclusion that SIRT and ART3 are the best, convolution is the worst and ART2 is somewhere in between for this kind of noisy data seems to be well established. One must however bear in mind the possible improvement to the convolution method by filtering, especially in view of its inexpensiveness and its excellent performance on noiseless data (see [15]).

For example, suing the Fourier space cut off proposed by Bracewell and Riddle [2], which is lower than the one discussed in [20] and [15], we obtain the values $\delta = 0.60$ and 0.44 for reconstructions of Fig. 6a from 10 and 25 views and 10% noise. ART3 can be considered to be preferable to SIRT because of the reduced cost.

Finally, on Fig. 8d, e, f we show some reconstructions using ART3 on the patterns in Fig. 8a, b, c. The values of δ for the reconstructions which are shown in this figure are marked by an asterisk in Table 1.

5. Conclusions

The method proposed in this paper can be used to produce inexpensive and accurate reconstructions from noisy X-ray projections. It has been implemented and found useful in the reconstruction of a beating heart. It can similarly be used in other application areas of the reconstruction problem.

Acknowledgment

The author is grateful to the following individuals for their cooperation and for many discussions: P.W. Aitchison, J.F. Greenleaf, R. Gordon, J.A. Hinds, S.A. Johnson, S.L. Lee, A. Lent, E.L. Ritman, R.A. Robb, S.W. Rowland, K.T. Smith, and E.H. Wood.

References

[1] R.N. Bracewell, "Strip integration in radio astronomy", *Australian Journal of Physics* 9 (1956) 198–217.

[2] R.N. Bracewell and A.C. Riddle, "Inversion of fanbeam scans in radio astronomy", *Astrophysical Journal* 150 (1967) 427–434.

[3] A.M. Cormack, "Representation of a function by its line integrals, with some radiological applications", *Journal of Applied Physics* 34 (1963) 2722–2727.

[4] D.J. DeRosier and A. Klug, "Reconstruction of three-dimensional structures from electron micrographs", *Nature* 217 (1968) 130–134.

[5] G. Frieder and G.T. Herman, "Resolution in reconstructing objects from electron micrographs", *Journal of Theoretical Biology* 33 (1971) 189–211.

[6] N.T. Gaarder and G.T. Herman, "Algorithms for reproducing objects from their X-rays", *Computer Graphics and Image Processing* 1 (1972) 97–106.

[7] P.F.C. Gilbert, "Iterative methods for the reconstruction of three-dimensional objects from projections", *Journal of Theoretical Biology* 36 (1972) 105–117.

[8] J.L. Goffin, "On the finite convergence of the relaxation method for solving systems of inequalities", Ph.D. thesis, University of California, Berkeley (1971).

[9] R. Gordon, R. Bender and G.T. Herman, "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography", *Journal of Theoretical Biology* 29 (1970) 471–481.

[10] R. Gordon and G.T. Herman, "Three dimensional reconstruction from projections: a review of algorithms", in: G.F. Bourne and J.F. Danielli, eds., *International review of cytology* (Academic Press, New York, 1974) pp. 111–151.

[11] J.F. Greenleaf, S.A. Johnson, S.L. Lee, G.T. Herman and E.H. Wood, "Algebraic reconstruction of spatial distributions of acoustic absorption within tissue from their two-dimensional acoustic projections", in: P.S. Green, ed., *Acoustical holography and imaging* (Plenum Press, New York, 1973) pp. 591–603.

[12] G.T. Herman, "Two direct methods for reconstructing pictures from their projections: a comparative study", *Computer Graphics and Image Processing* 1 (1972) 123–144.

[13] G.T. Herman, "Reconstruction of binary patterns from a few projections", in: A. Gunther, B. Levrat and H. Lipps, eds., *International computing symposium 1973* (North-Holland, Amsterdam, 1973) pp. 371–380.

[14] G.T. Herman, A. Lent and S. Rowland, "ART: Mathematics and applications (a report on the mathematical foundations and on the applicability to real data of the Algebraic Reconstruction Techniques)", *Journal of Theoretical Biology* 42 (1973) 1–32.

[15] G.T. Herman and S. Rowland, "Three methods for reconstructing objects from x-rays: a comparative study", *Computer Graphics and Image Processing* 2 (1973) 151–178.

[16] S.A. Johnson, R.A. Robb, J.F. Greenleaf, E.L. Ritman, G.T. Herman, S.L. Lee and E.H. Wood, "Temporal and spatial x-ray projective reconstruction of a beating heart", *Mayo Clinic Proceedings,* to appear.

[17] S.A. Johnson, R.A. Robb, J.F. Greenleaf, E.L. Ritman, S.L. Lee, G.T. Herman, R.E. Sturm and E.H. Wood, "The problem of accurate measurement of left ventricular shape and dimensions from multiplane Roentgenographic data", *European Journal of Cardiology* 1 (1973) 137–154.

[18] T.S. Motzkin and I.J. Schoenberg, "The relaxation method for linear inequalities", *Canadian Journal of Mathematics* 6 (1954) 393–404.

[19] J. Radon, "Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten", *Berichte Sächsische Akademie der Wissenschaften (Leipzig) Mathematische-Physische Klasse* 69 (1917) 262–277.

[20] G.N. Ramachandran and A.V. Lakshminarayanan, "Three-dimensional reconstruction from radiographs and electron micrographs: application of convolutions instead of Fourier transforms", *Proceedings of the National Academy of Sciences USA* 68 (1971) 2236–2240.

[21] R.A. Robb, J.F. Greenleaf, E.L. Ritman, S.A. Johnson, J. Sjostrand, G.T. Herman and E.H. Wood, "Three-dimensional visualization of the intact thorax and contents: A technique for cross-section reconstruction from multiplanar x-ray views", *Computers and Biomedical Research* 7 (1974) 395–419.

[22] D. Sweeney, "Interferometric measurement of three-dimensional temperature fields", Ph. D. Thesis, University of Michigan, Ann Arbor, Mich. (1972).