Notes on block-iterative or ordered subset methods for image reconstruction

Charles Byrne (Charles_Byrne@uml.edu),
Department of Mathematical Sciences,
University of Massachusetts Lowell, Lowell, MA 01854

September 20, 2000

Abstract

There has been growing interest in block-iterative (also called ordered subset) methods for image reconstruction, due largely to the accelerated convergence some of these methods provide. The block-iterative methods of interest to us here can be derived as incremental optimization procedures, in which the cost function $h(x)$ to be minimized can be decomposed as a sum of simpler functions, $h(x) = \sum_{i=1}^{I} h_i(x)$, and the iterative procedure involves the gradients of only a few of the $h_i(x)$ at each step. We present a convergence theorem for such algorithms and consider a number of special cases, such as the algebraic reconstruction technique (ART), its multiplicative version (MART) and rescaled block-iterative variants of MART and of the expectation maximization maximum likelihood (EMML) method. We include an examination of the sparse matrix case, in which additional acceleration is possible through the use of relaxation parameters that depend on the degree of sparseness. Optimal relaxation to avoid limit cycles is also considered.

1 Introduction

Our topic is the reconstruction of a discrete image from finite data pertaining to that image. Because realistic models relating the data to the image pixels (or voxels) typically preclude closed form solutions, we shall focus here on iterative algorithms. For reasons to be presented shortly, the algorithms we shall consider are optimization methods, in which we seek to maximize or minimize some function over the set of feasible images, that is, those satisfying whatever constraints, such as nonnegativity, we have imposed .

When the data is essentially noise-free, but insufficient to determine a unique image, one may choose that feasible image consistent with the data, for which some function, such as entropy, is maximized, or some measure of image roughness or distance to a prior estimate of the image is minimized. When the data is noisy, there may be no feasible image consistent with the data. In such cases, one may choose to minimize a function that measures deviation from data consistency, with or without an additional regularizing term.

In typical image reconstruction situations both the data set and the number of pixels or voxels to be determined are large; in addition, time considerations are important. The overall objective is the practical one of producing a useful reconstructed image quickly, rather than the more theoretical one of finding the solution of an optimization problem. Therefore iterative methods that produce fairly accurate reconstructed images in a short time are desired. For such practical reasons there has been growing interest in certain *block-iterative* or *ordered subset* methods [22], [6], [25], which provide the main topic of this paper.

Block-iterative methods are called *incremental* methods in the optimization literature [1]. The basic idea is as follows. Suppose that we wish to minimize a function $h : R^J \to (-\infty, +\infty)$. Iterative gradient methods would require us to calculate the gradient of h at each step. If h is the sum of a large number of simpler functions h_i whose gradients are easier to calculate, so

$$h(x) = \sum_{i=1}^{I} h_i(x), \tag{1.1}$$

then at the k-th step we would need to compute

$$\nabla h(x^k) = \sum_{i=1}^{I} \nabla h_i(x^k). \tag{1.2}$$

For example, consider the least squares problem of finding a minimizer of the function $h(x) = \frac{1}{2}||Ax - b||^2$, where A is any real I by J matrix. The gradient of h is $\nabla h(x) = A^T(Ax-b)$. We can put h into the form of equation (1.1) using $h_i(x) = \frac{1}{2}((Ax)_i - b_i)^2$, which has for its gradient $\nabla h_i(x) = ((Ax)_i - b_i)a^i$, where a^i is the i-th column of the matrix A^T.

To avoid computing the large sum in (1.2), we might consider using only those gradients ∇h_i whose indices i belong to some predetermined *block* B_n, where n depends on k; we assume throughout this paper that $\{B_1, ..., B_N\}$ denotes a partition of the set $\{i = 1, ..., I\}$ into disjoint subsets. We then proceed *incrementally*, using only these partial gradients to determine the direction to the next iterate. Stated this way, block-iterative methods appear to reduce computation at each step; but if the price we pay is to increase the number of steps needed to produce a good reconstructed image, we have gained nothing. Several of the block-iterative methods we shall discuss here do not require an increased number of steps, hence provide considerable time reduction in the reconstruction process.

When there is only a single block, that is $N = 1$, we say that the method is *simultaneous*. When each block contains only a single i, so there are I blocks, we call the method *sequential* or *successive*; for problems involving the solution of matrix equations sequential methods have also been called *row-action* methods [14].

Because most of the functions we encounter in image reconstruction can be decomposed as in equation (1.1), obtaining block-iterative versions of iterative optimization algorithms is usually not difficult; but this is not enough. In order for a block-iterative method to be useful it must satisfy certain requirements, which we shall discuss in this paper. These requirements pertain to acceleration of convergence, as well as to the manner in which the method handles noise in the data.

Block-iterative methods are not new and the literature on the subject is extensive; see the book by Censor and Zenios [18] and the references therein. We do not intend to present here a survey of this field. Our goal is to tell a coherent story about block-iterative methods as they relate to acceleration in image reconstruction. To illustrate a number of points that will concern us, we begin with some well known examples, the *algebraic reconstruction technique* (ART) and its multiplicative version, MART [21]. We then turn to a fairly general procedure, the *interior point algorithm* (IPA), which includes, as special cases, most of the particular algorithms we study here. The proof of convergence of the general IPA is given in [12] and we do not repeat it here. Instead, we present somewhat less technical convergence proofs of certain special cases, in an attempt to isolate the precise mechanism for acceleration of convergence.

2 The ART and related methods

In this section and the next we consider two well known sequential methods, the algebraic reconstruction technique (ART) and its multiplicative version, MART. Both algorithms were introduced by Gordon, Bender and Herman in [21]; the ART is a special case of Kaczmarz's algorithm [26]. Both ART and MART have block-iterative and simultaneous counterparts, which we shall also discuss.

In the case of ART the function we wish to minimize is $h(x) = \frac{1}{2}||Ax - b||^2$, where A is a general I by J real matrix. Notice that the function $h(x)$ can be written in the form given in equation (1.1) using $h_i(x) = \frac{1}{2}((Ax)_i - b_i)^2$.

Each equation in the system $Ax = b$ corresponds to a hyperplane in the space

R^J. The idea of ART is to project orthogonally onto each of these hyperplanes in succession, according to some predetermined order. In closed form, the ART is the following:

The ART: For $k = 0, 1, ...$ and $i = k(\bmod I) + 1$ set

$$x_j^{k+1} = x_j^k + A_{ij} \frac{(b_i - (Ax^k)_i)}{\sum_{l=1}^J A_{il}^2}. \tag{2.1}$$

The *relaxed ART* (REART) has the following iterative step:

The REART:

$$x_j^{k+1} = x_j^k + \gamma_k A_{ij} \frac{(b_i - (Ax^k)_i)}{\sum_{l=1}^J A_{il}^2}, \tag{2.2}$$

with γ_k positive scalars.

A simultaneous version of the ART (SART) is obtained by projecting orthogonally onto each hyperplane simultaneously, then averaging the result. In closed form the SART is the following:

The SART: For $k = 0, 1, ...$ let

$$x_j^{k+1} = x_j^k + \frac{1}{I} \sum_{i=1}^I A_{ij} \frac{(b_i - (Ax^k)_i)}{\sum_{l=1}^J A_{il}^2}; \tag{2.3}$$

with

$$G_{ij} = A_{ij} / (\sum_{l=1}^J A_{il}^2)^{1/2} \tag{2.4}$$

and

$$c_i = b_i / (\sum_{l=1}^J A_{il}^2)^{1/2}, \tag{2.5}$$

the iteration in equation (2.3) becomes

$$x^{k+1} = x^k + \frac{1}{I} G^T (c - Gx^k). \tag{2.6}$$

Relaxed SART (RESART) employs positive relaxation parameters γ_k in place of $\frac{1}{I}$:

The RESART:

$$x^{k+1} = x^k + \gamma_k G^T (c - Gx^k). \tag{2.7}$$

In a later section we give a convergence proof for the RESART that requires the relaxation parameters to satisfy the inequality $0 < \gamma_k < 2/L$, where L is the largest eigenvalue of the matrix $G^T G$. Since the trace of GG^T is I, we know that $L \leq I$. This is a quite conservative estimate, in most cases, particularly if the matrix A is sparse. Let s_j be the number of nonzero entries in the j-th column of A and let s be the maximum of the s_j. As we shall see, $L \leq s$, which says that the relaxed SART converges with $\gamma_k = \frac{1}{s}$. To illustrate, suppose that $s = I^{1/2}$. Then the factor I^{-1} in SART can be replaced with $I^{-1/2}$, which significantly accelerates convergence. We can obtain additional acceleration by passing to a block-iterative version of ART.

Because the computations in SART can be performed simultaneously, the SART has the advantage of being parallelizable. In practice, it might be more efficient for only a subset of these computations to be performed simultaneously. In that case, block-iterative versions of SART would be more appropriate. We consider those now.

We can obtain a block-iterative version of ART (equivalently, of SART) (BI-ART) by projecting orthogonally onto each hyperplane corresponding to an index in the current block, then averaging the result. For $n = 1, ..., N$ let I_n be the cardinality of the block B_n.

The *block-iterative* ART (BI-ART) has the following iterative step:

The BI-ART: For $k = 0, 1, ...$ and $n = n(k) = k(\mathrm{mod} N) + 1$ let

$$x_j^{k+1} = x_j^k + \frac{1}{I_n} \sum_{i \in B_n} A_{ij} \frac{(b_i - (Ax^k)_i)}{\sum_{l=1}^J A_{il}^2}. \tag{2.8}$$

Obtain the matrix G_n from G in equation (2.4) by removing the i-th row of G for those i not in B_n. Similarly, obtain vector c^n from c in equation (2.5). Then the iteration in equation (2.8) becomes

$$x^{k+1} = x^k + \frac{1}{I_n} G_n^T (c^n - G_n x^k). \tag{2.9}$$

Relaxed BI-ART (RE-BI-ART) employs positive relaxation parameters γ_k in place of $\frac{1}{I_n}$:

The RE- BI-ART: For $k = 0, 1, ...$ and $n = n(k) = k(\mathrm{mod} N) + 1$ let

$$x^{k+1} = x^k + \gamma_k G_n^T (c^n - G_n x^k). \tag{2.10}$$

When there are solutions of $Ax = b$ the RE-BI-ART converges to the solution closest to the starting point x^0, whenever $\sum_{k=0}^{+\infty} \gamma_k = +\infty$ and $0 < \gamma_k \leq \frac{1}{L_n}$, where L_n is the largest eigenvalue of the matrix $G_n^T G_n$. The same trace argument as used earlier tells us that $L_n \leq I_n$. If G_n is sparse, we can do much better. For fixed n and j, let s_{nj} be the number of nonzero entries in the j-th column of the matrix G_n; let s_n be the maximum of the s_{nj}. Then we have $L_n \leq s_n$, so that the factor $1/I_n$ in equation

(2.8) can be replaced by the factor $1/s_n$. This can lead to significant acceleration of convergence.

Suppose, for the sake of illustration, that each column of the matrix A has s nonzero entries and that $r = s/I$ is the proportion of nonzero entries in any column. Suppose that $I_n = I/N$ for each n. If N is not too large, we would expect s_{nj} to be nearly equal to $rI_n = rI/N$, for each j and n; then s_n is nearly $rI/N = \frac{s}{I}\frac{I}{N} = \frac{s}{N}$. So the factor $\frac{1}{I}$ in SART is replaced by $\frac{s}{N}$ in RE-BI-ART. But, unless $s_n = 0$, which means the matrix G_n is the zero matrix, we have $s_n \geq 1$, regardless of the size of N. So the factor $1/s_n$ is never larger than one, which is the factor used in unrelaxed ART. For a given value of s, we need to use approximately $N = s$ blocks to have s_n nearly equal to one. Therefore, the more sparse the matrix is, the fewer blocks we need to use for the factor $1/s_n$ to attain its maximum value. For very sparse matrices, few blocks are needed, allowing for a high degree of parallelization, since, within each block, the computation is simultaneous.

To deal with the issue of acceleration with some precision, we need to discuss some of the details of the proof of convergence of the RE-BI-ART; the complete proof we leave to a later section.

The RE-BI-ART is an example of an *iterative sequential unconstrained minimization technique*, an approach to minimization introduced by Fiacco and McCormick [20]. The idea is to replace an optimization problem that has no closed form solution with a sequence of simpler optimization problems, which can be solved in closed form and whose solutions converge to the solution of the original problem. For each k and $n = k(\bmod N) + 1$, let $H_k(x, z)$ be defined by

$$H_k(x, z) = ||x - z||^2 - \gamma_k||G_n x - G_n z||^2 + \gamma_k||G_n x - c^n||^2. \tag{2.11}$$

It is clear that x^{k+1} defined by equation (2.10) is the unique minimizer of $H_k(x, x^k)$. Suppose that $\hat{x}$ solves $Gx = c$. Then we can show that

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 = H_k(x^{k+1}, x^k) + \gamma_k||c^n - G_n x^k||^2. \tag{2.12}$$

In order to have the sequence $\{||\hat{x} - x^k||^2\}$ decreasing we would like to know that $H_k(x^{k+1}, x^k) \geq 0$ always. One way to guarantee this is to select γ_k so that $||x - z||^2 - \gamma_k||G_n x - G_n z||^2 \geq 0$ for all x and z; this can be done be taking $\gamma_k \leq 1/L_n$, where L_n is the largest eigenvalue of the matrix $G_n^T G_n$. On the other hand, the improvement we make from one step to the next, indicated by the difference $||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2$ in (2.12), is seen to be roughly proportional to γ_k, so a larger value of γ_k is desirable. The trick is to pass to a block-iterative scenario in which the G_n are selected so as to make the L_n as small as possible, which tends to happen when we use as many blocks as possible, and then to select $\gamma_k = 1/L_n$. Note also that the right side of (2.12) contains the term $||c^n - G_n x^k||^2$. If this value is large, the left side will also be large. If the blocks are ordered poorly, so that $||c^n - G_n x^k||^2$ is small, the convergence to a solution may be slower. This point concerning the importance of the ordering of the blocks was stressed with respect to ART by Herman and Meyer [22].

When there are solutions of the system $Ax = b$ then ART, BI-ART and SART converge to the solution of $Ax = b$ closest to the initial vector x^0, according to the Euclidean distance. When there are no solutions of $Ax = b$ SART converges to a geometric least squares solution, the minimizer of $||Gx - c||$ closest to x^0, while ART and BI-ART fail to converge. Instead, as Tanabe has shown [30], for each fixed i, as $m \to +\infty$, the ART subsequences $\{x^{mI+i}\}$ converge to distinct vectors $x^{\infty,i}$; we call this set of vectors the *limit cycle* (LC). The greater the minimum value of $||Ax - b||^2$ the more the vectors of the LC are distinct from one another. An analogous result holds for RE-BI-ART.

There are several ways to avoid the LC in ART and to obtain a least squares solution. One way, which does not easily generalize to other algorithms, is what we call *double ART* (DART):

The DART: We know that any b can be written as $b = A\hat{x} + \hat{w}$, where $A^T\hat{w} = 0$ and $\hat{x}$ is a minimizer of $||Ax - b||^2$. The vector $\hat{w}$ is the orthogonal projection of b onto the null space of the matrix transformation A^T. Therefore, in Step 1 of DART we apply the ART algorithm to the consistent system of linear equations $A^T w = 0$, beginning with $w^0 = b$. The limit is $w^\infty = \hat{w}$, the member of the null space of A^T closest to b. In Step 2, apply ART to the consistent system of linear equations $Ax = b - w^\infty = A\hat{x}$. The limit is then the minimizer of $||Ax - b||$ closest to x^0.

Another method for avoiding the LC is *strong underrelaxation* [15].

Strongly underrelaxed ART: Let $t > 0$. Replace the iterative step in ART with

$$x_j^{k+1} = x_j^k + tA_{ij}\frac{(b_i - (Ax^k)_i)}{\sum_{l=1}^J A_{il}^2}. \tag{2.13}$$

In [15] it is shown that, as $t \to 0$, the vectors of the LC approach the geometric least squares solution closest to x^0. Bertsekas [1] uses strong underrelaxation to obtain convergence of more general incremental methods.

A third way to avoid the LC in ART is to use a special case of the relaxed ART. We call this method the *least squares* ART (LS-ART).

The LS-ART: For $k = 0, 1, ...,$ and $i = k(\bmod I) + 1$ let

$$x_j^{k+1} = x_j^k + \alpha_k A_{ij}, \tag{2.14}$$

with

$$\alpha_k = (Aa^i)^T(b - Ax^k)/||Aa^i||^2, \tag{2.15}$$

and a^i the i-th column of the matrix A^T. The choice of $\alpha = \alpha_k$ minimizes the function $||A(x^k + \alpha a^i) - b||$ over all α. Since $\alpha = 0$ is one choice, we know that $||Ax^k - b|| \geq ||Ax^{k+1} - b||$. The LS-ART is a special case of the *optimally relaxed* method we discuss in the final section of this paper; there we show that the sequence $\{x^k\}$ converges to a minimizer of $||Ax - b||$. Note that it is not necessary to calculate the vector Ax^k; the LS-ART has the same computational cost as ART, if we can precompute and store the vectors $A^T Aa^i$.

In practical situations, one may use only a few iterations of an algorithm and be less concerned with the limiting vector (or vectors) than with the behavior of the iterates for small values of k. When the minimum value of $||Ax - b||^2$ is not too large (that is, the measured data is not too noisy), the ART has been shown to provide usable reconstructions with very few iterations, particularly when the equations are carefully ordered and some amount of underrelaxation is used [22]. In contrast, the SART is typically quite slow to converge.

It is important to note that acceleration of convergence need not require passing from a simultaneous method to a block-iterative method. The example of SART and BI-ART in the case of a sparse matrix A shows that part of the reason why SART is slow is that it does not employ an appropriate relaxation parameter. If we know a good upper bound on the eigenvalues of G^TG then we can improve SART by using RESART with better values of γ_k. If we have no *a priori* estimate, we could begin with $\gamma_k = 1/I$ and begin to lower the γ_k as the iteration proceeds, checking for divergence. In the sparse case, as we have seen, we can get significant acceleration with RESART by making use of the degree of sparseness of the matrix G.

3 The MART and related methods

As a second example, we consider the *multiplicative* ART (MART), also due to Gordon, Bender and Herman [21].

We consider now a system of linear equations $y = Px$, in which the I by J matrix P has nonnegative entries, with columns that sum to one, the entries of y are positive and x has nonnegative entries. The function to be minimized is now $h(x) = KL(Px, y)$; here $KL(u, v)$ is the Kullback-Leibler (or cross-entropy) distance, defined for nonnegative vectors u and v by

$$KL(u,v)=\sum_{m=1}^{M}KL(u_m,v_m), \tag{3.1}$$

where $KL(a,b)=a\log\frac{a}{b}+b-a$, $KL(0,b)=b$ and $KL(a,0)=+\infty$ for positive scalars a and b. With $h_i(x)=KL((Px)_i,y_i)$ we see that h has the decomposition given by equation (1.1).

The MART algorithm is the following:

The MART: The *multiplicative algebraic reconstruction technique* (MART) [21] begins with a strictly positive vector x^0 and has the iterative step

$$x_j^{k+1}=x_j^k\Big(\frac{y_i}{(Px^k)_i}\Big)^{P_{ij}}, \tag{3.2}$$

for $j=1,2,...,J$ and $i=k(\bmod I)+1$. The simultaneous MART (SMART) algorithm is then

The SMART: The *simultaneous* MART (SMART) (see [3], [17], [24]) begins with a strictly positive vector x^0 and has the iterative step

$$x_j^{k+1}=x_j^k\prod_{i=1}^{I}\Big(\frac{y_i}{(Px^k)_i}\Big)^{P_{ij}}, \tag{3.3}$$

for $j=1,2,...,J$. The block-iterative SMART (BI-SMART) is as follows:

The BI-SMART: The *block-iterative* SMART (BI-SMART) [5] begins with a strictly positive vector x^0 and has the iterative step

$$x_j^{k+1}=x_j^k\prod_{i\in B_n}\Big(\frac{y_i}{(Px^k)_i}\Big)^{P_{ij}}, \tag{3.4}$$

for $j=1,2,...,J$ and $n=k(\bmod N)+1$. Clearly, MART and SMART are special cases of the BI-SMART method. We introduce relaxation into the BI-SMART as follows:

The relaxed BI-SMART: The relaxed BI-SMART begins with a strictly positive vector x^0 and has the iterative step

$$x_j^{k+1}=x_j^k\prod_{i\in B_n}\Big(\frac{y_i}{(Px^k)_i}\Big)^{\gamma_k P_{ij}}, \tag{3.5}$$

for $j=1,2,...,J$ and $n=k(\bmod N)+1$.

In the consistent case, that is, when there are vectors $x\geq 0$ with $y=Px$, BI-SMART converges to the nonnegative solution that minimizes $KL(x,x^0)$. When there are no such nonnegative vectors, the SMART converges to the unique nonnegative minimizer of $KL(Px, y)$ for which $KL(x, x^0)$ is minimized (see [3]); for $N > 1$, the BI-SMART fails to converge. What is always observed, but for which no proof exists, is that, for each fixed $i = 1, 2, ..., I$, as $m \to +\infty$, the MART subsequences $\{x^{mI+i}\}$ converge to separate limit vectors, say $x^{\infty,i}$. This *limit cycle* LC $= \{x^{\infty,i} | i = 1, ..., I\}$ reduces to a single vector whenever there is a nonnegative solution of $y = Px$. The greater the minimum value of $KL(Px, y)$ the more distinct from one another the vectors of the limit cycle are. An analogous result is observed for BI-SMART.

The MART will converge, in the consistent case, provided that $0 \leq P_{ij} \leq 1$, for all i and j; this condition holds here since we have assumed that the columns of P sum to one. Since I is typically quite large, the P_{ij} are likely to be a great deal smaller than one. We can accelerate the convergence of MART by rescaling the equations, obtaining what we have called the REMART.

The REMART: The *rescaled multiplicative algebraic reconstruction technique* (REMART) [5] begins with a strictly positive vector x^0 and has the iterative step

$$x_j^{k+1} = x_j^k \Big(\frac{y_i}{(Px^k)_i} \Big)^{m_i^{-1} P_{ij}}, \tag{3.6}$$

for $j = 1, 2, ..., J$ and $i = k (\mathrm{mod}\, I) + 1$, with $m_i = \max\{P_{ij} | j = 1, ..., J\}$.

Although the importance of the rescaling for accelerating MART was not remarked upon in earlier papers on MART, the rescaling was often a part of actual implementations [23].

Similarly, the BI-SMART will converge, in the consistent case, provided that $0 \leq \sum_{i \in B_n} P_{ij} \leq 1$, for all n and j; this condition holds here since we have assumed that the columns of P sum to one. Since N may be large, the $\sum_{i \in B_n} P_{ij}$ are likely to be a great deal smaller than one. We can accelerate the convergence of BI-SMART by rescaling the equations, obtaining what we have called the *rescaled block-iterative* SMART (RBI-SMART).

The RBI-SMART: The *rescaled block-iterative* SMART (RBI-SMART) [5] begins with a strictly positive vector x^0 and has the iterative step

$$x_j^{k+1} = x_j^k \prod_{i \in B_n} \Big(\frac{y_i}{(Px^k)_i} \Big)^{m_n^{-1} P_{ij}}, \tag{3.7}$$

for $j = 1, 2, ..., J$ and $n = k (\mathrm{mod}\, N) + 1$, with $m_n = \max\{\sum_{i \in B_n} P_{ij} | j = 1, ..., J\}$.

The BI-SMART and RBI-SMART converge whenever there is a common nonnegative minimizer of the functions $h_i(x), i = 1, .., I$. When there is no such vector, we get a limit cycle LC.

As we did with the RE-BI-ART we consider details of the proof of convergence of relaxed BI-SMART to focus on the mechanisms available for accelerating convergence.

The BI-SMART is a particular case of the *interior point algorithm* (IPA), to be discussed later; the full proof of convergence of the IPA is in [12].

For $k = 0, 1, ...,$ and $n = k(\mathrm{mod}\, N) + 1$ we can see easily that x^{k+1} in (3.5) is the unique minimizer of the function $G_k(x, x^k)$ given by

$$G_k(x, x^k) = KL(x, x^k) - \gamma_k \sum\nolimits^n KL(Px_i, Px_i^k) + \gamma_k \sum\nolimits^n KL(Px_i, y_i), \qquad (3.8)$$

where $\sum^n$ denotes the sum over those indices i in the block B_n and $Px_i^k = (Px^k)_i$. Let $\hat{x}$ be an arbitrary nonnegative solution of $y = Px$. Then we can show that

$$KL(\hat{x}, x^k) - KL(\hat{x}, x^{k+1}) = G_k(x^{k+1}, x^k) + \gamma_k \sum\nolimits^n KL(y_i, Px_i^k). \qquad (3.9)$$

We want to conclude that the sequence $\{KL(\hat{x}, x^k)\}$ is decreasing. To be sure that $G_k(x^{k+1}, x^k) \geq 0$ we select γ_k so that $1/\gamma_k \geq \sum^n P_{ij}$ for all j. We reason here as follows. It is easily shown that $KL(x, z) \geq KL(x_+, z_+)$, for any nonnegative vectors x and z, with x_+ and z_+ denoting the sums of the entries of vectors x and z, respectively. Then $KL(x, z) \geq \gamma_k \sum_j (\sum^n P_{ij}) KL(x_j, z_j) \geq \gamma_k \sum^n KL(Px_i, Pz_i)$. At the same time, we see that the improvement in distance to a solution, as described by the left side of equation (3.9), is roughly proportional to γ_k, so we want γ_k as large as possible. This suggests taking $\gamma_k = m_n^{-1}$, where $m_n = \max\{\sum^n P_{ij} | \, j = 1, ..., J\}$; this is the choice used in the RBI-SMART. We note finally that the right side of equation (3.9) also contains the term $\sum^n KL(y_i, Px_i^k)$, which we want to be large also. As in the case of relaxed BI-ART, the ordering of the blocks affects the rate of convergence.

In all of the examples we have just considered we have convergence to a solution in the consistent case, but expect limit cycles for the block-iterative methods in the inconsistent case.

In the next section we consider another example, the *expectation maximization maximum likelihood* (EMML) method. We show how one particular attempt to form a block-iterative version of EMML, the *ordered subset* EM (OSEM), usually fails to converge in the consistent case and we show how to obtain a corrected algorithm.

4 Another example: the EMML

In emission tomography, for the model of independent Poisson emitters distributed over the region of interest, the maximum likelihood problem is equivalent to minimizing the cross-entropy, or Kullback-Leibler distance,

$$KL(y, Px) = \sum_{i=1}^{I} y_i \log \frac{y_i}{(Px)_i} + (Px)_i - y_i, \qquad (4.1)$$

where P is an I by J matrix of nonnegative entries with column sums equal to one, y is the vector of (positive) counts at each detector and x is vector with nonnegative entries. Say we are in the consistent case if there is a nonnegative x with $y = Px$; otherwise, we are in the inconsistent case. The EMML algorithm has the following iterative step:

The EMML:

$$x_j^{k+1} = x_j^k \sum_{i=1}^{I} P_{ij}\Big(\frac{y_i}{(Px^k)_i}\Big). \tag{4.2}$$

In the consistent case the EMML converges to a nonnegative solution of $y = Px$; in the inconsistent case it converges to the (almost always) unique minimizer of $KL(y, Px)$ [3], [4], [5]. If we had not redefined P and x so as to have the columns of P sum to one, the EMML would have had the iterative step

$$x_j^{k+1} = x_j^k \Big[\sum_{i=1}^{I} P_{ij}\Big(\frac{y_i}{(Px^k)_i}\Big)\Big] / \Big[\sum_{i=1}^{I} P_{ij}\Big]. \tag{4.3}$$

The OSEM method was derived from equation (4.3) by replacing both sums in (4.3) with partial sums over just those i in B_n [25]. The OSEM has the following iterative step: with $n = k(\operatorname{mod} N) + 1$

The OSEM:

$$x_j^{k+1} = x_j^k \Big[\sum\nolimits^n P_{ij}\Big(\frac{y_i}{(Px^k)_i}\Big)\Big] / \Big[\sum\nolimits^n P_{ij}\Big]. \tag{4.4}$$

where $\sum^n$ denotes the sum over those indices i in B_n.

The OSEM is mathematically incorrect. To be specific, it fails to converge to a solution in the consistent case, except for the quite special case of subset balance. The partition is said to have the *subset balance* property if, for each fixed value of j, the sums $\sum^n P_{ij}$ are independent of n. The OSEM produces, in the consistent case, limit cycles typical of the behavior of block-iterative methods in the noisy (or inconsistent) case; in the inconsistent case, it is noisier still. How distinct from one another the vectors of this limit cycle are depends on the extent to which subset balance fails, as much as on the relative noise level.

The corrected version of OSEM, called the *rescaled block-iterative* EMML (RBI-EMML) method, was presented in [6] (see also [7] and [8]). The RBI-EMML has the following iterative step:

The RBI-EMML:

$$x_j^{k+1} = (1 - m_n^{-1} \sum\nolimits^n P_{ij}) x_j^k + m_n^{-1} x_j^k \sum\nolimits^n P_{ij}\Big(\frac{y_i}{(Px^k)_i}\Big), \tag{4.5}$$

where $m_n = \max_j\{\sum^n P_{ij}\}$. When subset balance holds, the RBI-EMML reduces to the OSEM. The RBI-EMML converges, in the consistent case, to a solution, for every choice of subsets. In the inconsistent case the RBI-EMML produces a limit cycle; how distinct from one another the vectors of the limit cycle are depends on how large the minimum value of $KL(y, Px)$ is. In contrast, the OSEM, applied in the inconsistent case, produces a limit cycle with the differences between vectors dependent not only on the noise in the data vector y but also on the deviation from subset balance. This causes the OSEM to appear noisier than it should.

Both the OSEM and the RBI-EMML appear noisier than EMML in the inconsistent case, early in the iteration, for another reason. In the inconsistent case, the ML solution can have at most $I-1$ nonzero entries (for almost all matrices P)[3]; if there are more unknowns than equations ($J > I$) then this means the ML solution will have zero entries and these tend to be sprinkled throughout the image. Fast methods such as OSEM and RBI-EMML get near this poor ML solution sooner than the EMML algorithm does, so they look noisier.

There is another reason why block-iterative reconstructions can appear noisier than their simultaneous counterparts. The individual vectors in the limit cycle have their own noise component; if we averaged over the vectors of the limit cycle to get the final result, instead of simply taking the last vector computed, the noise would be somewhat smoothed.

The RBI-EMML algorithm converges in the consistent case to a nonnegative solution of the linear system $y = Px$. As with ART, strong underrelaxation can be used to achieve convergence in the inconsistent case. Such a method, called the *row-action maximum likelihood algorithm* (RAMLA), was discovered independently by Browne and De Pierro [2]. The RAMLA has the following iterative step:

The RAMLA:

$$x_j^{k+1} = (1 - \lambda_k \sum^n P_{ij})x_j^k + \lambda_k x_j^k \sum^n P_{ij}\Big(\frac{y_i}{(Px^k)_i}\Big), \tag{4.6}$$

where the positive relaxation parameters λ_k converge to zero and $\sum_{k=0}^{+\infty} \lambda_k = +\infty$.

Before leaving this section, we point out that when there are $N = I$ blocks, so that each B_n contains a single value of i, the RBI-EMML algorithm provides an analogue of the REMART in equation (3.6):

The RBI-EMML for N=I: for $k = 0, 1, ...$ and $i = k(\mathrm{mod}\, I) + 1$ let

$$x_j^{k+1} = (1 - m_i^{-1}P_{ij})x_j^k + m_i^{-1}x_j^k P_{ij}\Big(\frac{y_i}{(Px^k)_i}\Big), \tag{4.7}$$

where $m_i = \max_j\{P_{ij}\}$.

We have been considering a number of examples of block-iterative algorithms derived from simultaneous ones. But how can we obtain block-iterative methods directly from the optimization problem? We consider this issue in the next section.

5 Incremental optimization

We began by discussing the problem of minimizing functions of the form

$$h(x) = \sum_{i=1}^{I} h_i(x), \tag{5.1}$$

where $h_i(x)$ typically measures the extent to which the (vectorized) image x is consistent with one of the measured data values. Let us see how we might obtain block-iterative algorithms for such problems.

In [12] a block-iterative *interior point algorithm*, the IPA, was presented. The objective of this algorithm is to minimize a convex function $h(x)$ subject to certain constraints on the vector x. The IPA is called an interior point method because the vector x^k obtained at each step satisfies the desired constraints. The constraints on x are imposed through the use of another convex function $F(x)$ defined to be finite only on those vectors x that satisfy the constraints.

Denote by $h^n(x)$ the sum

$$h^n(x) = \sum\nolimits^n h_i(x), \tag{5.2}$$

where $\sum^n$ denotes the sum over those i in B_n.

Our interior point algorithm (IPA) is the following [12]:

Algorithm 5.1 The IPA: *For $k = 0, 1, ...,$ and $n = k(\bmod N) + 1$ and having determined x^k, let x^{k+1} be the unique solution of the gradient equation*

$$\nabla F(x^{k+1}) = \nabla F(x^k) - t_n \sum\nolimits^n \nabla h_i(x^k), \tag{5.3}$$

where $t_n > 0$ is chosen so that $f^n(x) = F(x) - t_n \sum^n h_i(x)$ is convex.

Subject to certain technical conditions placed on the behavior of the functions F and h^n, the algorithm converges whenever there are common minimizers of the functions $h^n(x)$; this seems contrived in the case of optimization, but for the functions $h(x)$ that occur in our examples, it simply means that there are vectors x that satisfy the constraints and match the measured data. Unless strong underrelaxation is used with incremental methods, as Bertsekas does in [1] and as was done in [15] and in [2], we would expect the algorithms to converge only when there are common minimizers of the functions $h^n(x)$. Otherwise, we expect subsequential convergence to a limit cycle, which is what we observe in all the examples. Except for ART, however, there is no proof of the existence of limit cycles for incremental methods.

The proof of convergence of the IPA, as given in [12], reveals that the rate of convergence is nearly proportional to the size of the relaxation parameters t_n. On the other hand, we need the functions f^n to be convex, which places upper bounds on the t_n. Accelerating convergence of the IPA then involves selecting the t_n as large as possible, subject to the convexity constraints. We shall illustrate this point in the examples to follow.

The proof of convergence of the IPA involves assumptions about the functions F and h^n and theoretical properties of Bregman-Legendre functions and will not be given here; see [12] for details. Instead, we shall illustrate this algorithm with several examples.

5.1 BI-ART as a case of the IPA:

Consider the system of equations $Ax = b$. Let $h(x) = \frac{1}{2}||Gx - c||^2$, where G and c are given by equations (2.4) and (2.5), respectively. Let G_n and c^n be as in the discussion of BI-ART earlier. Let $h_i(x) = \frac{1}{2}((Gx)_i - c_i)^2$. Let $F(x) = \frac{1}{2}||x||^2$ and $t_n \leq 1/L_n$, where L_n is the largest eigenvalue of the matrix $G_n^T G_n$. Then the IPA becomes relaxed BI-ART.

5.2 RBI-SMART as a case of the IPA:

Consider the system of linear equations $y = Px$, as in the earlier discussion of MART. Let $h(x) = KL(Px, y)$ and $h_i(x) = KL(y_i, (Px)_i)$. Let $F(x) = \sum_{j=1}^J x_j \log x_j - x_j$. Let $t_n = m_n^{-1}$ as defined in the discussion of RBI-SMART. Then the IPA becomes RBI-SMART.

If, for each j, we have $\alpha_j < \beta_j$, then we can enforce the constraints that x be such that $\alpha_j \leq x_j \leq \beta_j$ by taking $F(x) = F_{\alpha\beta}(x)$, defined by

$$F_{\alpha\beta}(x) = \sum_{j=1}^{J} (x_j - \alpha_j) \log(x_j - \alpha_j) + (\beta_j - x_j) \log(\beta_j - x_j). \tag{5.4}$$

In [12] this and other examples are discussed in more detail.

5.3 An IPA algorithm for transmission tomography:

In [29] we considered the application of the IPA to transmission tomography image reconstruction. In this case the function $h(x)$ we wish to minimize is usually taken to be the regularized negative log-likelihood function associated with a Poisson model, given, to within an additive constant, by

$$h(x) = \sum_i KL(y_i, c_i \exp(-(Lx)_i)). \tag{5.5}$$

Here y_i denotes the count associated with the ith line segment through the object, the entry L_{ij} of the matrix L is the length of the intersection of the ith line segment with the jth pixel and constant $c_i > 0$ denotes the input intensity of the radiation along the ith line segment. Let $h_i(x) = KL(y_i, c_i \exp(-(Lx)_i))$ and $h^n(x) = \sum^n h_i(x)$.

With the selection of the function $F(x)$ we impose desired constraints on the reconstructed image. In order to obtain reconstructed images $x = (x_1, ..., x_J)^T$ with $0 \leq \alpha_j \leq x_j \leq \beta_j$, for $j = 1, ..., J$, we shall use the function $F(x) = F_{\alpha\beta}(x)$ defined in (5.4). The essential domain of F is the set $D = \{x | \alpha_j \leq x_j \leq \beta_j,\ j = 1, ..., J\}$. Then, at the k-th step, we solve

$$\nabla F(x^{k+1}) = \nabla F(x^k) - t_n \nabla h^n(x^k), \tag{5.6}$$

where $t_n > 0$ is to be chosen so that the function $f^n(x) = F(x) - t_n h^n(x)$ is convex. The algorithm we obtain is the following: having calculated x^k, let x^{k+1} be determined by (5.6). Then we have

Algorithm 5.2 *Let $\alpha < x^0 < \beta$ be chosen. For $k = 0, 1, 2, ...$ and $n = k(\text{mod} N) + 1$, let*

$$x_j^{k+1} = \omega_j^k \alpha_j + (1 - \omega_j^k)\beta_j, \tag{5.7}$$

where

$$\omega_j^k = (\beta_j - x_j^k) / \Big[(\beta_j - x_j^k) + (x_j^k - \alpha_j) \exp(-t_n \frac{\partial h^n}{\partial x_j}(x^k))\Big]. \tag{5.8}$$

Using as h the function defined in (5.5) we get
the transmission AB algorithm (TAB): with

$$\omega_j^k = (\beta_j - x_j^k) / \Big[(\beta_j - x_j^k) + (x_j^k - \alpha_j) \exp(-t_n \sum^n (L_{ij}(y_i - c_i \exp(-(Lx^k)_i))))\Big] \tag{5.9}$$

the iterative step is

$$x_j^{k+1} = \omega_j^k \alpha_j + (1 - \omega_j^k)\beta_j. \tag{5.10}$$

To conclude from the theory of the IPA that we have convergence to a constrained minimizer of the function $h(x)$, we must choose t_n so that, among other things, the function $F(x) - t_n h^n(x)$ is a convex function. We obtain an upper limit on the acceptable values of t_n by considering the Hessian matrix of the function f^n; for details see [29].

5.4 Regularized block-iterative methods

In most of the examples we have considered here the function $h(x0$ to be minimized measures the extent to which a given image x matches the measured data. When the data is noisy, we may not want an exact match; regularization provides for smoother reconstructions by imposing a trade-off between matching the data and satisfying other, more qualitative, conditions, such as smoothness. For example, consider the EMML algorithm. Instead of minimizing the function $h(x) = KL(y, Px)$, as the EMML does, we might consider minimizing $h(x) = KL(y, Px) + \gamma U(x)$, where $U(x)$ is a penalty function that imposes prior soft constraints, such as smoothness, on x. To enforce nonnegativity in x, we select $F(x) = \sum_{j=1}^J x_j \log x_j - x_j$ as before. The IPA iterative step is obtained by solving the following equations:
the IPA for regularized maximum likelihood:

$$\log x_j^{k+1} = \log x_j^k - \tau \Big[\Big(\sum_{i=1}^I P_{ij} \Big(1 - \frac{y_i}{(Px^k)_i} \Big) \Big) + \gamma \frac{\partial U}{\partial x_j}(x^k) \Big], \tag{5.11}$$

where $\tau > 0$ is selected so that the function $F(x) - \tau h(x)$ is strictly convex (as well as satisfying the other theoretical conditions for Bregman-Legendre functions). In practice, it is usually easier to find suitable τ experimentally than by using the theoretical conditions.

In the next two sections we present a convergence theorem for the RESART and RE-BI-ART.

6 A convergence theorem for RESART

The relaxed SART algorithm (RESART) is a simultaneous iterative procedure for solving the system of linear equations $Gx = c$, where, for now, G is an arbitrary real I by J matrix.

Algorithm 6.1 : the RESART algorithm *Let x^0 be arbitrary. For $k = 0, 1, 2, \ldots$, having calculated the vector x^k, the next member of the iterative sequence is given by*

$$x^{k+1} = x^k + \gamma_k G^T (c - Gx^k). \tag{6.1}$$

The γ_k are positive scalars.

We prove that the RESART algorithm converges to a least squares solution for suitably restricted relaxation parameters. For related results see Theorem 1 in the paper of Lakshminarayanan and Lent [27] and also the paper by Dax [19].
We have the following result:

Theorem 6.1 : the RESART theorem *Let L be the maximum eigenvalue of the matrix G^TG. Let $\gamma_k > 0$ be chosen so that, for some $\epsilon > 0$ and positive integer K, and for all $k = K, K+1, \ldots$, we have $\gamma_k \leq (2/L) - \epsilon$ and $\sum_{k=0}^{+\infty} \gamma_k = +\infty$. Then the iterative sequence $\{x^k\}$ obtained from equation (6.1) converges to the minimizer of the function $||Gx - c||$ for which the distance $||x - x^0||$ is minimized.*

When the matrix G is given by equation (2.4) the iteration in (6.1) is a relaxed version of the SART. In that case, the trace of GG^T, which is the sum of the eigenvalues of G^TG, is I. Therefore $L \leq I$, although this is a quite conservative estimate, in most cases, particularly when A is sparse, as we shall see. Consequently, SART converges, but quite slowly, in most cases.

We can get some feel for why methods such as SART can converge slowly, in comparison to ART. It often happens that the measured data in image reconstruction is somewhat redundant. To take an extreme case, for illustration, imagine that we repeat each equation in $Ax = b$ many times. It has no effect on the SART algorithm. However, for the ART algorithm, if the equations are ordered in such a way that duplicate equations do not occur in succession, then the effect on ART is more like several iterations through the original set of equations. This also illustrates the importance of the order of the equations in ART, or, more precisely, the importance of avoiding a particularly unfortunate ordering.

6.1 Proof of the RESART theorem:

To prove the RESART theorem, we need several preliminary results. For the remainder of this section we let $\hat{x}$ be an arbitrary minimizer of the function $f(x) = ||Gx - c||$; then we have $G^TG\hat{x} = G^Tc$. Expanding norms in terms of inner products, we find that

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 = 2(\hat{x} - x^k)^T(x^{k+1} - x^k) - ||x^k - x^{k+1}||^2. \tag{6.2}$$

Using $G^T G\hat{x} = G^T c$ and equation (6.1), we have

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 = 2\gamma_k ||G\hat{x} - Gx^k||^2 - ||x^k - x^{k+1}||^2. \tag{6.3}$$

Using equation (6.1) again to write

$$||x^k - x^{k+1}||^2 = (\gamma_k)^2 ||G^T(c - Gx^k)||^2 = (\gamma_k)^2 ||G^T(G\hat{x} - Gx^k)||^2,$$

we obtain

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 = 2\gamma_k ||G\hat{x} - Gx^k||^2 - (\gamma_k)^2 ||G^T(G\hat{x} - Gx^k)||^2. \tag{6.4}$$

Since $||G^T u||^2 \leq L||u||^2$ for any u, it follows that

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 \geq (2\gamma_k - (\gamma_k)^2 L)||G\hat{x} - Gx^k||^2 \geq 0. \tag{6.5}$$

Proof of the theorem: We can conclude from (6.5) that the sequence $\{||\hat{x} - x^k||\}$ is decreasing, the sequence $\{x^k\}$ is bounded and the sequence $\{(2\gamma_k - (\gamma_k)^2 L)||G\hat{x} - Gx^k||^2\} \to 0$. Since $2 - (\gamma_k)L \geq \epsilon > 0$ we have

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 \geq \epsilon\gamma_k ||G\hat{x} - Gx^k||^2, \tag{6.6}$$

for $k \geq K$. Summing over $k = K, K+1, ...$ on both sides and using the fact that $\sum_{k=0}^{+\infty} \gamma_k = +\infty$, we conclude that the sequence $\{||G\hat{x} - Gx^k||^2\} \to 0$.

Therefore, for any cluster point x^* of the bounded sequence $\{x^k\}$ we have $||Gx^* - c|| = ||G\hat{x} - c||$. Therefore x^* can be used in place of $\hat{x}$, to obtain that the sequence $\{||x^* - x^k||\}$ is decreasing. It follows then that $x^k \to x^*$.

Since the right side of equation (6.3) does not depend on which particular minimizer of $||Gx - c||$ we have chosen, neither does the left side. Summing both sides over k, we can conclude that $||\hat{x} - x^0||^2 - ||\hat{x} - x^*||^2$ is also independent of which minimizer of $||Gx - c||$ we have chosen; therefore x^* is the minimizer of $||Gx - c||$ for which $||x - x^0||$ is minimized. This completes the proof of the theorem. ∎

Remark: Inequality (6.5) suggests that the RESART algorithm will converge faster if the relaxation parameters γ_k are near $1/L$, the maximum of $2\gamma_k - (\gamma_k)^2 L$.

6.2 The RESART for sparse matrices

As we noted earlier, for the matrix G given by equation (2.4), the conclusion that eigenvalues of $G^T G$ are not larger than I gives a fairly conservative estimate, in most cases. In particular, if the matrix G (equivalently, A) is sparse, we can obtain a much better estimate and, thereby, faster convergence using RESART.

For each j let s_j denote the number of nonzero entries in the j-th column of the matrix A and let s be the maximum of the s_j, over all j. We are interested in the case in which s is much smaller than I, the so-called sparse matrix case. The idea of using sparseness to accelerate convergence was suggested by the work of Censor, Gordon and Gordon in [16].

Let D be the diagonal matrix with entries

$$D_{ii} = \sum_{j=1}^{J} s_j A_{ij}^2, \tag{6.7}$$

and let $F = D^{-1/2}A$, so that $F^T F = A^T D^{-1} A$.
The following lemma is due to Arnold Lent [28].

Lemma 6.1 *Let F be the matrix given above. Then the eigenvalues of $F^T F$ are not greater than one.*

Proof: Let v be a nonzero eigenvector of $F^T F$, with $A^T D^{-1} A v = \lambda v$. Then $AA^T D^{-1} A v = \lambda A v = \lambda D(D^{-1} A v)$. With $w = D^{-1} A v$ we then have $AA^T w = \lambda D w$, so that $w^T A A^T w = \lambda w^T D w$, or $||A^T w||^2 = \lambda w^T D w$. Writing out both sides of the last equation and switching the order of summation, we obtain

$$\sum_{j=1}^{J} \Big(\sum_{i=1}^{I} A_{ij} w_i \Big)^2 = \lambda \sum_{i=1}^{I} w_i^2 \Big(\sum_{j=1}^{J} s_j A_{ij}^2 \Big) = \lambda \sum_{j=1}^{J} s_j \Big(\sum_{i=1}^{I} w_i^2 A_{ij}^2 \Big).$$

From Cauchy's inequality we have

$$\Big(\sum_{i=1}^{I} A_{ij} w_i \Big)^2 \leq \Big(\sum_{i=1}^{I} w_i^2 A_{ij}^2 \Big) s_j.$$

So we have

$$\sum_{j=1}^{J} \Big(\sum_{i=1}^{I} A_{ij} w_i \Big)^2 \leq \sum_{j=1}^{J} s_j \Big(\sum_{i=1}^{I} w_i^2 A_{ij}^2 \Big).$$

The result follows. ∎

Corollary 6.1 *For G given by equation (2.4), the maximum eigenvalue of the matrix $G^T G$ is not greater than s.*

Proof: Replacing zero entries of G with very small nonzero entries as needed, we modify G so that s_j is the same for each j. The result then follows from Lent's lemma and the continuity of the eigenvalues as a function of the entries of the matrix G. ∎

It follows from the RESART theorem and the above corollary that the factor $1/I$ in SART can be replaced by $1/s$ without destroying convergence.

In the next section we prove convergence of RE-BI-ART, for the consistent case.

7 Convergence of the RE-BI-ART algorithm

The *relaxed block-iterative* ART (RE-BI-ART) is an iterative procedure for solving the system of linear equations $Ax = b$. We prove that RE-BI-ART converges in the consistent case, for suitably restricted relaxation parameters. For notational convenience, we shall consider an algorithm that appears to be somewhat more general than BI-ART, which we call the *generalized block-iterative* ART (GBI-ART). The GBI-ART is RE-BI-ART applied to a system of linear equations equivalent to $Ax = b$, that is, a system having the same solutions.

Let G be a matrix with the same number of columns as the matrix A. We say that A and G are *interchangeable* if, whenever $Ax = b$ is consistent, there is a vector c such that $Ax = b$ and $Gx = c$ have the same set of solutions. We have the following result:

Proposition 7.1 *The following are equivalent:*

1. *A and G are interchangeable;*
2. *$Ax = 0$ if and only if $Gx = 0$;*
3. *there are matrices D and C with $G^T = A^T D$ and $A^T = G^T C$.*

Proof: Assume A and G are interchangeable. Then, since $Ax = 0$ is consistent, there is c so that $Ax = 0$ if and only if $Gx = c$. Using $x = 0$, we find that $c = 0$. So $Ax = 0$ if and only if $Gx = 0$.

Now assume $Ax = 0$ if and only if $Gx = 0$; that is, the null space of A is the same as the null space of G in R^J. Since the column space of A^T is the orthogonal complement of the null space of A, it follows that the matrices A^T and G^T have the same column spaces in R^J. This means that each column of A^T is a linear combination of the columns of G^T, and conversely. Therefore there are matrices D and C such that $A^T = G^T C$ and $G^T = A^T D$.

Finally, assume that there are matrices D and C such that $A^T = G^T C$ and $G^T = A^T D$. We show that A and G are interchangeable. Suppose that $Ax = b$ is consistent. Let $c = D^T b$. If $Az = b$ then $Gz = D^T Az = D^T b = c$. On the other hand, if $Gz = c$ then $Az = C^T Gz = C^T c = C^T D^T b$. Since we have assumed that $Ax = b$ is consistent, let x be such that $Ax = b$. Then we have $Az = C^T D^T b = C^T D^T Ax$. But $C^T D^T A = C^T G = A$ so $Ax = Az = b$. This completes the proof. ▮

Given I by J matrix A, how do we find other matrices interchangeable with A? One way is to select some positive integer M and an M by I matrix D^T that is one-to -one and let $G = D^T A$. Then $Gx = D^T Ax = 0$ implies $Ax = 0$, since D^T is one-to-one. But the latter condition requires that $M \geq I$. Applying the same condition to

C^T we would need to use $M = I$. But there are examples of interchangeable matrices that do not have the same number of rows.

Example: Let G be the $2I$ by I matrix with $G^T = [A^T A^T]$. Then $D = [I\ I]$ and $C^T = [\frac{1}{2}I\ \frac{1}{2}I]$. The matrix C^T is not one-to-one, but it is one-to-one when restricted to the range of G. The matrices A and G are interchangeable.

Suppose that the dimension of the range of the matrix A is L. Let $\{z^1, ..., z^L\}$ be a basis for the range of A in R^I. Define the linear transformation U from R^I to R^L by

$$U(Ax + v) = (\alpha_1(Ax), ..., \alpha_L(Ax))^T,$$

where $A^T v = 0$ and $Ax = \alpha_1(Ax)z^1 + ... + \alpha_L(Ax)z^L$. Then let $M \geq L$ and W be any M by L matrix that is one-to-one and take $D^T = WU$. Then $G = D^T A$ and A are interchangeable.

For the remainder of this paper we shall adopt the following notation. Let M be some positive integer and G an M by J matrix such that A and G are interchangeable. Let $N \leq M$ be a positive integer and let $\{B_n,\ n = 1, ..., N\}$ be a partition of the set $\{m = 1, ..., M\}$. Let I_n be the cardinality of the block B_n and G_n the I_n by J matrix obtained from G by deleting the m-th row of G for all m not in B_n. From the relations $G^T = A^T D$ and $A^T = G^T C$ we obtain, for each $n = 1, ..., N$, the matrices D_n, C_n and G_n with $G_n^T = A^T D_n$ and $A^T = \sum_{n=1}^N G_n^T C_n$. If $Ax = b$ is consistent, then $Ax = b$ if and only if $G_n x = D_n^T b = c^n$ for each n.

7.1 A generalization of BI-ART

We assume now that the system $Ax = b$ is consistent. We generalize the BI-ART method by applying block-iteration to the equivalent system $Gx = D^T b = c$. The block structure is as just described above.

Algorithm 7.1 the GBI-ART algorithm: *Let x^0 be arbitrary. For $k = 0, 1, 2, ...$ and $n = n(k) = k(\mathrm{mod}\, N) + 1$, having calculated the vector x^k, the next member of the iterative sequence is given by*

$$x^{k+1} = x^k + \gamma_k G_n^T(c^n - G_n x^k). \tag{7.1}$$

The γ_k are positive scalars.

Choosing G_n and c^n properly, the BI-ART is seen to be a special case of the GBI-ART algorithm.

We have the following result:

Theorem 7.1 the GBI-ART theorem: *Let there be solutions of the system $Ax = b$. For $n = 1, 2, ..., N$, let L_n be the maximum eigenvalue of the matrix $G_n^T G_n$. For $k = 0, 1, ...$ and $n = n(k) = k(\mathrm{mod}\, N) + 1$, let $\gamma_k > 0$ be chosen so that, for some $\epsilon > 0$ and positive integer K, and for all $k = K, K+1, ...$, we have $\gamma_k \leq (2/L_n) - \epsilon$ and $\sum_{k=0}^{+\infty} \gamma_k = +\infty$. Then the iterative sequence $\{x^k\}$ obtained from equation (7.1) converges to the solution of $Ax = b$ for which the distance $||x - x^0||$ is minimized.*

7.2 Proof of the GBI-ART theorem:

To prove the GBI-ART theorem, we need several preliminary results. For the remainder of this section we let $\hat{x}$ be an arbitrary solution of $Ax = b$. Expanding norms in terms of inner products, we find that

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 = 2(\hat{x} - x^k)^T(x^{k+1} - x^k) - ||x^k - x^{k+1}||^2. \qquad (7.2)$$

Using $G_n\hat{x} = c^n$ and equation (7.1), we have, for $n = n(k)$,

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 = 2\gamma_k||G_n\hat{x} - G_n x^k||^2 - ||x^k - x^{k+1}||^2. \qquad (7.3)$$

Using equation (7.1) again to write

$$||x^k - x^{k+1}||^2 = (\gamma_k)^2||G_n^T(c^n - G_n x^k)||^2 = (\gamma_k)^2||G_n^T(G_n\hat{x} - G_n x^k)||^2,$$

we obtain

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 = 2\gamma_k||G_n\hat{x} - G_n x^k||^2 - (\gamma_k)^2||G_n^T(G_n\hat{x} - G_n x^k)||^2. \quad (7.4)$$

Since $||G_n^T u||^2 \leq L_n||u||^2$ for any u, it follows that

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 \geq (2\gamma_k - (\gamma_k)^2 L_n)||G_n\hat{x} - G_n x^k||^2 \geq 0. \qquad (7.5)$$

Proof of the theorem: We can conclude from (7.5) that the sequence $\{||\hat{x} - x^k||\}$ is decreasing, the sequence $\{x^k\}$ is bounded and the sequence $\{(2\gamma_k - (\gamma_k)^2 L_n)||G_n\hat{x} - G_n x^k||^2\} \to 0$. Since, for $k \geq K$, $2 - (\gamma_k)L_n \geq \epsilon > 0$, we have

$$||\hat{x} - x^k||^2 - ||\hat{x} - x^{k+1}||^2 \geq \epsilon\gamma_k||G_n\hat{x} - G_n x^k||^2, \qquad (7.6)$$

for $k \geq K$. Summing over $k = K, K+1, ...$ on both sides and using the fact that $\sum_{k=0}^{+\infty} \gamma_k = +\infty$, we conclude that the sequence $\{||G_n\hat{x} - G_n x^k||^2\} \to 0$. It follows that the sequence $\{||x^{k+1} - x^k||\} \to 0$ also.

Because the sequence $\{x^k\}$ is then bounded, so is the subsequence $\{x^{mN+1} | m = 1, 2, ...\}$. Let $x^{*,1}$ be a cluster point of the latter sequence. Then extract a subsubsequence of $\{x^{mN+2}\}$ converging to $x^{*,2}$, etc. We end up with an infinite sequence of positive integers, $\{m_l | l = 1, 2, ...\}$, such that, for each fixed n the sequence $\{x^{m_l N+n}\}$ converges to $x^{*,n}$. We also know then that $G_n x^{*,n} = c^n$. Because the sequence $\{||x^{k+1} - x^k)||\}$ converges to zero, it follows that $x^{*,n} = x^{*,n+1}$, for $n = 1, ..., N-1$. Consequently, the cluster points are equal, that is, $x^{*,n} = x^*$ for all n and $Gx^* = c$.

Using x^* in place of $\hat{x}$, we find that $||x^* - x^k||^2$ is decreasing. But a subsequence converges to zero. Therefore the entire sequence converges to zero.

Since the right side of equation (7.3) does not depend on which particular solution of $Ax = b$ we have chosen, neither does the left side. Summing both sides over k, we can conclude that $||\hat{x} - x^0||^2 - ||\hat{x} - x^*||^2$ is also independent of which solution of we have chosen; therefore x^* is the solution of $Ax = b$ for which $||x - x^0||$ is minimized. This concludes the proof. ∎

Several of the iterative methods we have considered so far can be put into the following general framework. Having determined x^k select a *direction vector* p^k and scalar α_k so that the next term in the iteration is

$$x^{k+1} = x^k + \alpha_k p^k.$$

In the case of ART we take p^k to be the i-th column of A^T and then select α_k so that $A(x^{k+1})_i = b_i$. In the case of RESART the p^k is the gradient of the function $f(x) = \frac{1}{2}||Ax - b||^2$, evaluated at x^k and the α_k is selected subject to constraints that guarantee convergence. In the case of GBI-ART we have $p^k = G_n^T(c^n - G_n x^k)$ and $\alpha_k = \gamma_k$.

In the *conjugate direction* method the p^k have the property $(p^j)^T A^T A p^l = 0$ for $j, l = 0, 1, ..., J-1$ and $j \neq l$. Then the α_k are chosen so that $\alpha = \alpha_k$ minimizes the function $||A(x^k + \alpha p^k) - b||$. In the *conjugate gradient* (CG) method the conjugate directions p^k are calculated as the iteration proceeds, with convergence to a least squares solution of $Ax = b$ guaranteed (in theory) in at most J steps. The LS-ART also employs the idea of *optimal relaxation*; the p^k is a^i, the i-th column of A^T, as in ART, but now we select α_k so that $\alpha = \alpha_k$ minimizes the function $||A(x^k + \alpha a^i) - b||$.

Once the p^k is selected, it is not too difficult to choose an appropriate scalar α_k; we can pick these scalars simply as relaxation parameters, provided that they satisfy certain constraints necessary for convergence, or we can choose the scalars to meet certain other objectives, as in ART. But how do we select the p^k in the first place?

A common approach, used in SART, is to take the direction that is best locally, namely the (negative of the) gradient at x^k. This requires considerable computation, however, and may not lead to a fast algorithm, in any case. The ART, by contrast, puts no effort into selecting the p^k, taking the already available a^i. The CG method employs direction vectors p^k that are A^TA-orthogonal, which leads to convergence in finitely many steps, but with no guarantee that the steps made early in the iteration are good ones. Experience with ART suggests that it is not necessary to spend too much effort obtaining good p^k, so long as we select the scalars α_k reasonably. However, because the ART scalars are not chosen optimally, with respect to the overall minimization problem, that is, we get limit cycles in the inconsistent case. The LS-ART is designed to converge even in the inconsistent case, by using scalars that are optimally selected.

The ideal case would seem to involve some modest effort at selecting good p^k, combined with optimal relaxation. In the next section we consider such an approach that combines the GBI-ART approach with the idea of optimal relaxation, to obtain convergence to a least squares solution in the inconsistent case.

8 Optimally relaxed methods and extensions

We extend the GBI-ART iterative algorithm to avoid limit cycles and to obtain a least squares solution of $Ax = b$. The basic idea is to use the matrices G_n to describe the form of the increment $x^{k+1} - x^k$ and then to minimize $||Ax - b||$ over x having the desired form. Using the same notation as in the previous section, we consider an iterative step of the form

$$x^{k+1} = x^k + G_n^T z^k, \tag{8.1}$$

where $n = n(k) = k(\mathrm{mod}\, N) + 1$ and z^k is a vector to be chosen. Let us select $z = z^k$ to minimize the function $||A(x^k + G_n^T z) - b||^2$ over all vectors z. A little matrix algebra reveals that the solution is

$$z^k = (G_n A^T A G_n^T)^{-1} G_n A^T (b - Ax^k). \tag{8.2}$$

If $N = I$ and the matrix G_n^T is the column vector a^i, then we get the LS-ART algorithm. If $N = J$ and the matrix G_n^T is the conjugate direction p^j, for $n = 0, 1, .., J - 1$, then we get the CG method. We have the following theorem:

Theorem 8.1 *The sequence $\{x^k\}$ given by (8.1) converges to the least squares solution of $Ax = b$ for which the distance $(x - x^0)^T A^T A(x - x^0)$ is minimized.*

To prove this theorem we need a few lemmas.

Lemma 8.1 *For $n = n(k)$ we have $G_n A^T (Ax^{k+1} - b) = 0$.*

Proof: We have

$$G_nA^T(Ax^{k+1}) = G_nA^T(Ax^k + AG_n^Tz^k)$$

$$= G_nA^T(Ax^k) + G_nA^T(AG_n^T(G_nA^TAG_n^T)^{-1}G_nA^T(b - Ax^k))$$

$$= G_nA^T(Ax^k) + G_nA^T(b - Ax^k) = G_nA^Tb.$$

The conclusion follows. ∎

Lemma 8.2 *For each $k = 0, 1, ...$ we have*

$$||Ax^k - b||^2 = ||Ax^{k+1} - b||^2 + ||Ax^k - Ax^{k+1}||^2$$

Proof: We have

$$||Ax^k - b||^2 = ||Ax^k - Ax^{k+1} + Ax^{k+1} - b||^2$$

$$= ||Ax^k - Ax^{k+1}||^2 - 2\langle Ax^{k+1} - Ax^k, Ax^{k+1} - b\rangle + ||Ax^{k+1} - b||^2.$$

Using $Ax^{k+1} - Ax^k = AG_n^Tz^k$ and the previous lemma, it follows that $2\langle Ax^{k+1} - Ax^k, Ax^{k+1} - b\rangle = 0$. The result follows. ∎

Therefore we know that the sequence $\{||Ax^k - b||\}$ is decreasing and the sequence $\{||Ax^k - Ax^{k+1}||\}$ converges to zero. Now let $\hat{x}$ be an arbitrary least squares solution of $Ax = b$; that is, let $A^TA\hat{x} = A^Tb$. Let $||u||^2_{A^TA} = u^TA^TAu$ for any vector u.

Lemma 8.3 *For each $k = 0, 1, ...$ and $\hat{x}$ as above, we have*

$$||\hat{x} - x^k||^2_{A^TA} = ||\hat{x} - x^{k+1}||^2_{A^TA} + ||x^k - x^{k+1}||^2_{A^TA}.$$

Proof: Expanding the norms using inner products, as was done in the proof of the previous lemma, we obtain

$$||\hat{x} - x^k||^2_{A^TA} - ||\hat{x} - x^{k+1}||^2_{A^TA} - ||x^k - x^{k+1}||^2_{A^TA}$$

$$= 2(z^k)^TG_nA^T(A\hat{x} - Ax^{k+1}) = 0.$$

The last equation follows from $A^TA\hat{x} = A^Tb$ and $G_nA^T(Ax^{k+1} - b) = 0$. ∎

So we now know that the sequence $\{||\hat{x} - x^k||_{A^TA}\}$ is decreasing and that the sequence $\{||x^k - x^{k+1}||^2_{A^TA} = ||Ax^k - Ax^{k+1}||^2\}$ converges to zero. Since the sequence $\{||Ax^k - b||^2\}$ is decreasing, the sequence $\{Ax^k\}$ is bounded. We show next that the sequence $\{x^k\}$ is also bounded.

For some vector u^0, we can write $x^0 = A^Tu^0 + w^0$, where $Aw^0 = 0$. Since $x^{k+1} - x^k$ lies in the space spanned by the columns of the matrix G_n^T, it follows from our assumption that the column spaces of A^T and G^T are the same that, for each k there is some u^k such that $x^k = A^T u^k + w^0$. Each u^k can be written as $u^k = Av^k + t^k$, with $A^T t^k = 0$. Therefore, we can write $x^k = A^T A v^k + w^0$. We know the sequence $\{Ax^k = AA^T Av^k\}$ is bounded. The following lemma is helpful here.

Lemma 8.4 *There is a positive scalar β with the property that, for all vectors v, we have $||Av|| \leq \beta ||AA^T Av||$.*

Proof: If no such β exists then there is a sequence $\{y^n = Av^n/||Av^n||\}$ such that $||AA^T y^n|| \to 0$. Since $||y^n|| = 1$ for each n and the unit ball is compact, we may assume that there is y^* with $y^n \to y^*$ and $||y^*|| = 1$. Each y^n has the form $y^n = As^n$, so we must have $y^* = As^*$ for some s^*. It follows that $AA^T As^* = 0$, so that $(s^*)^T A^T AA^T As^* = ||A^T As^*||^2 = 0$. Therefore $A^T As^* = 0$; consequently $(s^*)^T A^T As^* = ||As^*||^2 = 0$. So we have $y^* = As^* = 0$, which is contradictory, since $||y^*|| = 1$. This completes the proof. ∎

Proof of the theorem: Using the previous lemma and the fact that the sequence $\{Ax^k = AA^T Av^k\}$ is bounded, we conclude that the sequence $\{Av^k\}$ is also bounded. Therefore, the sequence $\{x^k = Av^k + w^0\}$ is bounded, as well.

We know that the sequence $\{||x^k - x^{k+1}||^2_{A^T A} = ||Ax^k - Ax^{k+1}||^2\}$ converges to zero. This means that, with $n = n(k)$, we have

$$[G_n A^T(b - Ax^k)]^T (G_n A^T A G_n^T)^{-1} [G_n A^T(b - Ax^k)] \to 0.$$

Since the matrix $(G_n A^T A G_n^T)^{-1}$ is positive definite, all its eigenvalues are positive. Therefore, it follows that the sequence $\{||G_{n(k)} A^T(b - Ax^k)||^2\}$ converges to zero.

Since the sequence $\{x^k\}$ is bounded, so is the subsequence $\{x^{mN}\}$; let $x^{*,0}$ be an arbitrary cluster point of the sequence $\{x^{mN}\}$. Specifically, let $x^{m_l N} \to x^{*,0}$. Then let $x^{*,1}$ be a cluster point of the sequence $\{x^{m_l N+1}\}$. Continue in this way to generate $x^{*,n-1}$ for $n = 1, 2, ..., N$. Since $||Ax^k - Ax^{k+1}||$ converges to zero, so does $\{||Av^k - Av^{k+1}||\}$, and therefore so does $\{||x^k - x^{k+1}||\}$. Therefore, we have $x^* := x^{*,0} = x^{*,n-1}$ for all n. From $||Ax^k - Ax^{k+1}|| \to 0$ it follows that $G_n A^T(b - Ax^*) = 0$ for all n. This means that $A^T(b - Ax^*)$ is in the null space of each of the G_n, hence in the null space of A. But it is also in the range of A^T; consequently, $A^T(b - Ax^*) = 0$, which says that x^* is a minimizer of the function $||Ax - b||$.

We can then replace $\hat{x}$ with x^*, to find that $||x^* - x^k||^2_{A^T A}$ is decreasing. But a subsequence converges to zero; therefore the entire sequence $\{||x^* - x^k||^2_{A^T A}\}$ converges to zero. Arguing as above, we conclude that $x^k \to x^*$.

Since the difference $||\hat{x} - x^k||^2_{A^TA} - ||\hat{x} - x^{k+1}||^2_{A^TA}$ does not depend on which particular minimizer of $||Ax - b||$ we have selected, neither does the difference $||\hat{x} - x^0||^2_{A^TA} - ||\hat{x} - x^*||^2_{A^TA}$. It follows that x^* is the minimizer of $||Ax - b||$ for which the distance $||x - x^0||^2_{A^TA}||$ is minimized. This completes the proof of the theorem. ∎

The square matrix $G_nA^TAG_n^T$ has dimensions I_n by I_n; if N is large and each I_n is small, performing the inversion indicated in (8.2) is feasible. In particular, if $I_n = 1$ for each n, the inversion is trivial. Consider a couple of special cases.

Example 1: Let $G^T_{n(k)} = a^{i(k)}$, the $i(k)$-th column of the matrix A^T. Then equation (8.1) becomes (2.14); the extended GBI-ART becomes LS-ART.

Example 2: Let $p^0, p^1, ..., p^{J-1}$ be A^TA-conjugate directions, that is, $\langle p^j, A^TAp^l\rangle = 0$ for $j \neq l$. In the conjugate gradient method such conjugate directions are calculated as the iteration proceeds. Let $G_n^T = p^{n-1}$ for $n = 1, 2, ..., N = J$. Then the extended GBI-ART becomes the conjugate gradient method.

The advantage of the conjugate gradient method is that convergence to a least squares solution of $Ax = b$ in not more than J steps is guaranteed. However, calculating the p^n as we proceed is expensive. Perhaps the least squares ART is preferable, if our goal is to obtain reasonable images is a few iterations.

9 Summary and Conclusions

We have presented a number of block-iterative image reconstruction methods as particular cases of an interior point incremental optimization algorithm. These algorithms involve relaxation parameters that, when increased, tend to accelerate convergence, although convexity conditions place upper bounds on these parameters. The use of blocks tends to increase these upper bounds, hence the observed accelerated convergence with such methods. For sparse matrices, significant acceleration without the use of blocks can also be achieved through overrelaxation. Optimal relaxation is introduced as a way of avoiding the limit cycles; extension of this idea to include optimal selection of the direction vector is also considered.

References

[1] Bertsekas, D.P. (1997) A new class of incremental gradient methods for least squares problems, *SIAM J. Optim.*, **7**, pp. 913-926.

[2] Browne, J., and De Pierro, A. R. (1996) A row-action alternative to the EM algorithm for maximizing likelihoods in emission tomography, *IEEE Transactions on Medical Imaging*, **15**, pp. 687–699.

[3] Byrne, C.L. (1993) Iterative image reconstruction algorithms based on cross-entropy minimization, *IEEE Transactions on Image Processing*, **IP-2**: 96–103.

[4] Byrne, C.L. (1995) Erratum and addendum to "Iterative image reconstruction algorithms based on cross-entropy minimization", *IEEE Transactions on Image Processing*, **IP-4**, 225–226.

[5] Byrne, C. L. (1996) Iterative reconstruction algorithms based on cross-entropy minimization, in: *Image Models (and their Speech Model Cousins)*, (S.E. Levinson and L. Shepp, Editors), the IMA Volumes in Mathematics and its Applications, Volume 80, Springer-Verlag, New York, pp. 1–11.

[6] Byrne, C. L. (1996) Block-iterative methods for image reconstruction from projections, *IEEE Transactions on Image Processing*, **IP-5**, pp. 792-794.

[7] Byrne, C. L. (1997) Convergent block-iterative algorithms for image reconstruction from inconsistent data, *IEEE Transactions on Image Processing*, **IP-6**, pp. 1296–1304.

[8] Byrne, C. L. (1998) Accelerating the EMML algorithm and related iterative algorithms by rescaled block-iterative methods, *IEEE Transactions on Image Processing*, **IP-7**, pp. 100-109.

[9] Byrne, C. L. (1998) Iterative algorithms for deblurring and deconvolution with constraints, *Inverse Problems*, **14**, pp. 1455-1467.

[10] Byrne, C. L. (1999) Iterative projection onto convex sets using multiple Bregman distances, *Inverse Problems*, **15**, pp. 1295-1313.

[11] Byrne, C. L. (1999) An approximate successive generalized projections method for convex programming, *submitted for publication.*

[12] Byrne, C. L. (2000) Block-iterative interior point optimization methods for image reconstruction from limited data, *Inverse Problems*, to appear.

[13] Byrne, C. L. (2000) Bregman-Legendre multidistance projection algorithms for convex feasibility and optimization, *Proceedings of the Workshop on Inherently*

Parallel Algorithms in Feasibility and Optimization, Haifa, March 2000, to appear.

[14] Censor, Y. (1981) Row-action methods for huge and sparse systems and their applications, *SIAM Review*, **23**, pp. 444–464.

[15] Censor, Y., Eggermont, P.P.B., and Gordon, D. (1983) Strong underrelaxation in Kaczmarz's method for inconsistent systems, *Numerische Mathematik*, **41**, pp. 83-92.

[16] Censor, Y., Gordon, D., and Gordon, R. (2000) Component averaging: An efficient iterative parallel algorithm for large and sparse unstructured problems, *Parallel Computing*, to appear.

[17] Censor, Y. and Segman, J. (1987) On block-iterative maximization, *J. of Information and Optimization Sciences*, **8**, pp. 275-291.

[18] Censor, Y., and Zenios, S.A. (1997) *Parallel Optimization: Theory, Algorithms and Applications*, Oxford University Press, New York.

[19] Dax, A. (1990) The convergence of linear stationary iterative processes for solving singular unstructured systems of linear equations, *SIAM Review*, **32**, pp. 611–635.

[20] Fiacco, A., and McCormick, G. (1990) *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, SIAM Classics in Mathematics.

[21] Gordon, R., Bender, R., and Herman, G.T. (1970) Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography, *J. Theoret. Biol.*, **29**, pp. 471-481.

[22] Herman, G.T., and Meyer, L. (1993) Algebraic reconstruction techniques can be made computationally efficient, *IEEE Transactions on Medical Imaging*, **12**, pp. 600-609.

[23] Herman, G.T. (1999) *private communication.*

[24] Holte, S., Schmidlin, P., Linden, A., Rosenqvist, G. and Eriksson, L. (1990) Iterative image reconstruction for positron emission tomography: a study of convergence and quantitation problems, *IEEE Transactions on Nuclear Science*, **37**, pp. 629–635.

[25] Hudson, H. M., and Larkin, R. S. (1994) Accelerated image reconstruction using ordered subsets of projection data, *IEEE Transactions on Medical Imaging*, **13**, pp. 601-609.

[26] Kaczmarz, S. (1937) Angenäherte Auflösung von Systemen linearer Gleichungen, *Bulletin de l'Academie Polonaise des Sciences et Lettres*, **A35**, 355-357.

[27] Lakshminarayanan, A. V., and Lent, A. (1979) Methods of least squares and SIRT in reconstruction, *J. Theor. Biol.*, **76**, pp. 267–295.

[28] Lent, A. (1999) *private communication*.

[29] Narayanan, M., Byrne, C., and King, M. (2000) An interior point iterative maximum likelihood reconstruction algorithm incorporating upper and lower bounds, with application to SPECT transmission imaging, *submitted for publication*.

[30] Tanabe, K. (1971) Projection method for solving a singular system of linear equations and its applications, *Numer. Math.*, **17**, pp. 203–214.