Accelerating The EMML Algorithm and Related Iterative Algorithms by Rescaled Block-Iterative Methods

Charles L. Byrne, *Member IEEE*

***Abstract*—Analysis of convergence of the algebraic reconstruction technique (ART) shows it to be predisposed to converge to a solution faster than simultaneous methods, such as those of the Cimmino–Landweber type, the expectation maximization maximum likelihood method for the Poisson model (EMML), and the simultaneous multiplicative ART (SMART), which use all the data at each step. Although choice of ordering of the data and of relaxation parameters are important, as Herman and Meyer have shown, they are not the full story. The analogous multiplicative ART (MART), which applies only to systems $y = Px$ in which $y > 0, P \geq 0$ and a nonnegative solution is sought, is also sequential (or "row-action"), rather than simultaneous, but does not generally exhibit the same accelerated convergence relative to its simultaneous version, SMART. By dividing each equation by the maximum of the corresponding row of P, we find that this rescaled MART (RMART) does converge faster, when solutions exist, significantly so in cases in which the row maxima are substantially less than one. Such cases arise frequently in tomography and when the columns of P have been normalized to have sum one.**

Between simultaneous methods, which use all the data at each step, and sequential (or row-action) methods, which use only a single data value at each step, there are the block-iterative (or ordered subset) methods, in which a single block or subset of the data is processed at each step. The ordered subset EM (OSEM) of Hudson *et al.* is significantly faster than the EMML, but often fails to converge. The "rescaled block-iterative" EMML (RBI-EMML) is an accelerated block-iterative version of EMML that converges, in the consistent case, to a solution, for any choice of subsets; it reduces to OSEM when the restrictive "subset balanced" condition holds. Rescaled block-iterative versions of SMART and MART also exhibit accelerated convergence.

***Index Terms*—Accelerated iterative image reconstruction, EM algorithm, tomography.**

I. Introduction and Background

In a recent article [1], Herman and Meyer present simulation reconstructions for positron emission tomography (PET) to support their claim that, by careful adjustment of the order in which the projection data is accessed and of the relaxation parameters in the algebraic reconstruction technique (ART), one can achieve an order-of-magnitude acceleration in convergence compared to the standard expectation-maximization approach to maximize likelihood (EMML), as it occurs in emission tomography [2]–[5]. Guan and Gordon discuss a projection access ordering for ART involving a multilevel scheme that, with a small number of iterations, gives results comparable to that achieved by filtered backprojection [6]. Simulation studies by the author on small scale problems showed that the ART algorithm is typically significantly faster than algorithms of the Cimmino–Landweber type (C–L) [7]–[11], the simultaneous multiplicative ART (SMART) [12]–[14], or the expectation maximization maximum likelihood method (EMML) based on the Poisson model. In this paper, we investigate why some algorithms should be faster than others and use our findings to develop accelerated versions of the EMML and SMART.

Manuscript received December 24, 1995; revised April 11, 1997. This work was supported in part by the National Cancer Institute, NIH, under Grants CA42165 and CA51071. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Andrew Yagle.

The author is with the Department of Mathematical Sciences, University of Massachusetts at Lowell, Lowell, MA 01854 USA (e-mail: byrnec@cs.uml.edu).

Publisher Item Identifier S 1057-7149(98)00309-1.

The ART algorithm is a sequential method, in which only a single equation is used at each step; Censor [15] calls such algorithms "row-action." The multiplicative ART (MART) [16] is also sequential. The C–L type methods are similar to ART, but use all the equations at each step; we refer to such methods as "simultaneous." Are sequential methods inherently faster than simultaneous ones? The MART is not always faster than its simultaneous version, SMART. The EMML and the simultaneous MART (SMART) are to MART as C–L type methods are to ART. All these methods involve the "project and average" format. Indeed, as we shall see, whereas ART and C-L methods use ordinary orthogonal projection onto hyperplanes, all the others just mentioned use a certain "Bregman projection," that is, a generalized projection based on a Bregman distance [17]–[19]. These analogies will help us discover accelerated versions of EMML and SMART.

Between sequential (or "row-action") methods and simultaneous ones are the so-called "block-iterative" methods [20]–[23], called "ordered subset" methods in [24]. For these methods, one partitions the data into disjoint subsets and then proceeds sequentially, using all the data in a single subset at each step. Hudson *et al.* [24] presented a block-iterative version of the EMML—which they called the "ordered subset EM" (OSEM)—that, in their experience, occasionally failed to converge, but more often converged much faster than the EMML. Hudson *et al.* were able to prove convergence only for an impractical special case, in which the subsets chosen corresponded to a restrictive "subset balanced" condition in the matrix; however, the OSEM seemed to perform well in a broader setting. Block-iterative versions of the SMART algorithm, such as the block-iterative MART of Censor and Segman [20] and the BI-SMART [21], are not generally faster than SMART. The block-iterative EMML (BI-EMML) in [21] is not generally faster than EMML.

There are two questions raised by these findings. 1) Why are some sequential or block-iterative methods faster than simultaneous ones, while others are not? 2) Is OSEM the proper block-iterative version of EMML, and if so, is there a proof of convergence for the general (consistent) case? As we discovered, the answers are as follows. 1) sequential and block-iterative methods provide the opportunity for block-dependent rescaling of the equations, which can lead to accelerated convergence. The ART and the OSEM turn out to be appropriately rescaled already, hence, the observed accelerated convergence. 2) the OSEM is not the correct block-iterative version of EMML. It turns out that the OSEM does not converge for the general consistent case; to have convergence without subset balance there is a second term required, which disappears in the subset balanced case. The correct version, called the rescaled block-iterative EMML (RBI-EMML), reduces to OSEM when there is "subset balance" and to EMML when there is but one subset. In general, however, the RBI-EMML involves a second term, missing in OSEM, and converges (in the consistent case) for any choice of subsets. When subset balance is missing, the OSEM is observed to have a limit cycle consisting of as many distinct images as there are subsets. How distinct these images are depends on how out of balance the subsets are. This limit cycle behavior occurs in all block-iterative methods when the system of equations is inconsistent due to noise; the problem with OSEM is that it produces this behavior even when there is no noise.

Except for the ART and C–L type methods, for which we consider general matrix equations of the form $Ax = b$, we shall restrict our study to linear systems of equations $y = Px$, where y and P are nonnegative and a nonnegative solution x is sought; we say that the system $y = Px$ is inconsistent if there is no such nonnegative solution. The matrix P is I by J, where typically both I and J are large. To simplify notation, we shall assume that the problem has been normalized so that the column sums of P are all equal to one. This amounts to redefining P and x: divide $P_{i,j}$ by the sum of the jth column and multiply x_j by the same sum.

We compare here the convergence of ART and EMML, and of the related sequential and simultaneous iterative algorithms just mentioned. We find that ART is predisposed to outperform simultaneous methods, even though care in choosing order and relaxation helps. We also find that the MART, while not generally as fast as ART, can be modified to match its order-of-magnitude acceleration compared to EMML. The modification, which involves rescaling the original equations, is then used to accelerate the other block-iterative methods as well.

We consider here only the so-called feasible case, in which the system of equations $Ax = b$ is consistent or $y = Px$ has a nonnegative solution; in the "infeasible" case, sequential and block-iterative methods cannot converge to a single vector. Instead, they approach a limit cycle of several vectors, typically one image for each data block. The "infeasible" case is considered in [25].

We begin our discussion with an overview of the algorithms and then take a more detailed look at the ART, the C–L type methods, EMML, MART, and SMART. We then discuss the block-iterative methods and acceleration. These block-iterative methods were introduced in [21] but without any derivation. The treatment here is mainly theoretical; reports of results on actual images have appeared in [26]–[28].

II. Overview: The Algorithms

A. The ART Algorithm

The ART algorithm [16] (or Kaczmarz's algorithm [29]) can be applied to any system of linear equations. The algorithm consists of projecting the current vector orthogonally onto the hyperplane corresponding to the next equation; when all the equations have been considered one begins again with the first one, and so on. We consider only a fixed (but random) ordering of the equations. For the M by N system of equations $Ax = b$, the projection $P_m(z)$ of vector z onto the mth hyperplane, denoted $B_m = \{x | Ax_m = b_m\}$, is given by

$$P_m(z)_n = z_n + A_{m,n}(b_m - Az_m) \Big/ \left(\sum_n A_{m,n}^2 \right) \tag{1}$$

where Σ_n denotes the sum over index n. For notational convenience, we assume that the original equations have been rescaled so that $\Sigma_n A_{m,n}^2 = 1$ for all m; then the denominator in (1) disappears.

B. The C–L Type Iteration Methods

In (unrelaxed) ART, we project orthogonally onto the next hyperplane at every step. The C–L type methods have us project orthogonally, with or without relaxation, onto each hyperplane simultaneously, and then average these vectors arithmetically to get the successor (see [7]–[11], [30], [31] and references there). The original Cimmino algorithm [30] used reflection in the hyperplane, rather than projection onto it; it is common, however, to refer to the Cimmino step from the current z to its successor z^* as that given by

$$z^* = z + cA^T(b - Az) \tag{2}$$

where A^T denotes the transpose of matrix A. If $c = 1/M$ then we are averaging the orthogonal projections; if $c = 2/M$ we are reflecting in each hyperplane. The algorithm in (2) is also called the Landweber algorithm [31]; convergence to a solution holds if $c \leq 2/\lambda_{\max}$, where $\lambda_{\max}$ is the largest eigenvalue of the matrix AA^T.

C. The EMML algorithm

The EMML algorithm, based on the model of independent Poisson random variables, as it occurs in emission tomography [2]–[5], is

$$x_j^{k+1} = x_j^k \sum_i P_{i,j} y_i / Px_i^k, \qquad k = 0, 1, \cdots \tag{3}$$

where $x_0 > 0, Px_i^k = (Px^k)_i$ denotes the ith entry of the vector Px^k, and Σ_i denotes summation over the index i.

For block-iterative methods, we shall assume that the set $\{i = 1, \cdots, I\}$ has been partitioned into disjoint subsets $S_1, \cdots, S_N$, and that Σ^n denotes summation over those i in the nth subset S_n.

D. The OSEM Algorithm

Starting with $x^0 > 0$ and using cyclic ordering $n = k(\text{mod } N) + 1$, the OSEM algorithm [24] is

$$x_j^{k+1} = x_j^k \left[\sum^n P_{i,j} y_i / P x_i^k \right] \Big/ \left[\sum^n P_{i,j} \right] \quad k = 0, 1, \cdots. \tag{4}$$

The matrix P is then called "subset balanced" if $\Sigma^n P_{i,j}$ depends only on j and not on n; it is a quite restrictive condition and no invertible P can have this property. Hudson *et al.* show that, provided $y = Px$ has nonnegative solutions and "subset balance" holds, (4) converges to a solution. In simulations, the convergence is often observed to be quite rapid, although one also observes divergence at times [28].

E. The MART Algorithm

The multiplicative ART (MART) algorithm [16] is as follows: beginning with $x^0 > 0$ and using cyclic ordering $i = k(\text{mod } I) + 1, k = 0, 1, \cdots$

$$x_j^{k+1} = x_j^k (y_i / P x_i^k)^{P_{i,j}}. \tag{5}$$

We can rewrite (5) as

$$x_j^{k+1} = [x_j^k]^{1-P_{i,j}} [x_j^k (y_i / P x_i^k)]^{P_{i,j}} \tag{6}$$

which shows the MART step to be a j-dependent geometric average of the current vector x^k and the scaling-projection of x^k onto the hyperplane Y_i, the set of all vectors x such that $y_i = Px_i$. The scaling-projection $B_i(x)$ of x onto Y_i, given by $B_i(x)_j = x_j(y_i/Px_i)$, plays an important role in helping us answer the second question posed earlier; when we begin to look for it in the various algorithms we find that it is everywhere. For example, the EMML (3) can be rewritten as follows:

$$\begin{aligned} x_j^{k+1} &= \sum_i P_{i,j} [x_j^k (y_i / P x_i^k)] \\ &= \sum_i P_{i,j} [B_i(x^k)_j], \qquad k = 0, 1, \cdots \end{aligned} \tag{7}$$

which shows the EMML step to be a j-dependent weighted arithmetic mean of all the scaling-projections of the current x^k.

F. The SMART Algorithm

The simultaneous MART algorithm (SMART) [12]–[14] is similar to the EMML, and is computed as follows: with $x^0 > 0$ let

$$x_j^{k+1} = x_j^k \exp \left[\sum_i P_{i,j} \log(y_i / P x_i^k) \right], \qquad k = 0, 1, \cdots \tag{8}$$

the SMART can also be rewritten to reveal the scaling-projections

$$\begin{aligned} x_j^{k+1} &= \exp \left[\sum_i P_{i,j} \log[x_j^k (y_i / P x_i^k)] \right] \\ &= \exp \left[\sum_i P_{i,j} \log(B_i(x^k)_j) \right], \qquad k = 0, 1, \cdots, \end{aligned} \tag{9}$$

which shows the SMART step to be a j-dependent weighted geometric mean of all the scaling-projections of the current x^k, with the same weights as used in the EMML.

G. Block-Iterative SMART (BI-SMART)

Between the MART and SMART, there are the block-iterative versions [20]–[24]. For our choice of subsets the BI-SMART is the following: beginning with $x^0 > 0$ and taking $n = k(\text{mod } N) + 1$, we have

$$x_j^{k+1} = x_j^k \exp \left[\sum^n P_{i,j} \log(y_i / P x_i^k) \right], \qquad k = 0, 1, \cdots. \tag{10}$$

Taking logs we have that

$$\begin{aligned} \log(x_j^{k+1}) &= \left(1 - \sum^n P_{i,j} \right) \log(x_j^k) \\ &\quad + \sum^n P_{i,j} \log(B_i(x^k)_j), \qquad k = 0, 1, \cdots. \end{aligned} \tag{11}$$

From (11), we see that the BI-SMART step is a j-dependent geometric mean of the current x^k and the scaling-projections of x^k, for all those i in the nth subset S_n. This is the key observation for answering the second question posed earlier. We are now in a position to extend the EMML to a block-iterative version.

H. Block-Iterative EMML (BI-EMML)

Recalling that the EMML and SMART differ in that the former uses a weighted arithmetic mean while the latter uses a weighted geometric mean, with the same weights, we are led to the BI-EMML algorithm by replacing the geometric means in BI-SMART (11) with arithmetic ones. The BI-EMML is then

$$x_j^{k+1} = \left(1 - \sum^n P_{i,j} \right) x_j^k + \sum^n P_{i,j} (B_i(x^k)_j), \quad k = 0, 1, \cdots \tag{12}$$

which can also be written as

$$x_j^{k+1} = \left(1 - \sum^n P_{i,j} \right) x_j^k + x_j^k \sum^n P_{i,j} (y_i / P x_i^k) \quad k = 0, 1, \cdots. \tag{13}$$

When each subset has only a single member, so $N = I$ and $S_n = \{n\}$, we get the EMML version of MART, which we call EMART: with $i = k (\text{mod } I) + 1$, we have EMART, as follows:

$$x_j^{k+1} = (1 - P_{i,j})x_j^k + x_j^k P_{i,j}(y_i/Px_i^k), \qquad k = 0, 1, \cdots. \tag{14}$$

It is clear that when there is but one subset, (12) reduces to (3). However, when subset balance holds, (12) does not reduce to OSEM in (4). To get the OSEM and thereby answer the second question posed above, it turns out that we must turn our attention briefly to the first question, concerning the acceleration.

The BI-SMART algorithm converges to a nonnegative solution of $y = Px$ whenever there are such solutions. The key step in proving this result is establishing the following lemma; the proof of convergence, given in the Appendix, is similar to that for the convergence of SMART in [12]–[14].

Lemma 1: For any $x \geq 0$ such that $y = Px$, we have Fejer monotonicity; that is,

$$KL(x, x^k) - KL(x, x^{k+1}) \geq \sum^n KL(y_i, Px_i^k) \geq 0 \tag{15}$$

where $\{x^k\}$ is the BI-SMART iterative sequence and KL is the nonnegative Kullback–Leibler distance [32] between nonnegative vectors: for nonnegative vectors $a = (a_m)$ and $b = (b_m)$ of arbitrary (but the same) dimension, we set $KL(a, b) = \Sigma_m a_m \log(a_m/b_m) + b_m - a_m$. Note that $KL(a, b)$ and $KL(b, a)$ are usually not the same and for any scalar $c > 0$ we have $KL(ca, cb) = cKL(a, b)$.

This lemma, hence the convergence of the algorithm, can be proven provided $\Sigma^n P_{i,j} \leq 1$; this is of course true here, since we have normalized so that the columns sum to one. But it also suggests that there might be an advantage in rescaling the equations.

I. The Rescaled BI-SMART Algorithm

Let $m_n = \max_j\{\Sigma^n P_{i,j}\}$ and for i in S_n divide the ith equation on both sides by m_n. The BI-SMART algorithm (10) then becomes the rescaled BI-SMART (RBI-SMART), as follows:

$$x_j^{k+1} = x_j^k \exp\left[m_n^{-1} \sum^n P_{i,j} \log(y_i/Px_i^k)\right]. \quad k = 0, 1, \cdots; \tag{16}$$

The inequality (15) now becomes

$$KL(x, x^k) - KL(x, x^{k+1}) \geq m_n^{-1} \sum^n KL(y_i, Px_i^k) \geq 0. \tag{17}$$

Especially for large-scale problems with a sizable N, the maximum m_n can be quite a bit smaller than one, so its reciprocal in (17) can be quite large, leading to rapid convergence.

J. The Rescaled BI-EMML Algorithm

The corresponding accelerated version of the BI-EMML is then the rescaled BI-EMML (RBI-EMML) given by

$$x_j^{k+1} = \left(1 - m_n^{-1} \sum^n P_{i,j}\right) x_j^k + m_n^{-1} \sum^n P_{i,j}(x_j^k y_i/Px_i^k), \qquad k = 0, 1, \cdots. \tag{18}$$

The inequality (17) also holds for (18). When there is subset balance, (18) reduces to the OSEM (4); we have answered the second question posed earlier using the answer to the first one.

K. Rescaled MART and EMART

When we apply the rescaling to the MART and the EMART algorithms, we see significantly accelerated convergence to a solution (in the consistent case). The rescaled versions of these algorithms are as follows. let $m_i = \max_j\{P_{i,j}\}$. Then, with $k = 0, 1, \cdots$, the rescaled MART is RMART, as follows:

$$x_j^{k+1} = x_j^k (y_i/Px_i^k)^{P_{i,j}/m_i} \tag{19}$$

and the rescaled EMART is REMART, as follows:

$$x_j^{k+1} = (1 - m_i^{-1} P_{i,j})x_j^k + x_j^k m_i^{-1} P_{i,j}(y_i/Px_i^k). \tag{20}$$

As we remarked earlier, the scaling-projection $B_i(x^k)$ of the current x^k occurs in all of the algorithms we have considered here, except for the ART and Cimmino. As a result, these algorithms can be put into the familiar framework of "project and average" methods, in which each step consists of some sort of projection onto all or some of the hyperplanes (or more general sets), followed by some sort of averaging, possibly including the current vector. Making the role of the scaling-projection explicit also serves to unify the various algorithms; for example, it revealed that the only structural difference between SMART and EMML is the use of the geometric or the arithmetic average. While the arithmetic average simplifies the computation somewhat, it complicates the theory; there are questions that remain open for EMML that have been answered for SMART. For example, we know that EMML converges to a solution in the consistent case, but we have no characterization of that solution. We would like to know how it depends on the starting vector x^0. The SMART converges to the nonnegative solution x minimizing $KL(x, x^0)$ [12].

If there is a nonnegative solution of $y = Px$, then the RBI-SMART converges to the unique solution x closest to the starting vector x^0, in the sense that $KL(x, x^0)$ is minimized. This happens regardless of the configuration of subsets we use. The RBI-EMML also converges to a solution; however, we cannot say which solution it is, and it also may vary with the choice of the subset configuration.

If no nonnegative solution of $y = Px$ exists, then all block-iterative methods are observed to converge to a limit cycle of I distinct vectors, rather than to a single vector. How distinct the I vectors are depends on the extent to which the equations $y = Px$ are (nonnegatively) inconsistent. In this case one can employ a "feedback" approach, using the vectors of the limit cycle to construct a "pseudodata" vector, which is then used as y and the algorithm restarted [25].

In the next few sections, we take a closer look at the details.

we call EMART: with $i = k(\text{mod } I) + 1$, we have EMART, as follows:

$$x_j^{k+1} = (1 - P_{i,j})x_j^k + x_j^k P_{i,j}(y_i/Px_i^k), \qquad k = 0, 1, \cdots. \tag{14}$$

It is clear that when there is but one subset, (12) reduces to (3). However, when subset balance holds, (12) does not reduce to OSEM in (4). To get the OSEM and thereby answer the second question posed above, it turns out that we must turn our attention briefly to the first question, concerning the acceleration.

The BI-SMART algorithm converges to a nonnegative solution of $y = Px$ whenever there are such solutions. The key step in proving this result is establishing the following lemma; the proof of convergence, given in the Appendix, is similar to that for the convergence of SMART in [12]–[14].

Lemma 1: For any $x \geq 0$ such that $y = Px$, we have Fejer monotonicity; that is,

$$KL(x, x^k) - KL(x, x^{k+1}) \geq \sum^n KL(y_i, Px_i^k) \geq 0 \tag{15}$$

where $\{x^k\}$ is the BI-SMART iterative sequence and KL is the nonnegative Kullback–Leibler distance [32] between nonnegative vectors: for nonnegative vectors $a = (a_m)$ and $b = (b_m)$ of arbitrary (but the same) dimension, we set $KL(a, b) = \Sigma_m a_m \log(a_m/b_m) + b_m - a_m$. Note that $KL(a, b)$ and $KL(b, a)$ are usually not the same and for any scalar $c > 0$ we have $KL(ca, cb) = cKL(a, b)$.

This lemma, hence the convergence of the algorithm, can be proven provided $\Sigma^n P_{i,j} \leq 1$; this is of course true here, since we have normalized so that the columns sum to one. But it also suggests that there might be an advantage in rescaling the equations.

I. The Rescaled BI-SMART Algorithm

Let $m_n = \max_j\{\Sigma^n P_{i,j}\}$ and for i in S_n divide the ith equation on both sides by m_n. The BI-SMART algorithm (10) then becomes the rescaled BI-SMART (RBI-SMART), as follows:

$$x_j^{k+1} = x_j^k \exp\left[m_n^{-1} \sum^n P_{i,j} \log(y_i/Px_i^k)\right]. \qquad k = 0, 1, \cdots; \tag{16}$$

The inequality (15) now becomes

$$KL(x, x^k) - KL(x, x^{k+1}) \geq m_n^{-1} \sum^n KL(y_i, Px_i^k) \geq 0. \tag{17}$$

Especially for large-scale problems with a sizable N, the maximum m_n can be quite a bit smaller than one, so its reciprocal in (17) can be quite large, leading to rapid convergence.

J. The Rescaled BI-EMML Algorithm

The corresponding accelerated version of the BI-EMML is then the rescaled BI-EMML (RBI-EMML) given by

$$x_j^{k+1} = \left(1 - m_n^{-1} \sum^n P_{i,j}\right) x_j^k + m_n^{-1} \sum^n P_{i,j}(x_j^k y_i/Px_i^k), \qquad k = 0, 1, \cdots. \tag{18}$$

The inequality (17) also holds for (18). When there is subset balance, (18) reduces to the OSEM (4); we have answered the second question posed earlier using the answer to the first one.

K. Rescaled MART and EMART

When we apply the rescaling to the MART and the EMART algorithms, we see significantly accelerated convergence to a solution (in the consistent case). The rescaled versions of these algorithms are as follows. let $m_i = \max_j\{P_{i,j}\}$. Then, with $k = 0, 1, \cdots$, the rescaled MART is RMART, as follows:

$$x_j^{k+1} = x_j^k (y_i/Px_i^k)^{P_{i,j}/m_i} \tag{19}$$

and the rescaled EMART is REMART, as follows:

$$x_j^{k+1} = (1 - m_i^{-1}P_{i,j})x_j^k + x_j^k m_i^{-1} P_{i,j}(y_i/Px_i^k). \tag{20}$$

As we remarked earlier, the scaling-projection $B_i(x^k)$ of the current x^k occurs in all of the algorithms we have considered here, except for the ART and Cimmino. As a result, these algorithms can be put into the familiar framework of "project and average" methods, in which each step consists of some sort of projection onto all or some of the hyperplanes (or more general sets), followed by some sort of averaging, possibly including the current vector. Making the role of the scaling-projection explicit also serves to unify the various algorithms; for example, it revealed that the only structural difference between SMART and EMML is the use of the geometric or the arithmetic average. While the arithmetic average simplifies the computation somewhat, it complicates the theory; there are questions that remain open for EMML that have been answered for SMART. For example, we know that EMML converges to a solution in the consistent case, but we have no characterization of that solution. We would like to know how it depends on the starting vector x^0. The SMART converges to the nonnegative solution x minimizing $KL(x, x^0)$ [12].

If there is a nonnegative solution of $y = Px$, then the RBI-SMART converges to the unique solution x closest to the starting vector x^0, in the sense that $KL(x, x^0)$ is minimized. This happens regardless of the configuration of subsets we use. The RBI-EMML also converges to a solution; however, we cannot say which solution it is, and it also may vary with the choice of the subset configuration.

If no nonnegative solution of $y = Px$ exists, then all block-iterative methods are observed to converge to a limit cycle of I distinct vectors, rather than to a single vector. How distinct the I vectors are depends on the extent to which the equations $y = Px$ are (nonnegatively) inconsistent. In this case one can employ a "feedback" approach, using the vectors of the limit cycle to construct a "pseudodata" vector, which is then used as y and the algorithm restarted [25].

In the next few sections, we take a closer look at the details.

III. The ART and C–L Type Algorithms

We can introduce relaxation in ART by taking the successor z' to z to be not $P_m(z)$ but the vector

$$z_n' = z_n + tA_{m,n}(b_m - Az_m) \tag{21}$$

where, for now, t is any quantity (possibly depending on m or on the actual iteration number).

We suppose now that we have a solution x, so that $Ax = b$. We consider the improvement in the distance to x achieved as we pass from z to z'. For convenience we use the Euclidean squared distance $L^2(a,b) = \Sigma_n(a_n - b_n)^2$, where $a = (a_n)$ and $b = (b_n)$ denote arbitrary vectors of any length (possibly scalars). Then a simple calculation shows that, for arbitrary but fixed m

$$L^2(x,z) - L^2(x,z') = t(2-t)L^2(b_m, Az_m). \tag{22}$$

In order for each step of relaxed ART to represent progress toward a solution, it is necessary that $0 \le t \le 2$. We see from (22) that the ordering of the equations plays a role also; the improvement $L^2(x,z) - L^2(x,z')$ is larger if the current vector z is not close to the hyperplane B_m; that is, it is helpful if successive hyperplanes are more orthogonal and not nearly identical. The maximum value of $t(2-t)$ occurs at $t = 1$, so, in this sense at least, ART is doing about as well as can be expected. Other choices for t do speed up convergence, as Herman and Meyer demonstrate, but knowing how to choose these parameters is an "art" in itself.

Our theme here is the natural advantages sequential methods (or "row action" methods) have over simultaneous ones. So let us consider simultaneous cousins of ART, the C–L type iterative methods, as in (2).

A straightforward calculation using (2) shows that the improvement we achieve with C–L is the C–L improvement.

A. C-L Improvement

$$L^2(x,z) - L^2(x,z^*) = (b-Az)^T(2cI - c^2AA^T)(b-Az) \tag{23}$$

where I is the identity matrix. From (23) we have that the C–L improvement satisfies the inequality

$$L^2(x,z) - L^2(x,z^*) \le (2c - c^2\lambda_{\max})L^2(b,Az). \tag{24}$$

In order for the improvement in (23) to be nonnegative it is sufficient that the matrix $2cI - c^2AA^T$ be positive-definite, which will happen if we choose c so that $2/c \ge \lambda_{\max}$, where $\lambda_{\max}$ denotes the largest eigenvalue of AA^T. Since we have rescaled the rows of A so that their norms are one, that is, the sums $\Sigma_n\ A_{m,n}^2 = (AA^T)_{m,m} = 1$, we have $M = \text{trace}(AA^T) =$ sum of the eigenvalues of AA^T, so $c = 1/M$ is acceptable.

The maximum of the quantity $(2c - c^2\lambda_{\max})$ occurs at $c = 1/\lambda_{\max}$. We consider then the two extreme cases: 1) $\lambda_{\max} = 1$; 2) $\lambda_{\max} = M$. In the first case the optimal c is $c = 1$; the C-L improvement is about equal to $L^2(b,Az)$, while the improvement obtained with the choice $c = 1/M$ is about equal to $(1/M)L^2(b,Az)$. In the second case, the optimal c is $c = 1/M$; the C-L improvement is about equal to $(1/M)L^2(b,Az)$. So if there is some eigenvalue spread in AA^T, then it can be exploited through careful choice of the c; see [31] and [32] for further treatment of the Landweber method, including discussion of the importance of the starting vector.

Let us compare the C–L improvement with $c = 1/M$ with that achieved by a complete pass through all the equations using the ART algorithm.

1) C–L Improvement with c = 1/M: $L^2(x,z) - L^2(x,z^*)$ about equal to $(1/M)L^2(b,Az)$ generally.

2) Full Cycle ART Improvement:

$$L^2(x,z) - L^2(x,z^*) = \sum_m L^2(b_m, Az_m^{m-1})$$

where $z^0 = z$, for $m = 1,2,\cdots,M, z_m$ is the successor of z^{m-1}, that is, $z^m = P_m(z^{m-1})$, and $z^* = z^M$.

We see clearly that the main advantage of ART over C-L lies in the absence of the $1/M$ term in the full cycle ART improvement. The C-L algorithm can be generalized to include inequalities [7] and accelerated by nonlinear reweighting of terms before averaging [8].

In [1], Herman and Meyer focus on the advantages of ART over the EMML algorithm. We therefore turn next to EMML and then to related multiplicative methods, the MART and its simultaneous version, SMART.

IV. The EMML Algorithm Compared to ART

The EMML, MART, and SMART algorithms apply to systems of equations with positivity constraints. To remind us of these constraints, we adopt new notation: let the system of equations be $y = Px$, where P is an I by J matrix with non-negative entries, y a positive vector and we seek a nonnegative solution x. In the EMML algorithm, the successor of vector z is z^*, whose jth entry is given by (3). This algorithm is the result of applying the general expectation maximization algorithm for likelihood maximization [34] to the specific model of Poisson emitters used in single photon emission computed tomography; for details and further references, see [2]–[5]. To compute improvement relative to the EMML step it is convenient to use the Kullback–Leibler distance defined above. Titterington [35] pointed out the hidden role of the KL distance in the EMML method and the convenience of this distance over the L^2 for analyzing algorithms such as EMML, MART, and SMART was demonstrated in [12]–[14].

Suppose now that there is a nonnegative x for which $y = Px$. Then, by the concavity of the log function the improvement in KL distance to x achieved by the EMML step can be shown to satisfy

$$KL(x,z) - KL(x,z^*) \ge KL(y,Pz). \tag{25}$$

Note that because of normalization we have $\Sigma_j\ z_j = \Sigma_i\ Pz_i$ and $\Sigma_j\ z_j^* = \Sigma_i\ y_i$; see the Appendix.

To continue the Herman and Meyer comparison between ART and EMML, we must relate KL to L^2; for a_n/b_n sufficiently close to 1 we can approximate $KL(a,b)$ by $\Sigma_n(a_n - b_n)^2/a_n$. Since we are looking to corroborate the order-of-magnitude improvement in convergence speed observed by Herman and Meyer, we shall make fairly rough approximations in our comparison. Since $\Sigma_j\,(P_{i,j})^2 \neq 1$ now the ART improvement becomes full cycle ART improvement.

A. Full Cycle ART Improvement

$$L^2(x,z) - L^2(x,z^*) = \sum_i \left[L^2(y_i, Pz_i^{i-1}) \Big/ \left(\sum_j (P_{i,j})^2 \right) \right] \tag{26}$$

where, as before, z^i is the ART successor of z^{i-1}. Using the Cauchy inequality we have

$$J = \sum_i \sum_j P_{i,j} \leq \left(\sum_i \sum_j (P_{i,j})^2 \right)^{1/2} \left(\sum_i \sum_j (1)^2 \right)^{1/2} = \left(\sum_i \sum_j (P_{i,j})^2 \right)^{1/2} (IJ)^{1/2}$$

so that $\Sigma_j\,(P_{i,j})^2$ is on the order of $1/I$, provided that I and J are about equal. So we have full cycle ART improvement, as follows:

$$L^2(x,z) - L^2(x,z^*) \text{ about equal to } I \sum_i L^2(y_i, Pz_i^{i-1}). \tag{27}$$

Since $y = Px$, we have $\Sigma_i\, y_i = \Sigma_i\, Px_i = \Sigma_j\, x_j$, so, again assuming I and J are about the same, we have that $\langle y \rangle$, the average value of y_i, and $\langle x \rangle$, the average value of x_j, are about equal. Using our approximation for the KL distance we have EMML improvement, as follows.

B. EMML Improvement

$$[L^2(x,z) - L^2(x,z^*)]/\langle x \rangle \text{ about equal to } L^2(y,Pz)/\langle y \rangle, \text{ or } L^2(x,z) - L^2(x,z^*) \text{ about equal to } L^2(y,Pz). \tag{28}$$

The advantage Herman and Meyer observed for ART comes, in part, from the presence of the factor I in (27) and its absence from (28).

We consider now the sequential MART algorithm and its simultaneous version, the SMART.

V. The MART and SMART Algorithms

The multiplicative ART (MART) algorithm introduced in [16] is analogous to ART but applies only to systems of equations satisfying the assumptions we have placed on $y = Px$. As in ART, only one of the I equations is used at each step: if z is the current vector and the ith equation is being used, then the successor to z is the vector z' given by (5).

The simultaneous MART (SMART) algorithm [12]–[14] is to MART as C–L is to ART. Note that for convergence of MART in the feasible case it is required only that $0 \leq P_{i,j} \leq 1$; see the Appendix. For current vector z the SMART successor z^* is given by (8).

The analogy between ART and C–L on the one hand and MART and SMART on the other is made clearer through the use of Censor's notion of Bregman projection onto hyperplanes [15]. For fixed vector z and fixed index i, if we minimize the distance $D(x,z)$, subject to $y_i = Px_i$, then we get the D-projection of z onto the hyperplane determined by the single equation $y_i = Px_i$. Generally, x cannot be written in closed form. However, if we take as our distance D the separable Bregman distance given by

$$D(x,z) = D_i(x,z) = \sum_j P_{i,j} KL(x_j, z_j) \tag{29}$$

we find that the jth component of the resulting D-projection (or Bregman projection) of z onto the hyperplane is given by

$$x_j = B_i(z)_j = z_j(y_i/Pz_i). \tag{30}$$

As was shown in Section II, we can then express MART, EMML, and SMART in terms of these Bregman projections, to reinforce the analogy with the additive ART and C–L methods.

The Bregman projections will play a more important role later, when we consider how to extend the block-iterative EMML beyond the restrictive "subset balanced" case.

We assume now that there is a nonnegative solution x, so $y = Px$. The improvement we achieve in MART as we pass from z to z', in the KL distance, is MART improvement, as follows.

A. MART Improvement

$$KL(x,z) - KL(x,z') = KL(y_i, Pz_i) + y_i - Pz_i + \sum_j z_j - \sum_j z'_j. \tag{31}$$

Lemma 3 shows that the right side of (31) is at least equal to $KL(y_i, Pz_i)$, hence nonnegative.

The improvement achieved in one SMART step, as we pass from z to z^* is SMART improvement.

B. SMART Improvement

$$KL(x,z) - KL(x,z^*) = KL(y,Pz) + \sum_i y_i - \sum_i Pz_i + \sum_j z_j - \sum_j z_j^*. \tag{32}$$

Lemma 4 shows that the right side of (32) is at least equal to $KL(y,Pz)$, hence nonnegative. The following Lemma 2 and its corollaries are easily established. Lemmas 3 and 4 are special cases of the convergence proof of BI-SMART given in the Appendix.

Lemma 2: Let a and b be any nonnegative vectors and let a_+ and b_+ denote the sums of the entries of a and b, respectively. Then we have $KL(a,b) = KL(a_+,b_+) + KL(a,(a_+/b_+)b)$, so that $KL(a,b) \geq KL(a_+,b_+)$.

Corollary 1: If $0 \leq P_{i,j} \leq 1$ for all i and j then, for any nonnegative vectors z and w and for any i, $KL(z,w) \geq KL(Pz_i, Pw_i)$.

Corollary 2: If P has been normalized so that each column sums to one then $KL(z,w) \geq KL(Pz,Pw)$.

Lemma 3: In the MART case, with index i chosen and with $0 \leq P_{i,j} \leq 1$, we have

$$y_i - Pz_i + \sum_j z_j - \sum_j z'_j = KL(z',z) - KL(Pz'i, Pz_i) + KL(Pz'i, y_i) \geq 0 \quad (33)$$

so the left side of (31) is nonnegative.

Lemma 4: In the SMART case, with the column sums equal to one, we have

$$\sum_i y_i - \sum_i Pz_i + \sum_j z_j - \sum_j z^*_j = KL(z^*,z) - KL(Pz^*,Pz) + KL(Pz^*,y) \geq 0 \quad (34)$$

so that the left side of (32) is nonnegative.

In the case of MART, if we set $z^0 = z$, and $z^i = (z^i - 1)'$, for $i = 1, \cdots I$, with the ith step involving the ith equation, then the MART cycle improvement after a complete pass through the data is the full cycle MART improvement (35).

C. Full Cycle MART Improvement

$$KL(x,z^0) - KL(x,z^I) \geq \sum_i KL(y_i, Pz_i^{i-1}). \quad (35)$$

We see, therefore, that the MART cycle improvement (35) does not appear much different than the improvement of SMART (32) or of EMML (25). There is a subtle difference, however. The MART calculations require only that the entries of P be in [0,1], while the SMART and EMML results are true only if we have first normalized so that the column sums are one. This suggests that it might be possible to accelerate the convergence of MART to a solution of $y = Px$ by increasing the exponential term in (5) without violating the condition that the matrix entries be in [0,1]. Specifically, for each i let $m_i = \max\{P_{i,j}, j = 1, \cdots, J\}$, and instead of (5) use the "rescaled" MART (RMART) given by (19): starting with z and using the ith equation, the RMART successor is given by (19); note that this is the usual MART algorithm applied to the rescaled system of equations in which the ith equation has been divided by m_i, so (35) becomes the full cycle RMART improvement (36).

D. Full Cycle RMART Improvement

$$KL(x,z^0) - KL(x,z^I) \geq \sum_i m_i^{-1} KL(y_i, Pz_i^{i-1}). \quad (36)$$

If the entries of P are already fairly close to one, or if at least one entry in each row is fairly close to one, then this modification buys very little. However, in many applications, such as single photon emission computed tomography (SPECT) image reconstruction with scattering and attenuation included in the P matrix, the individual entries of P can be quite small, so the acceleration due to the division in the exponential term can be significant.

Multiplying the exponential term of MART by a relaxation factor in (0,2) has been discussed Censor [15] among others but such large rescaling of the equations has not been considered for MART, probably because generally such rescaling cancels out, as in ART, or merely assigns new relative weights to the equations, as in SMART and EMML, without introducing an order-of-magnitude acceleration. But MART is peculiar in this regard. In simulations on small (20 by 20) systems of equations with random P matrices, we have observed quite dramatic improvements, comparable to those of ART. We shall report on full image reconstruction results in a follow-up paper.

Proof of convergence of MART in the feasible case is almost a direct consequence of the improvement inequality (35); we sketch the proof for the more general BI-SMART in the Appendix.

Proofs of convergence in general of EMML and SMART are given elsewhere (see [12–14]); proofs of convergence in the feasible case of the block-iterative methods discussed below are given in the Appendix.

We turn finally to block-iterative methods.

VI. Block-Iterative EMML and Smart

We consider now what Censor and Segman [20] call "block-iterative" methods (also termed "ordered subset" methods in [24]). We begin with a partition of the index set $\{i = 1, 2, \cdots, I\}$ into N disjoint subsets, $S_1, \cdots, S_N$. At each step of a block-iterative algorithm data from precisely one S_n is used; for a cyclically controlled block-iterative algorithm, in which each subset or block is visited in turn, a completed cycle consists of N such steps. Simultaneous methods can be accelerated by using parallel processing; as we have seen, in the absence of such techniques, sequential methods offer advantage. The block-iterative methods retain some of the parallelism of fully simultaneous methods while permitting speed-up through careful design and ordering of blocks. As we shall see, they can also be accelerated by equation rescaling, as we saw with MART.

Note that for convergence of BI-SMART (10) we only need that $\Sigma^n P_{i,j} \leq 1$; this suggests that we can accelerate (10) as we did MART. For each n we define scale factor $m_n = \max\{\Sigma^n P_{i,j}, j = 1, \cdots, J\}$. Then, after dividing the equations for i in S_n by m_n, the BI-SMART becomes the rescaled block-iterative SMART (RBI-SMART) given by (16).

It is not immediately obvious how to extend the EMML to a block-iterative version. The formulation given by Hudson *et al.* [24] is the OSEM in (4). They prove convergence in the feasible case, provided there is "subset balance," which means that for each n the sum $\Sigma^n P_{i,j}$ depends only on j, and not on n. This is a very restrictive condition (if a square P has it P must be noninvertible).

Failure of (4) to converge more generally in simulation studies has strongly suggested that (4) is not the correct extension for EMML. The correct formulation emerges once we rewrite the BI-SMART in terms of the Bregman projections, as in (11). Replacing the weighted geometric means in (11) with arithmetic ones we get the BI-EMML in (12).

As shown in the Appendix, proof of convergence of BI-SMART and BI-EMML to a solution in the feasible case is based on the improvement inequality below; here z' is the successor of z.

A. BI-SMART and BI-EMML Improvement

$$KL(x,z) - KL(x,z') \geq \sum^{n} KL(y_i, Pz_i). \quad (37)$$

This inequality follows for BI-SMART from the earlier lemmas and corollaries, just as the ones for MART and SMART improvement did. For BI-EMML it follows from the concavity of the log function. Rescaling for acceleration we find RBI-SMART and RBI-EMML improvement, as follows.

B. RBI-SMART and RBI-EMML Improvement

$$KL(x,z) - KL(x,z') \geq m_n^{-1} \sum^{n} KL(y_i, Pz_i). \quad (38)$$

These results hold for arbitrary choice of subsets.

VII. Normalization Can be Avoided

As we noted above, for convergence to a solution the MART and EMART require that $0 \leq P_{i,j} \leq 1$, while BI-SMART and BI-EMML require that for each n, $0 \leq \Sigma^n P_{i,j} \leq 1$. One way to guarantee that these conditions hold is to start by normalizing the problem. We redefine the matrix P by dividing each $P_{i,j}$ by the column sum $\Sigma_i P_{i,j}$, while redefining the unknowns x_j by multiplying x_j by the same column sum; the column sums of the new matrix are all one. In fact we have assumed, throughout the discussion above, that this normalization and redefinition had been performed. But once we rescale to achieve accelerated convergence we alter the P and y (but not the x), so the columns of the rescaled P no longer sum to one. We may well ask if it is allowable and even desirable to bypass the initial normalization and proceed directly to the rescaling. It is allowable and there are situations in which avoiding redefinition of the x_j is desirable. We give an example taken from transmission tomography to illustrate these points.

In transmission tomography, one has (ideally) line integrals of the attenuation density function along line segments through the region being scanned. The objective is to recover this density function from these integrals. Specifically, suppose that $x_j \geq 0$ is the attenuation density for pixel (or voxel) $j, L_{i,j}$ is the length of that portion of the ith line segment that lies within pixel j, a_i is the mean intensity of the radiation input along segment i, and b_i the mean intensity output from segment i. Then, according to the standard model, we have $b_i = a_i \exp(-\Sigma_j L_{i,j} x_j)$ for each i. Taking logs, we have $\Sigma_j L_{i,j} x_j = s_i = \log(a_i/b_i) \geq 0$ for each i. In some applications, with low dosage, the mean intensities b_i are not measured; only photon counts are available, from which the b_i must be estimated. We do not consider that here. With $L = [L_{i,j}]$ we then have to solve $Lx = s$ for $x \geq 0$.

Each of L, s and x are nonnegative, but the columns of L are otherwise unrestricted. Because of peculiarities in the scanning geometry, some of the pixels may not get as much attention as others; that is, the sums $\Sigma_i L_{i,j}$ may vary significantly with j. If we normalize to get P with column sums equal to one, we must replace each x_j with $x_j(\Sigma_i L_{i,j})$. If we then apply MART or BI-SMART, we minimize $KL(x, x^0)$ in the new variable domain, not in the original one. The effect is to pay more attention to those pixels j for which the sum $\Sigma_i L_{i,j}$ is larger. This uneven attention to pixels may not be desirable. To remain within the original domain we must avoid normalization. We shall consider how we might apply our accelerated methods directly to $s = Lx$. Application of these approaches is being considered by our group for medical transmission/emission tomography and in [36] for ionospheric tomography. In both cases, there are not as many segments as one might wish and some pixels receive less attention than others.

To apply RMART, set $m_i = \max\{L_{i,j} | j = 1, \cdots, J\}$ as before. We then replace each s_i by $y_i = s_i/m_i$, and each $L_{i,j}$ by $P_{i,j} = L_{i,j}/m_i$. Since $0 \leq P_{i,j} \leq 1$ the MART (5) applied to $y = Px$ then converges to the solution minimizing $KL(x, x^0)$, where the x are in the original domain. The EMART (14) applied to $y = Px$ also converges to a solution.

To apply RBI-SMART to $Lx = s$ set $m_n = \max\{\Sigma^n L_{i,j} | j = 1, \cdots, J\}$. We then let $y_i = s_i/m_n$ and $P_{i,j} = L_{i,j}/m_n$, for i in S_n. Then apply RBI-SMART (16) to $y = Px$; since for each j and n $\Sigma^n P_{i,j} \leq 1$, (16) will converge to the solution minimizing $KL(x, x^0)$, again with x in the original domain. The RBI-EMML (18) applied to $y = Px$ also converges to a solution.

We can even obtain simultaneous methods that avoid normalization, by taking only one subset, namely all of $\{1, 2, \cdots, I\}$, in the last paragraph. Then RBI-SMART (16) looks like SMART (8), even though we have not normalized. Once again, since $\Sigma_i P_{i,j} \leq 1$, (8) converges to the solution minimizing $KL(x, x^0)$, with x in the original domain. The RBI-EMML (18) in this case still has the two terms, since the sums $\Sigma_i P_{i,j}$ are not equal to one for all j.

There may be situations in which we want to minimize a weighted KL distance; we can do that using the methods just discussed. Suppose, for example, that we have weights $w_j > 0$ and we wish to find the solution of $y = Px$ for which the weighted cross-entropy, $\Sigma_j w_j KL(x_j, x_j^0)$ is minimized. We assume for now only that $P_{i,j} \geq 0$. Let $m = \min\{w_j | j = 1, \cdots, J\}$. Define the new matrix Q by $Q_{i,j} = (m/w_j) P_{i,j}$ and new unknowns $u_j = (w_j/m) x_j$. Then $y = Px$ if and only if $y = Qu$. Apply the RMART or RBI-SMART to $y = Qu$ just as we did to $s = Lx$ above. These algorithms then converge to minimum cross-entropy solutions in the (u_j) domain, which is the weighted (x_j) domain, as desired.

Optimization algorithms using distances more general than KL are given in [37]. In [38], Browne and De Pierro present a new algorithm called the *row-action maximum likelihood algorithm* (RAMLA), along with a block-iterative generalization. These algorithms are relaxed versions of the EMART and the BI-EMML. Because they are primarily interested in the inconsistent case, they introduce strong relaxation in order to avoid limit cycles. Our approach [25] is to go the other way and employ "feedback" methods based on convergence to limit cycles.

VIII. Conclusions

We have shown that one complete pass through the data with the sequential ART algorithm achieves an improvement in distance to a solution that is roughly an order-of-magnitude greater than that achieved by most simultaneous methods, such as Cimmino, EMML, and SMART. The multiplicative ART (MART) is also a sequential method, but does not generally exhibit such acceleration. We find, however, that by rescaling the system of equations so that the maximum matrix entry in each row is one, the rescaled MART (RMART) can be made to converge to a solution at about the same speed as ART. Further acceleration is achievable by subtle use of relaxation parameters and careful selection of the ordering of the equations, as Herman and Meyer noted for the ART. The block-iterative EMML suggested by Hudson *et al.*, the OSEM, is extended here to RBI-EMML, which gives accelerated convergence to a solution, in the consistent case, for any choice of subsets.

Appendix

A. Convergence of BI-SMART and BI-EMML

Proof of convergence of BI-SMART and BI-EMML to a solution in the feasible case is based on the improvement inequality (37) rewritten below: if $y = Px$ then for both BI-SMART and BI-EMML we have, with z' the successor of z

$$KL(x,z) - KL(x,z') \geq \sum^n KL(y_i, Pz_i). \tag{A1}$$

To prove (A1) for BI-EMML, we need only the concavity of the log function. For BI-SMART, we need to derive (A1) within a context of alternating minimization, as we did with SMART [12].

B. The BI-EMML Case

Using (12) with z in place of x_k and z' in place of x^{k+1} the left side of (A1) becomes

$$\begin{aligned} &KL(x,z) - KL(x,z') \\ &\quad = \sum_j x_j \log\left[\left(1 - \sum^n P_{i,j}\right) + \sum^n P_{i,j}(y_i/Pz_i)\right] \\ &\qquad + \sum_j (z_j - z'_j). \end{aligned} \tag{A2}$$

Since $(1 - \Sigma^n P_{i,j}) + \Sigma^n P_{i,j} = 1$, the concavity of the log function gives

$$\begin{aligned} &KL(x,z) - KL(x,z') \\ &\quad \geq \sum_j x_j \sum^n P_{i,j} \log(y_i/Pz_i) + \sum_j (z_j - z'_j) \\ &\quad = \sum^n [y_i \log(y_i/Pz_i) + Pz_i - y_i] \\ &\quad = \sum^n KL(y_i, Pz_i). \end{aligned} \tag{A3}$$

For BI-SMART inequality (A1) is harder to obtain.

C. The BI-SMART Case

To prove (A1) for BI-SMART we introduce the distance $G_n(x,z)$ given by

$$\begin{aligned} G_n(x,z) = KL(x,z) - \sum^n KL(Px_i, Pz_i) \\ + \sum^n KL(Px_i, y_i). \end{aligned} \tag{A4}$$

From Corollary 2 of Section II we know that $G_n(x,z) \geq 0$. The alternating minimization framework for BI-SMART is quite similar to that for SMART [12]–[14] . The following lemma gives the key identity.

Lemma A1: For all $x \geq 0$ and $z \geq 0$ we have

$$G_n(x,z) = G_n(z',z) + KL(x,z') \tag{A5}$$

where z' is the BI-SMART successor of z, as in (10). The proof is a simple calculation.

We now prove (A1) for BI-SMART. Notice that for all $x \geq 0$ we have $G_n(x,x) = \Sigma^n KL(Px_i, y_i)$. Now for $x \geq 0$ such that $y = Px$, we have, from (A5), that $KL(x,z) - KL(x,z') = \Sigma^n KL(y_i, Pz_i) + G_n(z',z)$, from which (A1) follows.

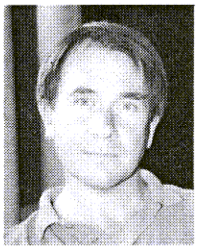
Suppose now that there is $x \geq 0$ with $y = Px$. According to (A1), the sequence $\{KL(x,z^k)\}$ is a decreasing sequence of positive numbers, where $\{z^k\}$ is the full sequence of iterates of either algorithm and the subset index n is related to k by $n = k(\bmod N) + 1$. Then, the sequence of differences $\{KL(x,z^k) - KL(x,z^{k+1})\}$ converges to zero; therefore, $\{\Sigma^n KL(y_i, Pz_i^k)\}$ converges to zero as k goes to infinity. The sequence $\{z^k\}$ is bounded, since each is closer to x than the previous one, so we can find a convergent subsequence, converging to cluster point z^*. The convergent subsequence may not involve each of the N subsets, but we do know that, for each $k, KL(x,z^*) \leq KL(x,z^k)$. We also know that for at least one n the subsequence $\{z^{mN+n-1} | m = 1,2,\cdots\}$ has z^* as a cluster point. Then $\{z^{mN+n} | m = 1,2,\cdots\}$ has the successor of $z^*, (z^*)'$ as a cluster point, where subset S_n is used to define the successor. It follows that, since both z^* and $(z^*)'$ are cluster points of $\{z^k\}$ and $\{KL(x,z^k)\}$ is decreasing, we have $KL(x,z^*) = KL(x,(z^*)')$; from (37) it follows that $z^* = (z^*)'$ and $y_i = Pz_i^*$ for all i in S_n. Therefore, $\{z^{mN+n} | m = 1,2,\cdots\}$ has z^* as a cluster point also. Therefore z^* is a cluster point for the subsequences of the form $\{z^{mN+n} | m = 1,2,\cdots\}$ for each n and $y = Pz^*$. Since we now know that z^* is a solution, it can be used as the x. So, $\{KL(z^*, z^k)\}$ is decreasing; but a subsequence actually goes to zero, so the whole sequence converges to zero. Therefore, z^k converges to solution z^*.

Acknowledgment

The author wishes to thank principal investigators M. King, University of Massachusetts Medical Center, Worcester, and B. Penney, now of the University of Chicago, for their continued support, and Y. Censor, University of Haifa, for many helpful discussions.

References

[1] G. Herman and L. Meyer, "Algebraic reconstruction techniques can be made computationally efficient," *IEEE Trans. Med. Imag.*, vol. 12, pp. 600–609, Sept. 1993.
[2] L. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imag.*, vol. MI-1, pp. 113–122, Oct. 1982.
[3] K. Lange and R. Carson, "EM reconstruction for emission and transmission tomography," *J. Comput. Assist. Tomog.*, vol. 8, pp. 306–312, 1984.
[4] Y. Vardi, L. Shepp, and L. Kaufman, "A statistical model for positron emission tomography," *J. Amer. Stat. Assoc.*, vol. 80, pp. 8–37, 1985.
[5] K. Lange, M. Bahn, and R. Little, "A theoretical study of some maximum likelihood algorithms for emission and transmission tomography," *IEEE Trans. Med. Imag.*, vol. MI-6, pp. 106–114, June 1987.
[6] H. Guan and R. Gordon, "A projection access order for speedy convergence of ART (algebraic reconstruction technique): A multilevel scheme for computed tomography," *Phys. Med. Biol.*, vol. 39, pp. 2005-2022, 1994.
[7] A. Iusem and A. DePierro, "Convergence results for an accelerated nonlinear Cimmino algorithm," *Numer. Math.*, vol. 49, pp. 367–378, 1986.
[8] R. Ansorge, "Connections between the Cimmino method and the Kaczmarz method for the solution of singular and regular systems of equations," *Computing*, vol. 33, pp. 367–375, 1986.
[9] M. Trummer, "SMART—An algorithm for reconstructing pictures from projections," *J. Appl. Math. Phys.*, vol. 34, pp. 746–753, Sept. 1983.
[10] A. Anderson and A. Kak, "Simultaneous algebraic reconstruction technique (SART): A new implementation of the ART algorithm," *Ultrason. Imag.*, vol. 6, pp. 81–94, Jan. 1984.
[11] P. Gilbert, "Iterative methods for three-dimensional reconstruction of an object from projections," *J. Theoret. Biol.*, vol. 36, pp. 105–117, 1972.
[12] C. Byrne, "Iterative image reconstruction algorithms based on cross-entropy minimization," *IEEE Trans. Image Processing*, vol. 2, pp. 96–103, Jan. 1993.
[13] ——, "Erratum and addendum to 'Iterative image reconstruction algorithms based on cross-entropy minimization,'" *IEEE Trans. Image Processing*, vol. 4, no. 2, pp. 225–226, 1995.
[14] ——, "Iterative reconstruction algorithms based on cross-entropy minimization," in *Image Models (And Their Speech Model Cousins)*, vol. 80, S. Levinson and L. Shepp, Eds. New York: Springer-Verlag, 1996.
[15] Y. Censor, "Row-action methods for huge and sparse systems and their applications," *SIAM Rev.*, vol. 23, pp. 444–466, Oct. 1981.
[16] R. Gordon, R. Bender, and G. Herman, "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography," *J. Theoret. Biol.*, vol. 29, pp. 471–481, 1970.
[17] L. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR Comput. Math. Math. Phys.*, vol 7, pp. 200–217, 1967.
[18] Y. Censor and A. Lent, "An iterative row-action method for interval convex programming," *J. Opt. Theory Appl.*, vol. 34, pp. 321–353, 1981.
[19] A. R. DePierro and A. N. Iusem, "A relaxed version of Bregman's method for convex programming," *J. Optimiz. Theory Appl.*, vol. 51, pp. 421–440, 1986.
[20] Y. Censor and J. Segman, "On block-iterative entropy maximization," *J. Inform. Optimiz. Sci.*, vol. 8, pp. 275–291, 1987.
[21] C. Byrne, "Block-iterative methods for image reconstruction from projections," *IEEE Trans. Image Processing*, vol. 5, pp. 792–794, 1996.
[22] P. Eggermont, G. Herman, and A. Lent, "Iterative algorithms for large partitioned linear systems, with applications to image reconstruction," *Lin. Alg. Appl.*, vol. 40, pp. 37–67, 1981.
[23] T. Elfving, "Block-iterative methods for consistent and inconsistent linear equations," *Numer. Math.*, vol. 35, pp. 1–12, 1980.
[24] H. Hudson and R. Larkin, "Accelerated image reconstruction using ordered subsets of projection data," *IEEE Trans. Med. Imag.*, vol. 13, pp. 601–609, Dec. 1994.
[25] C. Byrne, "Convergent block-iterative algorithms for image reconstruction from inconsistent data," *IEEE Trans. Image Processing*, vol. 6, pp. 1296–1304, Sept. 1997.
[26] D. Lalush, S. Karimi, and B. Tsui, "Convergence and resolution recovery of block-iterative EM algorithms modeling 3D detector response in SPECT," in *Conf. Rec., IEEE Medical Imaging Conf.*, Anaheim, CA, Nov. 1996, pp. 1618–1622.
[27] W. Xia, S. Glick, T.-S. Pan, E. Soares, and D.-S. Luo, "Region of interest evaluation of SPECT image reconstruction methods using a realistic brain phantom," in *Conf. Rec., IEEE Medical Imaging Conf.*, Anaheim, CA, Nov. 1996, pp. 1547–1548.
[28] C. Byrne, E. Soares, T.-S. Pan, S. Glick, and M. King, "Accelerating the EM algorithm using rescaled block-iterative methods," in *Conf. Rec., IEEE Medical Imaging Conf.*, Anaheim, CA, Nov. 1996, p. 1752–56.
[29] S. Kaczmarz, "Angenaeherte Aufloessung von Systemen linearen Gleichungen," *Bull. Acad. Polon. Sci. Lett.*, vol. A35, pp. 355–357, 1937.
[30] G. Cimmino, "Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari," *Ricerca Sci.*, Roma XVI, vol. 2, pp. 326–333, 1938.
[31] T. S. Pan and A. Yagle, "Acceleration of Landweber-type algorithms by suppression of projection on the maximum singular vector," *IEEE Trans. Med. Imag.*, vol. 11, pp. 479–487, Dec. 1992.
[32] T-S Pan, B. Tsui, and C. Byrne, "Choice of initial conditions in the ML reconstruction for transmission CT with truncated projection data," *IEEE Trans. Med. Imag.*, vol 16, pp. 426–438, Aug. 1997.
[33] S. Kullback and R. Leibler, "On information and sufficiency," *Ann. Math. Stat.*, vol. 22, pp. 79–86, 1951.
[34] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Stat. Soc. B*, vol. 39, pp. 1–38, 1977.
[35] D. Titterington, "On the iterative image space reconstruction algorithm for ECT," *IEEE Trans. Med. Imag.*, vol. MI-6, pp. 52–56, Mar. 1987.
[36] W. Kuklinski, "Ionospheric tomography via iterative cross-entropy minimization," *Radio Sci.*, vol. 32, pp. 1037–1049, May/June 1997.
[37] C. Byrne and Y. Censor, "Proximity function minimization and the convex feasibility problem for jointly convex Bregman distances," in preparation.
[38] J. Browne and A. DePierro, "A row-action alternative to the EM algorithm for maximizing likelihoods in emission tomography," *IEEE Trans. Med. Imag.*, vol. 15, pp. 687–699, Oct. 1996.



Charles L. Byrne (M'87) received the B.S. degree from Georgetown University, Washington, DC, in 1968 and the M.A. and Ph.D. degrees from the University of Pittsburgh, Pittsburgh, PA, in 1970 and 1972, respectively, all in mathematics.

From 1972 to 1986, he was a member of the Department of Mathematics, Catholic University of America, Washington, DC, serving as Department Chairman from 1983 to 1986. Since 1986, he has been with the Department of Mathematical Sciences, University of Massachusetts, Lowell, serving as Department Chairman from 1987 to 1990. He has served as consultant to the U.S. Naval Research Laboratory and to the Australian Department of Defense in the area of underwater acoustic signal processing. Since 1990, he also served as consultant in tomographic image reconstruction to the Department of Nuclear Medicine, University of Massachusetts Medical Center, Worcester.