58

Iterative projection onto convex sets using multiple Bregman distances

Charles Byrne

Department of Mathematical Sciences, University of Massachusetts Lowell, Lowell, MA 01854, USA

E-mail: byrnec@cs.uml.edu

Received 29 March 1999

Abstract. A number of inverse problems, in image reconstruction and elsewhere, can be formulated in terms of finding a vector in the intersection of certain convex sets that serve to constrain the solution. Finding such vectors is called the 'convex feasibility problem' (CFP). Algorithms to solve the CFP are usually iterative 'projection onto convex sets' (POCS) methods that employ orthogonal or more general projections onto the individual convex sets; Bregman's 'successive generalized projection' (SGP) is one such method. When the intersection of the convex sets is heterogeneous one may wish to optimize a certain function over that intersection; then we have a constrained optimization problem.

Generalized projections come from generalized distances, typically Bregman distances, chosen to incorporate prior information, such as non-negativity, about the image being reconstructed. Calculating a generalized projection onto a convex set can be simplified if the generalized distance can vary with the convex set. Censor and Elfving have discovered a simultaneous multiprojection algorithm that permits just such variation. Because simultaneous methods, which use all the convex sets at each step, can be slow to converge, there is considerable interest in faster, block-iterative methods that employ only some of the convex sets at each step. In this paper we present the first such method that permits multiple generalized projections.

We introduce a new notion of convex combination and apply it to obtain an extension of the SGP, called the 'multidistance SGP' (MSGP) method, that allows for projecton with respect to multiple generalized distances. We conclude with an extension of the MSGP to block-iterative algorithms involving relaxed generalized projections and paracontractive operators.

1. Introduction

A wide variety of inverse problems can be formulated in terms of finding a vector in the intersection of finitely many convex sets; in [18] Youla presents a mathematical theory of image restoration that highlights this formulation of the problem and lists a number of constraints typically encountered in image restoration that can be viewed as requiring that the image be a member of a certain convex set. He sketches the history of the use of the iterative 'projection onto convex sets' (POCS) method to solve this problem and also gives explicit description of the orthogonal projections associated with the convex sets in his list. The tutorial article by Combettes [10] gives a somewhat more up-to-date survey of this field, with greater emphasis on applications. In [1] Bauschke and Borwein provide a comprehensive theoretical discussion of the state of the art of POCS for Hilbert space (orthogonal) projections. All of these articles limit discussion to orthogonal projections. There is also a sizable literature on the use of generalized (or Bregman) projections in POCS: see Censor and Zenios [7].

For the convenience of the reader we review some examples of convex sets. These examples are all subsets of the infinite-dimensional complex Hilbert space $L^2(R)$. Analogous examples for the space R^J can easily be constructed.

Example 1. *Let Ω be a (measurable) subset of R. In the Hilbert space $L^2(R)$ the subset B_Ω consisting of those f in $L^2(R)$ whose Fourier transform, $\mathcal{F}(f)$, satisfies $\mathcal{F}(f)(\omega) = 0$, for ω not in Ω, is a closed convex set.*

Example 2. *Let T be a (measurable) subset of R. Those f in $L^2(R)$ such that $f(t) = 0$, for almost all $t \in T$, form a closed convex set.*

Example 3. *Let g be a fixed member of $L^2(R)$ and T be a subset of R. Those f in $L^2(R)$ such that $f(t) = g(t)$, for almost all t in T, form a closed convex set.*

Example 4. *Those f in $L^2(R)$ for which $\mathcal{F}(f)$ is non-negative almost everywhere in R form a closed convex set.*

Example 5. *Let h be a fixed member of $L^2(R)$ and let b be a fixed complex number. Let $\langle f, h\rangle$ denote the (complex) inner product between members of $L^2(R)$. The subset of $L^2(R)$ consisting of all f for which $\langle f, h\rangle = b$ is a closed convex set.*

Example 6. *For f in $L^2(R)$ let $\|f\|$ denote the norm of f. Let r be a fixed positive real number and let h be a fixed member of $L^2(R)$. The subset of all f for which $\|f - h\| \leqslant r$ is a closed convex set. If we select $h = 0$ then the subset consists of all f whose norm does not exceed r.*

Example 7. *Let g and h be fixed members of the real Hilbert space $L^2(R)$, with $g(t) \leqslant h(t)$, almost everywhere. Then the two subsets of $L^2(R)$ consisting of all f in $L^2(R)$ for which $g(t) \leqslant f(t)$ and $f(t) \leqslant h(t)$, respectively, for almost all t, form closed convex sets.*

The sets in examples 1 and 2 are closed subspaces of $L^2(R)$. The set in example 5 is a hyperplane. The set in example 3 is a closed linear manifold (that is, a translation of a closed subspace). The set in example 4 is a closed convex cone. The sets in example 6 are closed spheres in $L^2(R)$.

Let C_i, $i = 1, \ldots, I$, be closed convex subsets of a Hilbert space, with nonempty intersection C. The 'convex feasibility problem' (CFP) is to obtain a member of C. For example, the *bandlimited extrapolation problem* is to find a member f of $L^2(R)$ in the intersection of the closed convex sets described in examples 1 and 3 above. The problem of *power spectrum estimation* from finitely many estimates of the autocorrelation function is to find a member f of the convex set described in example 4, that is also in the intersection of a finite number of hyperplanes, as described in example 5. Regularization can be introduced into both of these problems by requiring, in addition, that f be contained within sets described in example 6; that is, we may impose norm constraints on f or require that f be within a fixed distance from some prior estimate of f. The problem of *image reconstruction from projections* is to find $f \geqslant 0$ in the intersection of finitely many hyperplanes, so this problem also fits into the framework of the CFP. Convex sets in R^J analogous to the intersection of those in example 7 will be used in section 6 of this paper.

In most applications C may be difficult to describe and so calculating directly an orthogonal or generalized projection onto C can be intractible, while calculating the projections onto the individual C_i can be simpler. The POCS methods are iterative algorithms whereby a member of C is obtained by repeated use of the projections onto the C_i.

As an example of a POCS method, consider the (unrelaxed) 'successive orthogonal projections' (SOP) method of Gubin *et al* [13]. Let C_i, $i = 1, \ldots, I$ be closed convex sets in Hilbert space X, having nonempty intersection C. For each $i = 1, \dots, I$ let P_i be the orthogonal projection onto C_i. For $k = 0, 1, \dots$ and $i = k(\text{mod } I) + 1$ define the iterative scheme by $x^{k+1} = P_i(x^k)$. Then the sequence $\{x^k\}$ converges weakly to a member of C. If the space X is finite-dimensional, then the convergence is strong. As we consider how this algorithm might be modified, we encounter several of the issues we shall deal with in this paper.

The SOP is a *sequential* method, in that only a single C_i is used at each step of the iteration. Other POCS methods are *simultaneous*, using all of the C_i at each step. More generally, there are the so-called *block-iterative* algorithms in which the C_i for i in a given subset (or block) of $\{i = 1, \dots, I\}$ are used at each step. The blocks can be fixed subsets of $\{i = 1, \dots, I\}$ or can vary as the iteration proceeds.

The SOP employs orthogonal projections onto the C_i. In [2] Bregman introduced what are now called *Bregman projections* with respect to generalized distances, such as cross entropy. His *successive generalized projection* (SGP) algorithm extends the SOP by replacing the orthogonal projections with Bregman projections. Bregman's main purpose in presenting this algorithm was to provide a method for optimizing convex functions subject to linear equality or inequality constraints; the function to be optimized is then employed in constructing the generalized distance.

The SOP algorithm as presented in [13] includes the possibility of *relaxation*. In the context of Hilbert space distance, relaxation is achieved through the use of ordinary convex combinations. Bregman's SGP method does not involve relaxation; indeed, the question of how to introduce relaxation in the context of generalized distances has been an issue for some time. In [6] it was shown that, by using the generalized distance to extend the notion of convex combination, one obtains a definiton of relaxation powerful enough to provide a basis for convergence proofs.

The Hilbert space (or Euclidean) distance used in SOP is a general purpose distance that makes computation relatively easy. Generalized distances present more difficulties computationally, but can often be tailored to the specific problem and designed to incorporate prior information about the vectors of interest. For example, when the vectors correspond to non-negative images, the use of cross entropy has been shown to be helpful. The next step down this road is to select the distances to match the particular C_i. Methods that employ multiple generalized distances are called *multiprojection* or *multidistance* methods in [8], where Censor and Elfving give a simultaneous multiprojection method for solving the CFP. If, on the other hand, we allow the distance used in Bregman's SGP method to define the generalized projections to vary from one step of the iteration to the next, then the iteration may not converge, as an example later will show. In this paper we present an extension of the SGP method that does permit multiple Bregman projections. Our algorithm is based on a notion of convex combination more general than that given in [6]. It has block-iterative versions and extends beyond Bregman projections to include more general operators.

Although most of these issues have been discussed in the context of Hilbert space or Banach space, most of the papers in the literature deal with the CFP within the space R^J, as we shall do here.

In the next section we survey some of the important results pertaining to the CFP and POCS. We then introduce a general notion of convex combination and show how it can be used to extend the notion of relaxation and to provide new iterative POCS methods for solving the CFP. As examples, we consider the 'multiplicative algebraic reconstruction technique' (MART) [12] for cross-entropy minimization, the 'split-feasibility problem' (SFP) and the problem of reconstructing an image subject to upper and lower bounds on the pixels. We close with an extension of these results to more general paracontractive operators.

2. Background

Perhaps the simplest example of POCS is the algorithm of Kaczmarz for solving the consistent system of linear equations $Ax = b$ [15]. With the rows of the I by J matrix A rescaled so that their Euclidean norms are equal to one, and using the notational convention $Ax_i = (Ax)_i = \sum_{j=1}^{J} A_{ij}x_j$, the orthogonal projection $P_i(z)$ of vector $z \in R^J$ onto the hyperplane $B_i = \{x|(Ax)_i = Ax_i = b_i\}$ is given by

$$(P_i(z))_j = z_j + A_{ij}(b_i - Az_i), \qquad j = 1, \ldots, J. \tag{2.1}$$

The Kaczmarz algorithm is as follows.

Algorithm 2.1 (Kaczmarz's algorithm). *Let $x^0 \in R^J$ be arbitrary. For $k = 0, 1, \ldots$ and $i = k(\text{mod } I) + 1$ let*

$$x_j^{k+1} = P_i(x^k)_j = x_j^k + A_{ij}(b_i - Ax_j^k). \tag{2.2}$$

We then have the following theorem.

Theorem 2.1. *If the linear system $Ax = b$ has solutions then the sequence $\{x^k\}$ obtained according to algorithm 2.1 converges to the solution of $Ax = b$ closest to x^0 in the Euclidean distance.*

A somewhat more general result, obtained by von Neumann in the 1930s [17] for the case of $I = 2$ and extended to general I by Halperin [14], applies when the C_i are closed linear subspaces in Hilbert space. Their result, extended by Youla [18] to the case in which the C_i are closed linear varieties, is the following.

Theorem 2.2. *Let C_i, $i = 1, \ldots, I$, be closed linear varieties in Hilbert space X, having nonempty intersection C. For each $i = 1, \ldots, I$ let P_i be the orthogonal projection onto C_i, $\lambda_i \in (0, 2)$ and $T_i(x) = x + \lambda_i(P_i(x) - x)$. For $k = 0, 1, \ldots$ and $i = k(\text{mod } I) + 1$ define the iterative scheme by $x^{k+1} = T_i(x^k)$. Then the sequence $\{x^k\}$ converges strongly to the orthogonal projection of x^0 onto C.*

The more general case, in which the C_i are closed convex sets in Hilbert space, was investigated by Gubin *et al* [13]; see also the discussion in Youla [18]. The theorem for this more general case is the following.

Theorem 2.3 (SOP). *Let C_i, $i = 1, \ldots, I$ be closed convex sets in Hilbert space X, having nonempty intersection C. For each $i = 1, \ldots, I$ let P_i be the orthogonal projection onto C_i, $\lambda_i \in (0, 2)$ and $T_i(x) = x + \lambda_i(P_i(x) - x)$. For $k = 0, 1, \ldots$ and $i = k(\text{mod } I) + 1$ define the iterative scheme by $x^{k+1} = T_i(x^k)$. Then the sequence $\{x^k\}$ converges weakly to a member of C.*

Note that when the C_i are no longer linear varieties, the limit need not be the orthogonal projection of x^0 onto C. Conditions sufficient to force strong convergence of the SOP iterative sequence are also presented in [13]; see also Bauschke and Borwein [1].

In his often cited paper [2] Bregman introduced a class of functions that have come to be called *Bregman functions* and used the associated *Bregman distances* to define generalized projections onto closed convex sets. The most commonly used Bregman distances are the Euclidean distance, associated with the Bregman function $f(x) = \sum_{j=1}^{J} x_j^2$, and the Kullback–Leibler distance between non-negative vectors, given by

$$KL(x, z) = \sum_{j=1}^{J} x_j \log\left(\frac{x_j}{z_j}\right) + z_j - x_j. \tag{2.3}$$

Bregman's *successive generalized projection* (SGP) method uses these generalized projections to solve the convex feasibility problem. See the appendix and the book by Censor and Zenios [7] for details concerning Bregman functions.

Let C_i, $i = 1, \ldots, I$ be closed convex sets in R^J with intersection C and suppose that $C \cap \overline{S} \neq \emptyset$. Denote by $P^f_{C_i}$ the Bregman projection operator associated with the Bregman function f and the convex set C_i (see the appendix for details). Bregman considers the following iterative procedure.

Algorithm 2.2 (Bregman's method of SGP). *Let $D_f(x, y)$ be the Bregman distance determined by the Bregman function f with zone S. Beginning with $x^0 \in S \subseteq R^J$, for $k = 0, 1, \ldots$, let $i = k(\mathrm{mod}\, I) + 1$ and*

$$x^{k+1} = P^f_{C_i}(x^k). \tag{2.4}$$

Bregman proves the following theorem [2].

Theorem 2.4. *If $x^k \in S$ for $k = 1, 2, \ldots$, the sequence $\{x^k\}$ given by (2.4) converges to a member of $C \cap \overline{S}$, whenever this set is nonempty.*

Algorithm 2.2 is a single-distance algorithm, in that the same Bregman function f is used in the definition of each of the generalized projections. For practical reasons, it would be helpful to have an algorithm that permits the use of multiple Bregman functions and projections. This would allow us to tailor the Bregman function to the individual convex set. In [8] Censor and Elfving presented such a multiprojection algorithm. Their 'simultaneous multiprojection' algorithm replaces the single Bregman function f with a family of Bregman functions $\{f_i, i = 1, \ldots, I\}$. Denote by $P^{f_i}_{C_i}$ the Bregman projection operator associated with the Bregman function f_i and the convex set C_i.

Algorithm 2.3 (the simultaneous multiprojections algorithm (SMA) of Censor and Elfving). Beginning with $x^0 \in S$, for $k = 0, 1, \ldots$, let x^{k+1} satisfy the vector equation

$$\sum_{i=1}^{I} \lambda_i \nabla f_i(x^{k+1}) = \sum_{i=1}^{I} \lambda_i \nabla f_i(P^{f_i}_{C_i}(x^k)), \tag{2.5}$$

where $\nabla f(x)$ denotes the gradient of the function f, evaluated at x and, for each $i = 1, \ldots, I$, $\lambda_i \in (0, 1)$, with $\sum_{i=1}^{I} \lambda_i = 1$.

Let S denote the intersection of the zones of the f_i. Censor and Elfving [8] prove the following result.

Theorem 2.5. *If $x^k \in S$ for $k = 1, 2, \ldots$, the sequence $\{x^k\}$ given by (2.5) converges to a member of $C \cap \overline{S}$, whenever this set is nonempty.*

Sequential and simultaneous algorithms are special cases of 'block-iterative' algorithms; for these one partitions the index set $\{i = 1, \ldots, I\}$ into N disjoint subsets (or blocks), B_n, $n = 1, \ldots, N$, and, at each step of the algorithm, only those i in a single B_n are used. One drawback of simultaneous algorithms is that they are often slow to converge; sequential and block-iterative methods have been used successfully to obtain accelerated convergence [3, 4]. In this paper we present a multiprojection block-iterative method for solving the CFP. Special cases of this method are algorithm 2.2, relaxed versions of algorithm 2.2, a multiprojection version of algorithms 2.2 and 2.3.

More general than Bregman projections onto convex sets are the Bregman paracontraction operators. In [6] Byrne *et al* introduce the notion of 'λ-relaxation' of operators with respect to a Bregman distance; subject to some mild restriction, the λ-relaxation of a Bregman projection is a Bregman paracontraction. They prove convergence, to a common asymptotic fixed point, of a block-iterative algorithm involving a family of operators that are paracontractive with respect to a single Bregman distance. The 'multiple distance' method we present in this paper extends this algorithm to include multiple Bregman distances.

Central to the development of our multiple distance methods is a general notion of convex combination, which is used to define the 'f, h-relaxation' of an operator with respect to an ordered pair of Bregman distances. This definition of relaxation is more general than the λ-relaxation used in [6]. We consider these topics next.

3. A general notion of convex combination

Let h and f be Bregman functions with zones S_h and S_f, respectively. Assume that their associated Bregman distances, D_h and D_f, satisfy the inequality $D_h(x, z) \geqslant D_f(x, z)$, for all $x \in \overline{S}_h \cap \overline{S}_f$ and $z \in S_h \cap S_f$. For example, let f and g be Bregman functions with associated distances D_f and D_g, respectively, and let $h = f + g$. However, the formulation above is much more general, since we do not assume that $D_h - D_f$ is again a Bregman distance; the difference function $h - f$ need not be convex and we can have $D_h(x, z) - D_f(x, z) = 0$ without having $x = z$.

For $x \in \overline{S}_h \cap \overline{S}_f$, $y \in S_f$ and $z \in S_h \cap S_f$, let $G(x; y, z, f, h)$ be the function of x defined as follows:

$$G(x; y, z, f, h) = D_f(x, y) + D_h(x, z) - D_f(x, z). \tag{3.1}$$

The next proposition provides a useful identity, which can be viewed as an analogue of Pythagoras' theorem. The proof is not difficult and we omit it.

Proposition 3.1. *If the non-negative function $G(x; y, z, f, h)$ has a unique minimizer $\hat{x} \in S_h \cap S_f$, for fixed y, z, f and h as above, then the gradient condition*

$$\nabla h(\hat{x}) = \nabla h(z) - \nabla f(z) + \nabla f(y) \tag{3.2}$$

holds and

$$G(x; y, z, f, h) = G(\hat{x}; , y, z, f, h) + D_h(x, \hat{x}). \tag{3.3}$$

When such an $\hat{x}$ exists, we say that $\hat{x}$ is the f, h-*generalized convex combination* of y and z and denote it $\hat{x} = C(y, z, f, h)$. This identity (3.3) is the key ingredient in the convergence proofs for the algorithms we present here.

Let T be any operator from $S_h \cap S_f$ into S_f. We then define the f, h-*relaxation of the operator* T to be the operator R given by $R(x) = C(T(x), x, f, h)$, as defined in the previous paragraph, whenever the latter exists. For $\lambda \in (0, 1)$ and $f = \lambda h$, the f, h-relaxation of T becomes the λ-relaxation of T, with respect to the function h, presented in [6]. We now use f, h-relaxation to derive multiprojection versions of several iteration algorithms.

4. Extending Bregman's SGP method to include multiple distances

We begin with a multiprojection successive generalized projection (MSGP) algorithm that extends algorithm 2.2, Bregman's SGP algorithm. Although this MSGP algorithm is a special case of the more general block-iterative algorithm we present later, we prove convergence of the MSGP here because the proof is simpler and serves to illustrate more clearly the role of f, h-relaxation. We then illustrate the MSGP algorithm with an example.

4.1. MSGP—the multiprojection extension of the SGP algorithm

Let f_i, $i = 1, \ldots, I$, be a family of Bregman functions and let h be a Bregman function that 'dominates' the family, that is, for which $D_h(x, z) \geqslant D_{f_i}(x, z)$, for all i, all $x \in \overline{S}_h \cap \overline{S}_{f_i}$ and all $z \in S_h \cap S_{f_i}$. Let $S = S_h \cap (\cap_{i=1}^{I} S_{f_i})$. The MSGP algorithm is the following.

Algorithm 4.1 (the MSGP algorithm). *For $k = 0, 1, \ldots$, and having calculated x^k, we obtain x^{k+1} as follows: with $i = k(\text{mod } I) + 1$, let $G^k(x) := G(x; P_{C_i}^{f_i}(x^k), x^k, f_i, h)$. We assume that $G^k(x)$ has a unique minimizer, which we take as x^{k+1}. We assume also that $x^{k+1} \in S_h$, so that*

$$\nabla h(x^{k+1}) = \nabla h(x^k) - \nabla f_i(x^k) + \nabla f_i(P_{C_i}^{f_i}(x^k)). \tag{4.1}$$

Finally, we assume that we have cyclic zone consistency*; that is, for each k, the vector x^{k+1} defined by (4.1) is in S_{f_m}, $m = (k + 1)(\text{mod } I) + 1$.*

We now have the following convergence theorem.

Theorem 4.1. *Let $C \cap \overline{S}$ be nonempty. Any sequence x^k obtained from the iterative scheme given by algorithm 4.1 converges to a member of $C \cap \overline{S}$.*

Proof. Let $\overline{x}$ be a member of $C \cap \overline{S}$. From the Pythagorean identity (3.3) it follows that

$$G^k(\overline{x}) = G^k(x^{k+1}) + D_h(\overline{x}, x^{k+1}). \tag{4.2}$$

At the same time, we have

$$G^k(\overline{x}) = D_h(\overline{x}, x^k) - D_{f_i}(\overline{x}, x^k) + D_{f_i}(\overline{x}, P_{C_i}^{f_i}(x^k)). \tag{4.3}$$

From Bregman's inequality we have that

$$D_{f_i}(\overline{x}, x^k) - D_{f_i}(\overline{x}, P_{C_i}^{f_i}(x^k)) \geqslant D_{f_i}(P_{C_i}^{f_i}(x^k), x^k). \tag{4.4}$$

Consequently, we know that

$$D_h(\overline{x}, x^k) - D_h(\overline{x}, x^{k+1}) \geqslant G^k(x^{k+1}) + D_{f_i}(P_{C_i}^{f_i}(x^k), x^k) \geqslant 0. \tag{4.5}$$

It follows that $\{D_h(\overline{x}, x^k)\}$ is decreasing and $\{D_{f_i}(P_{C_i}^{f_i}(x^k), x^k)\} \to 0$. Therefore, the sequence $\{x^k\}$ is bounded; let x^* be an arbitrary cluster point. From the fact that $\{D_{f_i}(P_{C_i}^{f_i}(x^k), x^k)\} \to 0$, it follows that x^* is a member of C. We can therefore use x^* as $\overline{x}$, to obtain that the sequence $\{D_h(x^*, x^k)\}$ is decreasing and so is converging to zero. From this, we conclude that $x^k \to x^*$. □

As a corollary, we obtain the convergence of Bregman's SGP algorithm.

Corollary 4.1. *Let f be a Bregman function with zone S. Let $C \cap \overline{S}$ be nonempty. For $k = 0, 1, \ldots$ and $i = k(\text{mod } I) + 1$, let $x^{k+1} = P_{C_i}^{f}(x^k)$. Then the sequence $\{x^k\}$ converges to a member of $C \cap \overline{S}$.*

Proof. In the theorem above, take $h = f = f_i$, for all i. □

Another special case of the MSGP algorithm is the 'λ-relaxed' SGP. For $\lambda \in (0, 1)$, $x^0 \in S_f$, $k = 0, 1, \ldots$, and $i = k(\text{mod } I) + 1$, let

$$\nabla f(x^{k+1}) = (1 - \lambda)\nabla f(x^k) + \lambda \nabla f(P_{C_i}^{f}(x^k)). \tag{4.6}$$

We then have the following result.

Corollary 4.2. *Let f be a Bregman function with zone S. Let $C \cap \overline{S}$ be nonempty. Then the sequence $\{x^k\}$ given by (4.6) converges to a member of $C \cap \overline{S}$.*

Proof. Let h be f and f be λf in the MSGP. □

When the Bregman functions involved are *separable*, a more general form of relaxation can be employed in the MSGP. Let g_j be scalar Bregman functions and $\lambda_{ij} \in (0, 1)$, with $\sum_{i=1}^{I} \lambda_{ij} = 1$, for all $j = 1, \ldots, J$. For $i = 1, \ldots, I$, let f_i be defined by

$$f_i(x) = \sum_{j=1}^{J} \lambda_{ij} g_j(x_j) \tag{4.7}$$

and let h be given by

$$h(x) = \sum_{i=1}^{I} f_i(x) = \sum_{j=1}^{J} g_j(x_j). \tag{4.8}$$

Then $h \geqslant f_i$ for all i and the MSGP applies.

For accelerated convergence, we can replace (4.7) with

$$f_i(x) = \sum_{j=1}^{J} (\lambda_{ij}/\lambda^i) g_j(x_j), \tag{4.9}$$

where $\lambda^i = \max_{j=1,\ldots,J}\{\lambda_{ij}\}$. We still have $h \geqslant f_i$ for all i and the MSGP can still be applied.

We illustrate these last two points with a discussion of the multiplicative algebraic reconstruction technique (MART) [12].

4.2. An example: accelerating the MART

To illustrate the MSGP algorithm we obtain the MART and the accelerated rescaled MART (REMART) [3] as special cases.

Consider the system of linear equations $y = Px$, in which P is an I by J matrix with non-negative entries, y an I-dimensional column vector with positive entries and a non-negative solution vector x is sought. Let $C_i = \{x|Px_i = y_i\}$, where $Px_i = (Px)_i$ denotes the ith entry of the vector Px. For $i = 1, \ldots, I$ let

$$f_i(x) = \sum_{j=1}^{J} P_{ij}(x_j \log x_j - x_j), \tag{4.10}$$

so that the associated Bregman distance is

$$D_{f_i}(x, z) = \sum_{j=1}^{J} P_{ij} \left(x_j \log \frac{x_j}{z_j} + z_j - x_j \right). \tag{4.11}$$

For any positive vector x the Bregman projection $P_{C_i}^{f_i}(x)$ can be written in closed form: we have

$$(P_{C_i}^{f_i}(x))_j = x_j(y_i/Px_i), \qquad j = 1, \ldots, J. \tag{4.12}$$

We see that the iterative scheme given by $x^{k+1} = P_{C_i}^{f_i}(x^k)$, $i = k(\text{mod } I) + 1$ cannot converge in this case, since each x^k is just a scalar multiple of the starting vector.

If we take $h(x) = \sum_{j=1}^{J}(x_j \log x_j - x_j)$, then $D_h(x, z) \geqslant c_i D_{f_i}(x, z)$ for all i, if $c_i \leqslant m_i^{-1}$, where $m_i = \max\{P_{ij}, j = 1, \ldots, J\}$. Applying the MSGP algorithm, with h as above, we get the MART: with $i = k(\text{mod } I) + 1$,

$$x_j^{k+1} = x_j^k (y_i/Px_i^k)^{P_{ij}}. \tag{4.13}$$

Applying the MSGP algorithm, with $m_i^{-1} f_i$ in place of f_i and h as above, we get the accelerated rescaled MART (REMART): with $i = k(\text{mod } I) + 1$,

$$x_j^{k+1} = x_j^k (y_i / P x_i^k)^{P_{ij}/m_i}. \tag{4.14}$$

4.3. A block-iterative version of MSGP

The MSGP algorithm can be extended to a block-iterative version. Let $\{B_n, n = 1, \dots, N\}$ be a partition of the index set $\{i = 1, \dots, I\}$. Let f_i, $i = 1, \dots, I$, be a family of Bregman functions. For each $n = 1, \dots, N$ denote by $\sum^n$ the summation over the indices i in the block B_n. Let h be a Bregman function for which $D_h(x, z) \geqslant \sum^n D_{f_i}(x, z)$, for all n, all $x \in \overline{S}_h \cap (\cap_{i \in B_n} \overline{S}_{f_i})$ and all $z \in S_h \cap (\cap_{i \in B_n} S_{f_i})$. Let $S = S_h \cap (\cap_{i=1}^I S_{f_i})$. The BIMSGP algorithm is the following.

Algorithm 4.2 (the block-iterative MSGP (BIMSGP) algorithm). *For $k = 0, 1, \dots,$ and having calculated x^k, we obtain x^{k+1} as follows: with $n = k(\text{mod } N) + 1$, let*

$$G_n^k(x) := D_h(x, x^k) - \sum\nolimits^n D_{f_i}(x, x^k) + \sum\nolimits^n D_{f_i}(x, P_{C_i}^{f_i}(x^k)).$$

We assume that $G_n^k(x)$ has a unique minimizer, which we take as x^{k+1}. We assume also that $x^{k+1} \in S_h$, so that

$$\nabla h(x^{k+1}) = \nabla h(x^k) - \sum\nolimits^n \nabla f_i(x^k) + \sum\nolimits^n \nabla f_i(P_{C_i}^{f_i}(x^k)). \tag{4.15}$$

Finally, we assume that we have block-cyclic zone consistency*; that is, for each k, the vector x^{k+1} defined by (4.15) is in S_{f_m}, for all $m \in B_n$, where $n = (k + 1)(\text{mod } N) + 1$.*

We have the following convergence theorem.

Theorem 4.2. *Let $C \cap \overline{S}$ be nonempty. Any sequence x^k obtained from the iterative scheme given by algorithm 4.2 converges to a member of $C \cap \overline{S}$.*

The BIMSGP algorithm is a special case of the algorithm presented in section 8, so the theorem above is a corollary to the convergence theorem proven there.

5. The SFP

Given closed convex sets C in R^N and Q in R^M and M by N full-rank matrix A, the SFP is to find x in C such that Ax is in Q, if such x exist. With $A^{-1}(Q) = \{x | Ax \in Q\}$, the SFP is to find a member of the intersection of C and $A^{-1}(Q)$, if there are any members. Formulated this way, the SFP becomes a special case of the CFP, in which we are to find a member of the nonempty intersection of finitely many closed convex sets. Iterative algorithms for solving the CFP, involving orthogonal or generalized projection onto the individual convex sets, can then be applied to solve the SFP. In typical applications, the sets C and Q are easy to describe and the orthogonal projections onto C and Q, denoted P_C and P_Q, respectively, are relatively simple to implement. In contrast, the orthogonal projections onto $A(C) = \{Ax | x \in C\}$ and $A^{-1}(Q)$ are not easily computed. We seek iterative methods that require only the orthogonal projections P_C and P_Q.

In [8] Censor and Elfving apply their *simultaneous multiprojections algorithm* (SMA) to the SFP, to obtain a iterative method that solves the SFP for the case in which $M = N$. Their method applies, with some modification, to the case in which $M < N$. It appears, however, that simultaneous methods such as the SMA will not work for the case in which $M > N$, and we need to use our sequential method, the MSGP.

Let G be an M by M positive-definite matrix with associated norm $\|z\|_G^2 = z^T G z$. For fixed x in R^N the G-projection of Ax onto the convex set $A(C)$ is the vector $P_{A(C)}^G(Ax) = A\hat{c}$ that minimizes the function $\|Ax - Ac\|_G^2 = (x-c)^T A^T G A(x-c)$ over all c in C. If there is G for which $A^T G A = I$, then it follows that $\hat{c}$ is the orthogonal projection of x onto C; therefore, we have $P_{A(C)}^G(Ax) = AP_C(x)$.

If $M = N$ then $G = (AA^T)^{-1}$ is a suitable choice; since A is invertible, we then have $P_{A(C)}^G(z) = AP_C(A^{-1}(z))$ for all z in R^M. Our iterative algorithms then involve P_Q and $P_{A(C)}^G$.

If $M < N$ there is no such matrix G. Moreover, the convex set $A(C)$ need not be closed. To obtain iterative algorithms for this case we augment A using the full-rank $N - M$ by N matrix B to obtain the N by N invertible matrix $T = [A^T\ B^T]^T$. Although it is not necessary, the choice of B such that $AB^T = 0$ is particularly helpful.

If $M > N$ the choice of $G = A(A^T A)^{-2} A^T$ gives $A^T G A = I$. With $A^+ = (A^T A)^{-1} A^T$ and $A(R^N)$ denoting the range of A, we have $A(C) = (A^+)^{-1}(C) \cap A(R^N)$, so $A(C)$ is now closed. Proceeding as above, we have $P_{A(C)}^G(Ax) = AP_C(x)$; however, for those z in R^M that are not in the range of A, we do not have a simple expression for $P_{A(C)}^G(z)$. If we are to employ P_Q and $P_{A(C)}^G$ iteratively, we must apply $P_{A(C)}^G$ only to vectors already in $A(R^N)$. This suggests that we formulate our problem as a special case of the CFP, involving the three closed convex sets $A(R^N)$, $A(C)$ and Q. Since we can only apply the projection $P_{A(C)}^G$ after we have applied the orthogonal projection onto the range of A, we see that we cannot use a simultaneous method, such as the SMA of Censor and Elfving; we need to use a sequential algorithm that permits the use of multiple distances, such as the MSGP algorithm. Using the MSGP, we obtain an iterative solution to the SFP for the case in which $M > N$, as well as iterative algorithms for the remaining cases that are somewhat simpler than those obtained by using the SMA.

The algorithm obtained by Censor and Elfving for the SFP for the case in which $M = N$ is the following.

Algorithm 5.1 (the split-feasibility algorithm of Censor and Elfving). *Let the matrix A be invertible. For $k = 0, 1, \ldots$, and having computed x^k, we obtain x^{k+1} as follows:*

$$x^{k+1} = (I + A^T A)^{-1}(P_C x^k + A^T P_Q A x^k). \tag{5.1}$$

In the consistent case the sequence $\{x^k\}$ converges to $x^\infty \in C$, such that $Ax^\infty \in Q$.

In order to apply algorithm 4.1 for the case in which $M = N$ we define $C_1 = \{x | Ax \in Q\}$, $C_2 = C$, $f_1(x) = \gamma x^T A^T A x$, and $h(x) = f_2(x) = x^T x$; here γ is a positive constant chosen so that

$$D_h(x,z) - D_{f_1}(x,z) = (x-z)^T(I - \gamma A^T A)(x-z) \geqslant 0, \tag{5.2}$$

for all x and z. Therefore, $\gamma \leqslant 1/\lambda_{max}(A^T A)$, where $\lambda_{max}(A^T A)$ denotes the largest eigenvalue of the matrix $A^T A$. If we normalize A so that each of its rows has Euclidean norm equal to one, then, since trace$(A^T A)$ = trace$(AA^T) = M$, it follows that we could choose $\gamma \leqslant 1/M$. The corresponding Bregman projections are

$$P_{C_1}^1(x) = (A^T A)^{-1} A^T P_Q A(x) \tag{5.3}$$

and

$$P_{C_2}^2(x) = P_C(x), \tag{5.4}$$

where P_Q and P_C denote the orthogonal projections onto Q and C, respectively.

Applying algorithm 4.1 we obtain the following algorithm.

Algorithm 5.2 (the multiprojection sequential split-feasibility algorithm). *Let $\gamma > 0$ be selected so that the matrix $I - \gamma A^T A$ is non-negative definite. For $k = 0, 1, \ldots,$ having obtained x^k, let*

$$x^{k+1} = P_C(I + \gamma A^T(P_Q - I)A)x^k. \tag{5.5}$$

Then, in the consistent case, the sequence $\{x^k\}$ given by (5.5) converges to an $x \in C$ for which $Ax \in Q$, for any choice of initial vector x^0.

Remark. It can be shown that the algorithm just given converges, as well, for the case in which $M < N$ and the matrix AA^T is invertible, even if the set $A(C)$ is not closed [5]. The basic idea is to augment A to obtain a square, invertible matrix $T = [A^T \ B^T]^T$, to formulate the SFP using convex sets C and $V = Q \times R^{N-M}$ and then to apply the algorithm for the square case.

For the case in which $M > N$, the algorithm based on the MSGP is formulated as a CFP in the space R^M and has the following iterative step:

$$w^{k+1} = P_Q A(I + \gamma(A^T A)^{-1}(P_C - I))(A^T A)^{-1} A^T w^k,$$

with $\gamma > 0$ chosen not greater than the smallest eigenvalue of $(A^T A)$ and $x^{k+1} = (A^T A)^{-1} A^T w^{k+1}$ converging to the desired solution. Again, see [5] for details. The case of $M > N$ is particularly interesting because it represents a situation in which the sequential nature of the MSGP plays a significant role. Because we have the simplification $P^G_{A(C)}(Ax) = AP_C(x)$, but do not have an expression for $P^G_{A(C)}(z)$ when z is not in the range of A, it becomes important that we apply the projection $P^G_{A(C)}$ after we have applied the orthogonal projection onto the range of A, not simultaneously with it, as the SMA requires.

6. Reconstruction algorithms with upper and lower bounds on the pixels

In this section we consider the problem of finding a solution vector x satisfying the system of linear equations $Ax = y$, where A is a matrix with non-negative entries, and the constraint that x be contained within a box in R^J, that is, $a_j \leqslant x_j \leqslant b_j$, for $j = 1, \ldots, J$, where $a = \{a_j\}$ and $b = \{b_j\}$ are prior lower and upper vector bounds on $x = \{x_j\}$. The ABSMART algorithm presented here is a special case of the BIMSGP discussed earlier; the ABEMML algorithm is an 'additive' version of ABSMART that generalizes the expectation maximization maximum likelihood (EMML) reconstruction method. The ABEMML and ABSMART algorithms converge to a solution of $y = Ax$ with $a \leqslant x \leqslant b$ and, in addition, the ABSMART algorithm minimizes the quantity $KL(x - a, x^0 - a) + KL(b - x, b - x^0)$ over these same x, provided $a < x^0 < b$ and there is a solution of $y = Ax$ with $a \leqslant x \leqslant b$. The negative of the quantity $KL(x - a, x^0 - a) + KL(b - x, b - x^0)$ is a generalization of the Fermi–Dirac generalized entropy, which is obtained by taking $a_j = 0$ and $b_j = 1$ for all $j = 1, \ldots, J$. We assume, for notational convenience, that the matrix A has column sums equal to one. Detailed proofs of convergence of these algorithms are found in [3].

We impose the constraint $a \leqslant x \leqslant b$ through the use of the distances

$$D_i^{ab}(x, z) = \sum_{j=1}^{J} A_{ij}[KL(x_j - a_j, z_j - a_j) + KL(b_j - x_j, b_j - z_j)]. \tag{6.1}$$

We find that calculating Bregman projections onto the sets $C_i = \{x | y_i = Ax_i\}$ using the distance D_i cannot be done in closed form, whereas we can calculate closed form projections onto the C_i using the distances

$$D_i^a(x, z) = \sum_{j=1}^{J} A_{ij} KL(x_j - a_j, z_j - a_j) \tag{6.2}$$

and

$$D_i^b(x, z) = \sum_{j=1}^{J} A_{ij} KL(b_j - x_j, b_j - z_j). \tag{6.3}$$

We obtain our algorithms by considering duplicates of each of the C_i and letting $D_i = D_i^a$, $i = 1, \ldots, I$, $D_i = D_{i-I}^b$, $i = I + 1, \ldots, 2I$.

For any partition $\{B_n, n = 1, \ldots, N\}$ of the set $\{i = 1, \ldots, I\}$ we obtain the related partition $\{B_n^*, n = 1, \ldots, N\}$ of $\{i = 1, \ldots, 2I\}$ by defining B_n^* to include the set B_n, as well as the value $i + I$, for each $i \in B_n$.

To obtain the ABSMART algorithm from the BIMSGP algorithm we let $D_h(x, z) = KL(b - x, b - z) + KL(x - a, z - a)$. The ABEMML is not a special case of BIMSGP.

6.1. The ABSMART algorithm

The ABSMART algorithm is the following: we assume that $Aa_i < y_i < Ab_i$ for all i and that $a_j < x_j^0 < b_j$ for all j. Then, for each $k = 0, 1, \ldots$ and $n = k(\text{mod } N) + 1$, we have

$$x_j^{k+1} = \alpha_j^k b_j + (1 - \alpha_j^k) a_j \tag{6.4}$$

with

$$\alpha_j^k = \frac{[c_j^k \prod^n (d_i^k)^{A_{ij}}]}{[1 + c_j^k \prod^n (d_i^k)^{A_{ij}}]}, \tag{6.5}$$

$$c_j^k = \frac{(x_j^k - a_j)}{(b_j - x_j^k)}, \tag{6.6}$$

and

$$d_i^k = \frac{(y_i - Aa_i)(Ab_i - Ax_i^k)}{(Ab_i - y_i)(Ax_i^k - Pa_i)}; \tag{6.7}$$

in (6.5) $\prod^n$ denotes the product over indices $i \in B_n$. All terms in (6.7) are positive. We see from (6.4) that each term of the iterative sequence $\{x_j^k\}$ is a convex combination of the a_j and b_j; the iteration proceeds until convergence to a convex combination for which $y = Ax$, if such exists. If there is no such solution of $y = Ax$ then the algorithm will converge to an approximate solution satisfying the constraints, as we shall see; specifically, the limit is the unique vector satisfying $a \leqslant x \leqslant b$ for which the function $KL(Ax - Aa, y - Aa) + KL(Ab - Ax, Ab - y)$ is minimized.

6.2. The ABEMML algorithm

The ABEMML algorithm is the following: we assume that $Aa_i < y_i < Ab_i$ for all i and that $a_j < x_j^0 < b_j$ for all j. Then, for each $k = 0, 1, \ldots$ and $n = k(\text{mod } N) + 1$, we have

$$x_j^{k+1} = \frac{\alpha_j^k b_j + \beta_j^k a_j}{d_j^k} \tag{6.8}$$

with

$$\alpha_j^k = (x_j^k - a_j)e_j^k, \tag{6.9}$$

$$\beta_j^k = (b_j - x_j^k)f_j^k, \tag{6.10}$$

$$e_j^k = \sum^n A_{ij}\left(\frac{y_i - Aa_i}{Ax_i^k - Aa_i}\right) + \left(1 - \sum^n A_{ij}\right) \tag{6.11}$$

$$f_j^k = \sum^n A_{ij}\left(\frac{Ab_i - y_i}{Ab_i - Ax_i^k}\right) + \left(1 - \sum^n A_{ij}\right) \tag{6.12}$$

and

$$d_j^k = \alpha_j^k + \beta_j^k. \tag{6.13}$$

We see from (6.8) that each term of the iterative sequence $\{x_j^k\}$ is a convex combination of the a_j and b_j; the iteration proceeds until convergence to a convex combination for which $y = Ax$, if such exists. If there is no such solution of $y = Ax$ then the algorithm will converge to an approximate solution satisfying the constraints; specifically, the limit is the unique vector satisfying $a \leqslant x \leqslant b$ for which the function $KL(y - Aa, Ax - Aa) + KL(Ab - y, Ab - Ax)$ is minimized.

By suitably rescaling the equations in the system $Ax = y$ we can accelerate convergence; see [3] for details.

7. Relaxed Bregman paracontractions

In this section we apply our notion of relaxation to Bregman paracontractions, special cases of which are unrelaxed and most λ-relaxed Bregman projections onto closed, convex sets. The next two definitions were given by Censor and Reich in [9].

Definition 7.1. *A point x^* is an asymptotic fixed point of an operator $T : S \subseteq R^J \to R^J$ if*

$$(x^*, x^*) \in \overline{G(T)}, \tag{7.1}$$

where $\overline{G(T)}$ denotes the closure of the graph of T, in the Euclidean topology.

Every fixed point of T is an asymptotic fixed point; if S is closed and T is continuous, the converse is also true. The set of all asymptotic fixed points of T will be denoted $\hat{F}(T)$. If T is the Bregman projection onto the closed convex set C, then $\hat{F}(T) = C \cap \overline{S}_f$, where S is the zone of the Bregman function f.

Definition 7.2. *An operator $T : S \subseteq R^J \to S$ with $\hat{F}(T)$ nonempty, is called a Bregman paracontraction, with respect to a Bregman function f with zone S, if the following hold:*

(i) for every $y \in \hat{F}(T)$,

$$D_f(y, T(x)) \leqslant D_f(y, x), \tag{7.2}$$

for every $x \in S$, and

(ii) if $\{x^k\} \subseteq S$ is a bounded sequence for which

$$\lim_{k\to\infty} (D_f(y, x^k) - D_f(y, T(x^k))) = 0, \tag{7.3}$$

for some $y \in \hat{F}(T)$, then

$$\lim_{k\to\infty} D_f(T(x^k), x^k) = 0. \tag{7.4}$$

For the special case of $f(x) = \frac{1}{2}]\|x\|^2$, and $S = \overline{S} = R^J$, a Bregman paracontraction is a paracontraction in the sense of Elsner *et al* [11]. We extend our notion of f, h-relaxation to Bregman paracontractions.

Definition 7.3. *Let T be a Bregman paracontraction with respect to the Bregman function f with zone S_f. Let $G(x; T(z), z, f, h)$ be as defined in (3.1). Then the f, h-relaxation of T is the operator R, defined for $z \in S_h \cap S_f$ by*

$$R(z) = \text{argmin}_{x \in \overline{S}_h \cap \overline{S}_f} G(x; T(z), z, f, h). \tag{7.5}$$

We assume that the function $G(x; T(z), z, f, h)$ has a unique minimizer $R(z)$ and that $R(z) \in S_h$. Then, from proposition 3.1, we have that, for any z in $S_h \cap S_f$,

$$G(x; T(z), z, f, h) = G(R(z); T(z), z, f, h) + D_h(x, R(z)). \tag{7.6}$$

This analogue of Pythagoras' theorem will be helpful in the proofs that follow. From (7.6) we obtain a useful identity:

$$D_h(x, z) - D_f(x, R(z)) = G(R(z); T(z), z, f, h) + D_f(x, z) - D_f(x, T(z)). \tag{7.7}$$

From the inequality $G(R(z); T(z), z, f, h) \leqslant G(T(z); T(z), z, f, h)$ we have

$$D_h(z, T(z)) - D_f(T(z), z) \geqslant D_h(R(z), z) - D_f(R(z), z) + D_f(R(z), T(z)). \tag{7.8}$$

From the inequality $G(R(z); T(z), z, f, h) \leqslant G(z; T(z), z, f, h)$ we have

$$D_f(z, T(z)) - D_f(R(z), T(z)) \geqslant D_h(R(z), z) - D_f(R(z), z). \tag{7.9}$$

We have the following result concerning the asymptotic fixed points of T and R.

Proposition 7.1. *The asymptotic fixed points of T are asymptotic fixed points of R.*

Proof. Let w be an asymptotic fixed point of T and let $x^k \to w$. From (7.7) we have

$$D_h(w, x^k) - D_h(w, R(x^k)) \geqslant D_f(w, x^k) - D_f(w, T(x^k)) + G(R(x^k); T(x^k), x^k, f, h). \tag{7.10}$$

The right-hand side above is non-negative, since T is a Bregman paracontraction with respect to f. So we have that

$$D_h(w, x^k) - D_h(w, R(x^k)) \geqslant 0. \tag{7.11}$$

Since $x^k \to w$, we have $D_h(w, x^k) \to 0$; therefore, $D_h(w, R(x^k)) \to 0$, from which we conclude that $R(x^k) \to w$. Therefore, w is an asymptotic fixed point of R. □

To show that the converse may not be true, in general, we consider the following example, taken from [6].

Example. *For real numbers $x \geqslant 0$ and $z > 0$ let $f(x) = x \log x - x$ and $D_f(x, z) = x \log \frac{x}{z} + z - x$. Let $C = \{1\}$ and $T = P_C^f$. For λ in $(0, 1]$ we have $R_\lambda(z) = z^{1-\lambda}$. If $x^k \to 0$ then $R(x^k) \to 0$, so 0 is an asymptotic fixed point of R_λ. But only the number 1 is an asymptotic fixed point of T. We also have*

$$D_f(0, z) - D_f(0, R_\lambda(z)) = z - z^{1-\lambda}, \tag{7.12}$$

which is negative for z and λ in $(0, 1)$; so R_λ is not a Bregman paracontraction.

8. A multidistance block-iterative algorithm for the CAFPP

The *common asymptotic fixed point problem (CAFPP)* is to find a common asymptotic fixed point for a given finite family of operators $\{T_i\}_{i=1}^I$, $T_i : S_i \subseteq R^J \to S_i$, $i = 1, \ldots, I$, each Bregman paracontracting with respect to a Bregman function f_i with zone S_i. We consider only the consistent case in which $\hat{F} \triangleq \cap_{i=1}^I \hat{F}(T_i)$ is nonempty.

For $i = 1, \ldots, I$ let T_i be a Bregman paracontraction, with respect to a Bregman function f_i with zone S_i, whose set of asymptotic fixed points is $\hat{F}(T_i)$. Let h be a Bregman function with zone S_h and associated Bregman distance $D_h(x, z)$. We assume that, for each $k \geqslant 0$ and $i = 1, \ldots, I$, we have weights λ_{ik} satisfying either $\lambda_{ik} \geqslant \epsilon > 0$, for an arbitrary fixed $\epsilon > 0$, or $\lambda_{ik} = 0$. In addition, we assume that, for each k, $\sum_{i=1}^I \lambda_{ik} = 1$. For each k let $I(k) = \{i \mid \lambda_{ik} > 0\}$ be the set of indices of the active weights and assume that each $i = 1, \ldots, I$ is in infinitely many sets $I(k)$. We also assume that, for each k,

$$D_h(x, z) \geqslant \sum_{i=1}^{I} \lambda_{ik} D_{f_i}(x, z) \tag{8.1}$$

holds, for all $z \in S_h \cap (\cap_{i=1}^I S_i)$ and $x \in \overline{S}_h \cap (\cap_{i=1}^I \overline{S}_i)$.

Algorithm 8.1 (multidistance block-iterative Bregman paracontractions algorithm).
Initialization: Let $x^0 \in S_h \cap (\cap_{i=1}^I S_i)$ *be arbitrary.*
Iterative step: $\nabla h(x^{k+1}) = \sum_{i=1}^I \lambda_{ik}[\nabla f_i(T_i(x^k)) - \nabla f_i(x^k)] + \nabla h(x^k)$.

The applicability of the algorithm depends on the ability to invert the gradient ∇h explicitly. If the Bregman function h is essentially smooth then ∇h is a one-to-one mapping with continuous inverse $(\nabla h)^{-1}$; see, e.g., Rockafellar [16]. We must also assume zone consistency; that is, all the gradients that appear in the iterative step are defined.

Our analysis shows that any sequence of iterates $\{x^k\}$ generated by this algorithm converges to a common asymptotic fixed point of the operators $\{T_i\}_{i=1}^I$, if such points exist.

Now we prove the convergence to a common asymptotic fixed point of the block-iterative scheme described above. We assume from now on that the set of common asymptotic fixed points is nonempty.

The block-iterative nature of the algorithm stems from the freedom to choose in each iteration a different active set $I(k)$. The blocks $\{T_i\}_{i \in I(k)}$ may vary in size and in composition; the weights λ_{ik} may also vary. For each $k = 0, 1, \ldots$ and each $x \in S$ let us define

$$G_k(x) = D_h(x, x^k) - \sum_{i=1}^{I} \lambda_{ik} D_{f_i}(x, x^k) + \sum_{i=1}^{I} \lambda_{ik} D_{f_i}(x, T_i(x^k)). \tag{8.2}$$

These functions will play important roles in what follows. The next lemma provides a 'Pythagorean-like' identity; the proof is a simple calculation.

Lemma 8.1. *For each* $x \in S$ *we have* $G_k(x) = G_k(x^{k+1}) + D_h(x, x^{k+1})$.

We have the following convergence theorem.

Theorem 8.1. *If the following assumptions hold:*

(i) for each $i = 1, \ldots, I$, $T_i : S_i \to S_i$ *is a Bregman paracontraction with respect to Bregman function* f_i *with zone* $S_i \subseteq R^J$;

(ii) for any $k \geqslant 0$ *and* $i = 1, 2, \ldots, I$, *the parameters* λ_{ik} *are either* $\lambda_{ik} = 0$ *or* $\lambda_{ik} \geqslant \epsilon > 0$ *for an arbitrary fixed* ϵ;

(iii) for all $k \geqslant 0$, $\sum_{i=1}^I \lambda_{ik} = 1$;

(iv) $\hat{F} \cap (\cap_{i=1}^{I} \overline{S}_i) \neq \emptyset$;
(v) each $i = 1, \ldots, I$ *appears in infinitely many sets* $I(k)$;
(vi) for each k,

$$D_h(x, z) \geqslant \sum_{i=1}^{I} \lambda_{ik} D_{f_i}(x, z) \tag{8.3}$$

holds, for all $z \in \cap_{i=1}^{I} S_i$ *and* $x \in \cap_{i=1}^{I} \overline{S}_i$, *then any sequence* $\{x^k\}_{k\geqslant 0}$ *generated by algorithm 8.1 converges to a point* $x^* \in \hat{F}$.

Proof. From lemma 8.1 it follows that for every $x \in S$ we have

$$D_h(x, x^k) - D_h(x, x^{k+1}) = \sum_{i=1}^{I} \lambda_{ik}(D_{f_i}(x, x^k) - D_{f_i}(x, T_i(x^k))) + G_k(x^{k+1}). \tag{8.4}$$

Selecting $\hat{x}$ in $\hat{F}$ and using $x = \hat{x}$ above we obtain

$$D_h(\hat{x}, x^k) - D_h(\hat{x}, x^{k+1}) = \sum_{i=1}^{I} \lambda_{ik}(D_{f_i}(\hat{x}, x^k) - D_{f_i}(\hat{x}, T_i(x^k))) \geqslant 0, \tag{8.5}$$

since $(D_{f_i}(\hat{x}, x^k) - D_{f_i}(\hat{x}, T_i(x^k))) \geqslant 0$ from definition 7.2. We conclude from this that the sequence $\{D_h(\hat{x}, x^k)\}$ is decreasing, and, by the non-negativity of D_h, that it has a limit. Therefore, the sequence $\{\lambda_{ik}(D_{f_i}(\hat{x}, x^k) - D_{f_i}(\hat{x}, T_i(x^k)))\}$ converges to zero, as $k \to \infty$, for each i.

We want to show that $\lim_{k\to\infty} x^k = x^\infty \in \hat{F}$. Since $\{D_h(\hat{x}, x^k)\}$ is decreasing, by the boundedness of the partial level sets of D_h (e.g., [7, p 31]), $\{x^k\}$ is also bounded (e.g., [7, p 55]) and therefore has cluster points. Once we show that every cluster point, say x^*, is in $\hat{F}$, then the existence of the limit $\lim_{k\to\infty} x^k = x^*$ will follow. This is so because if we suppose there were two cluster points $x^{*'} \neq x^*$, both in $\hat{F}$, with

$$\lim_{\substack{k\to\infty \\ k\in K_1}} x^k = x^* \qquad \text{and} \qquad \lim_{\substack{k\to\infty \\ k\in K_2}} x^k = x^{*'}, \tag{8.6}$$

for some $K_1 \subseteq N_0$, $K_2 \subseteq N_0$, $N_0 \triangleq \{0, 1, 2, \ldots\}$, then, by standard properties of the Bregman function h (e.g., [7, definition 2.2.1]) we have,

$$\lim_{\substack{k\to\infty \\ k\in K_1}} D_h(x^*, x^k) = 0 \qquad \text{and} \qquad \lim_{\substack{k\to\infty \\ k\in K_2}} D_h(x^{*'}, x^k) = 0. \tag{8.7}$$

Since $x^* \in \hat{F}$, we have

$$0 \leqslant D_h(x^*, x^{k+1}) \leqslant D_h(x^*, x^k), \tag{8.8}$$

for all $k \geqslant 0$, so that $\lim_{k\to\infty} D_h(x^*, x^k)$ exists, thus $\lim_{\substack{k\to\infty \\ k\in K_2}} D_h(x^*, x^k) = 0$. It is part of the definition of Bregman distances [7, definition 2.1.1(vi)] that we can conclude from this that $x^* = x^{*'}$. To finish the proof we now show that every cluster point of $\{x^k\}$ is in $\hat{F}$. The argument below is adapted from Bauschke and Borwein [1] and Censor and Reich [9] and is similar to the proof of a special case of this theorem in [6].

Let $\{x^{k_l}\}$ be a subsequence converging to a cluster point x^*. Without loss of generality, we may assume that, for each $l = 1, 2, \ldots,$

$$I(k_l) \cup I(k_l + 1) \cup \cdots \cup I(k_{l+1} - 1) = \{1, 2, \ldots, I\}; \tag{8.9}$$

this can be done since, for each i, $i \in I(k)$ for infinitely many values of k.

If x^* is not in $\hat{F}$, then we can consider the sets $I_{in} \triangleq \{i | x^* \in \hat{F}(T_i)\}$ and $I_{out} \triangleq \{i | x^* \notin \hat{F}(T_i)\}$; let $m_l \in \{k_l, \ldots, k_{l+1} - 1\}$ be minimal with respect to the condition that $I(m_l) \cap I_{out} \neq \emptyset$. We know that $\{x^{k_l}\} \to 0$; we shall show that $\{x^{m_l}\} \to 0$. If $k_l < m_l$, we have $I(r) \subseteq I_{in}$ for $r = k_l, \ldots, m_l - 1$.

Since $x^* \in \cap_{i \in I_{in}} \hat{F}(T_i)$ we have that

$$D_h(x^*, x^{k_l}) \geqslant D_h(x^*, x^{k_l+1}) \geqslant \cdots \geqslant D_h(x^*, x^{m_l}).$$

Now, since $I(m_l) \cap I_{out} \neq \emptyset$ for every l, some i occurs infinitely often; without loss of generality we assume that we have an i such that $i \in I(m_l) \cap I_{out}$ for all l. We have shown above that $\{\lambda_{ik}(D_{f_i}(\hat{x}, x^k) - D_{f_i}(\hat{x}, T_i(x^k)))\} \to 0$, as $k \to +\infty$, so that, as $k \to +\infty$, $\{(D_{f_i}(\hat{x}, x^k) - D_{f_i}(\hat{x}, T_i(x^k))\} \to 0$. From definitions 7.1 and 7.2 we know that $\{D_{f_i}(T_i(x^{m_l}), x^{m_l})\} \to 0$, as $l \to +\infty$. We know also that $\lim_{l \to \infty} x^{k_l} = x^*$ so we also have that $\lim_{l \to \infty} x^{m_l} = x^*$. Therefore $\{T_i(x^{m_l})\} \to x^*$, which says that x^* is in $\hat{F}(T_i)$, a contradiction. We must conclude that $x^* \in \hat{F}$. □

With particular choices of the λ_{ik}, the T_i, the f_i and h, we obtain several of the algorithms previously discussed as special cases of this general algorithm.

If $T_i = P_{C_i}^{f_i}$, $i = 1, \ldots, I$ and, for each $k = 0, 1, \ldots,$ we take $\lambda_{ik} = 1$, for $i = k(\text{mod } I)+1$ and zero otherwise, we obtain the MSGP algorithm (algorithm 4.1).

If, for $i = 1, \ldots, I$ and $k = 0, 1, \ldots,$ we take $T_i = P_{C_i}^{f_i}$, $\lambda_{ik} = \lambda_i \in (0, 1)$ and $h(x) = \sum_{i=1}^{I} \lambda_i f_i(x)$, then we get the simultaneous multiprojection algorithm of Censor and Elfving (algorithm 2.3).

If, for $i = 1, \ldots, I$, we take $f_i = f$ and let T_i be paracontractive with respect to f, with T_I the identity operator, then we have the block-iterative relaxed paracontractions algorithm presented in [6].

We close with a summary of the main results in this paper.

9. Summary

In simplest terms, inverse problems involve the determination of the input into a system, based on limited and noisy measurements of the output. The input is viewed as a vector in an ambient space (often a Hilbert or Banach space) and the system is described as an operator between that space and the space containing the data. Typically, the system itself is not known precisely and must be modelled. Inversion schemes that do not anticipate sensitivity to noise can produce unacceptable results. Improved inversion methods can be obtained by reformulating the problem to reduce sensitivity to noise, that is, by regularization, and by incorporating prior information about the solution being sought. These improvements are typically achieved through the imposition of certain constraints on the solution. As we have seen, constraints often take the form of requiring that the desired solution reside within certain closed convex subsets of the ambient space, permitting us to reformulate the original inverse problem as a CFP. Our main topic has been the development of iterative algorithms for solving the CFP.

Constraints on the solution can sometimes be imposed through the use of generalized distances, such as Bregman distances, that are defined only for vectors that satisfy the constraints. When we perform a generalized projection onto a closed convex set using such generalized distances, we obtain a vector that satisfies both the constraint associated with the convex set and the constraint imposed by the generalized distance.

We have found that the generalized projection onto a closed convex set can sometimes be expressed in closed form, if the generalized distance can be tailored to the convex set involved.

This leads us to consider iterative algorithms in which multiple generalized distances are allowed.

Censor and Elfving have discovered a simultaneous iterative algorithm that permits the use of multiple generalized distances. However, simultaneous methods can be slow to converge, prompting us to investigate the possibility of sequential or block-iterative algorithms that employ multiple generalized distances. We know that algorithms such as Bregman's SGP can fail to converge if multiple generalized distances are used; to obtain the algorithms we seek, something further is needed. The key idea is relaxation through the use of a generalized form of convex combination, that which we have called f, h-relaxation.

Using this idea, we have obtained the MSGP algorithm and its block-iterative versions. We have extended these algorithms to Bregman paracontractions, operators more general than Bregman projections. We have applied these iterative algorithms to several problems, including the SFP, the common asymptotic fixed point problem for Bregman paracontractions and the problem of reconstructing an image from linear projections, using upper and lower bounds on the entries of the solution.

Appendix. Bregman functions and Bregman projections

Let $f : \Lambda \subseteq R^J \to R$, let S be a nonempty open convex set whose closure, $\overline{S}$, is contained in Λ. For $x \in \overline{S}$ and $z \in S$ define

$$D_f(x, z) = f(x) - f(z) - \nabla f(z)^T(x - z), \tag{A.1}$$

where $\nabla f(z)$ denotes the gradient of f, evaluated at the point z. For $r \in R$ define the level sets as follows: for fixed $y \in S$ let

$$L_1^f(y, r) = \{x \in \overline{S} | D_f(x, y) \leqslant r\}; \tag{A.2}$$

and for fixed x in $\overline{S}$ let

$$L_2^f(x, r) = \{y \in S | D_f(x, y) \leqslant r\}. \tag{A.3}$$

Following Censor and Zenios [7] we say that f is a *Bregman function* with associated *Bregman distance* D_f, and write $f \in \mathcal{B}(S)$ provided the following conditions hold:

(1) f has continuous first partial derivatives for all $x \in S$;
(2) f is strictly convex on $\overline{S}$;
(3) f is continuous on $\overline{S}$;
(4) for every $r \in R$, the level sets $L_1^f(y, r)$ and $L_2^f(x, r)$ are bounded, for all $x \in \overline{S}$ and $y \in S$;
(5) if $\{y^n, n = 1, 2, \ldots\} \subseteq S$ and $y^n \to y^*$, then $D_f(y^*, y^n) \to 0$;
(6) if $\{x^n, n = 1, 2, \ldots\} \subseteq \overline{S}$ is a bounded sequence, $\{y^n, n = 1, 2, \ldots\} \subseteq S$, $\{y^n\} \to y^*$, and $D_f(x^n, y^n) \to 0$, then $\{x^n\} \to y^*$.

We then have the following result.

Lemma A.1. *Let $f \in \mathcal{B}(S)$. Let C be a nonempty closed convex subset of R^J such that $C \cap \overline{S} \neq \emptyset$. Then, for all $y \in S$, there is a unique vector, denoted $P_C^f(y)$ and called the Bregman projection of y onto C, that minimizes the function $D_f(c, y)$ over all $c \in C \cap \overline{S}$.*

As is usually done, we restrict discussion to those situations in which this generalized projection operator is *zone consistent*, that is, $P_C^f(y) \in S$ always holds. With this assumption, we have *Bregman's inequality*: for all $c \in C \cap \overline{S}$ and $y \in S$,

$$D_f(c, y) \geqslant D_f(c, P_C^f(y)) + D_f(P_C^f(y), y). \tag{A.4}$$

From the Bregman inequality we obtain the following characterization of $P_C^f(y)$: for every $y \in S$, $u = P_C^f(y)$ is the unique vector for which the inequality

$$\langle c - u, \nabla f(y) - \nabla f(u)\rangle \leqslant 0 \qquad \text{(A.5)}$$

holds for all $c \in C \cap \overline{S}$.

Acknowledgments

The author wishes to acknowledge helpful conversations with Professors Gabor T Herman of the University of Pennsylvania, Yair Censor of the University of Haifa, Israel and Tommy Elfving of the University of Linköping, Sweden.

References

[1] Bauschke H H and Borwein J M 1996 On projection algorithms for solving convex feasibility problems *SIAM Rev.* **38** 367–426
[2] Bregman L M 1967 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.* **7** 200–17
[3] Byrne C L 1998 Accelerating the EMML algorithm and related iterative algorithms by rescaled block-iterative (RBI) methods *IEEE Trans. Image Process.* **7** 100–9
[4] Byrne C L 1998 Iterative algorithms for deblurring and deconvolution with constraints *Inverse Problems* **14** 1455–67
[5] Byrne C L 1999 Iterative algorithms for the split-feasibility problem, submitted
[6] Byrne C, Censor Y and Herman G T 1998 Convergence of block-iterative algorithms with relaxed Bregman paracontractions, submitted for publication
[7] Censor Y and Zenios S A 1997 *Parallel Optimization: Theory, Algorithms and Applications* (New York: Oxford University Press)
[8] Censor Y and Elfving T 1994 A multiprojection algorithm using Bregman projections in a product space *Numer. Algorithms* **8** 221–39
[9] Censor Y and Reich S 1996 Iterations of paracontractions and firmly nonexpansive operators with applications to feasibility and optimization *Optimization* **37** 323–39
[10] Combettes P L 1993 The foundations of set theoretic estimation *Proc. IEEE* **81** 182–208
[11] Elsner L, Koltracht I and Neumann M 1992 Convergence of sequential asynchronous nonlinear paracontractions *Num. Math.* **62** 305–19
[12] Gordon R, Bender R and Herman G T 1970 Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography *J. Theor. Biol.* **29** 471–81
[13] Gubin L G, Polyak B T and Raik E V 1967 The method of projections for finding the common point of convex sets *USSR Comput. Math. Math. Phys.* **7** 1–24
[14] Halperin I 1962 The product of projection operators *Acta Sci. Math.* **23** 96–9
[15] Kaczmarz S 1937 Angenäherte Auflösung von Systemen linearer Gleichungen *Bull. Acad. Pol. Sci. Lett.* A **35** 355–7
[16] Rockafellar R T 1970 *Convex Analysis* (Princeton, NJ: Princeton University Press)
[17] von Neumann J 1950 *Functional Operators, Volume II: The Geometry of Orthogonal Spaces (Annals of Mathematics Studies vol 22)* (Princeton, NJ: Princeton University Press)
[18] Youla D C 1987 Mathematical theory of image restoration by the method of convex projections *Image Recovery: Theory and Applications* ed H Stark (Orlando, FL: Academic) pp 29–78