On sequential and parallel projection algorithms for feasibility and optimization

Yair Censor*
Department of Mathematics, University of Haifa, Mt. Carmel,
Haifa 31905, Israel

ABSTRACT

The convex feasibility problem of finding a point in the intersection of finitely many nonempty closed convex sets in the Euclidean space has many applications in various fields of science and technology, particularly in problems of image reconstruction from projections, in solving the fully discretized inevrse problem in radiation therapy treatment planning, and in other image processing problems. Solving systems of linear equalities and/or inequalities is one of them. Many of the existing algorithms use projections onto the sets and may: (i) employ orthogonal-, entropy-, or other Bregman-projections, (ii) be structurally sequential, parallel, block-iterative, or of the string-averaging type, (iii) asymptotically converge when the underlying system is, or is not, consistent, (iv) solve the convex feasibility problem or find the projection of a given point onto the intersection of the convex sets, (v) have good initial behavior patterns when some of their parameters are appropriately chosen.

Keywords: Projection algorithms, Block-iterative, Bregman projections, convex feasibility, string-averaging.

1. INTRODUCTION

Projection Algorithms employ projections onto convex sets in various ways. They may use different kinds of projections and, sometimes, even use different projections within the same algorithm. They serve to solve a variety of problems which are either of the feasibility or the optimization types. They have different algorithmic structures, of which some are particularly suitable for parallel computing, and they demonstrate nice convergence properties and/or good initial behavior patterns. This class of algorithms has witnessed great progress in recent years and its member algorithms have been applied with success to fully discretized models of problems in image reconstruction and image processing, see, e.g., Stark and Yang,[1] Censor and Zenios.[2] Our aim in this paper is to introduce the reader to this field by reviewing algorithmic structures and specific algorithms for the convex feasibility problem.

The *convex feasibility problem* is to find a point (any point) in the non-empty intersection $C := \cap_{i=1}^{m} C_i \neq \emptyset$ of a family of closed convex subsets $C_i \subseteq R^n$, $1 \leq i \leq m$, of the n-dimensional Euclidean space. It is a fundamental problem in many areas of mathematics and the physical sciences, see, e.g., Combettes[3,4] and references therein. It has been used to model significant real-world problems in image reconstruction from projections, see, e.g., Herman,[5] in radiation therapy treatment planning, see Censor, Altschuler and Powlis[6] and Censor,[7] and in crystallography, see Marks, Sinkler and Landree,[8] to name but a few, and has been used under additional names such as *set theoretic estimation* or the *feasible set approach.* A common approach to such problems is to use projection algorithms, see, e.g., Bauschke and Borwein,[9] which employ *orthogonal projections* (i.e., nearest point mappings) onto the individual sets C_i. The orthogonal projection $P_\Omega(z)$ of a point $z \in R^n$ onto a closed convex set $\Omega \subseteq R^n$ is defined by

$$P_\Omega(z) := \text{argmin}\{\| z - x \|_2 \quad | \quad x \in \Omega\}, \tag{1}$$

where $\| \cdot \|_2$ is the Euclidean norm in R^n. Frequently a *relaxation parameter* is introduced so that

$$P_{\Omega,\lambda}(z) := (1 - \lambda)z + \lambda P_\Omega(z) \tag{2}$$

is the *relaxed projection* of z onto Ω with relaxation λ.

Another problem that is related to the convex feasibility problem is the *best approximation problem* of finding the projection of a given point $y \in R^n$ onto the non-empty intersection $C := \cap_{i=1}^{m} C_i \neq \emptyset$ of a family of closed convex subsets $C_i \subseteq R^n$, $1 \leq i \leq m$, see, e.g., Deutsche's recent book.[10] In both problems the convex sets $\{C_i\}_{i=1}^{m}$ represent mathematical constraints obtained from the modeling of the real-world problem. In the convex feasibility approach

* E-mail: yair@math.haifa.ac.il

any point in the intersection is an acceptable solution to the real-world problem whereas the best approximation formulation is usually appropriate if some point $y \in R^n$ has been obtained from modeling and computational efforts which initially did not take into account the constraints represented by the sets $\{C_i\}_{i=1}^m$ and now one wishes to incorporate them by seeking a point in the intersection of the convex sets which is closest to the point y.

Iterative projection algorithms for finding a projection of a point onto the intersection of sets are more complicated then algorithms for finding just any feasible point in the intersection. This is so because they must have, in their iterative steps, some built-in "memory" mechanism to remember the original point whose projection is sought after. The sequential or parallel algorithms of Dykstra,[11] Haugazeau,[12] Bauschke[13] and others and their modifications employ different such memory mechanisms. We will not deal with these algorithms here although many of them share the same algorithmic structural features described below.

2. BREGMAN PROJECTIONS

Bregman projections onto closed convex sets were introduced by Censor and Lent,[14] based on Bregman's seminal paper,[15] and were subsequently used in a plethora of research works as a tool for building sequential and parallel feasibility and optimization algorithms, see, e.g., Censor and Elfving,[16] Censor and Reich,[17] Censor and Zenios,[2] De Pierro and Iusem,[18] Kiwiel,[19,20] Bauschke and Borwein[21] and references therein, to name but a few.

A *Bregman projection* of a point $z \in R^n$ onto a closed convex set $\Omega \subseteq R^n$ with respect to a, suitably defined, *Bregman function* f (see, e.g., Censor and Zenios[2]) is denoted by $P_\Omega^f(z)$. It is formally defined as

$$P_\Omega^f(z) := \operatorname{argmin}\{D_f(x,z) \mid x \in \Omega \cap \mathrm{cl}S\} \tag{3}$$

where cl$S$ is the closure of the open convex set S, which is the *zone* of f, and $D_f(x,z)$ is the so-called *Bregman distance*, defined by

$$D_f(x,z) := f(x) - f(z) - \langle \nabla f(z), x - z \rangle, \tag{4}$$

for all pairs $(x,z) \in \mathrm{cl}S \times S$, where $\langle \cdot, \cdot \rangle$ is the standard inner product in R^n. If $\Omega \cap \mathrm{cl}S \neq \emptyset$, then (3) defines a unique $P_\Omega^f(z) \in \mathrm{cl}S$, for every $z \in S$.[2] If, in addition, $P_\Omega^f(z) \in S$, for every $z \in S$, then f is called *zone consistent* with respect to Ω. If f is a Bregman/Legendre function (see Bauschke and Borwein[21]) and $S = \mathrm{int}(\mathrm{dom}\, f)$, then f is zone consistent with respect to any closed convex set Ω such that $\Omega \cap \mathrm{cl}S \neq \emptyset$.

Orthogonal projections are a special case of Bregman projections, obtained from (3) by choosing[2] $f(x) = (1/2)\|x\|^2$ and $S = R^n$. Bregman generalized distances and generalized projections are instrumental in several areas of mathematical optimization theory. They were used, among others, in special-purpose minimization methods, in the proximal point minimization method, and for stochastic feasibility problems. These generalized distances and projections were also defined in non-Hilbertian Banach spaces, where, in the absence of orthogonal projections, they can lead to simpler formulas for projections, see, e.g., Butnariu and Iusem[22] and references therein.

Bregman's method for minimizing a convex function (with certain properties) subject to linear inequality constraints employs Bregman projections onto the half-spaces represented by the constraints.[14,18] Recently the extension of this minimization method to nonlinear convex constraints has been identified with the Han-Dykstra projection algorithm for finding the projection of a point onto an intersection of closed convex sets, see Bregman, Censor and Reich.[11] It looks as if there might be no point in using non-orthogonal projections for solving the convex feasibility problem in R^n since they are generally not easier to compute. But this is not always the case. Shamir and co-workers[23,24] have used the multiprojection method of Censor and Elfving[16] to solve filter design problems in image restoration and image recovery posed as convex feasibility problems. They took advantage of that algorithm's flexibility to employ Bregman projections with respect to *different* Bregman functions within the same algorithmic run. Another example is the seminal paper by Csiszár and Tusnády,[25] where the central procedure uses alternating entropy projections onto convex sets. In their "alternating minimization procedure," they alternate between minimizing over the first and second arguments of the Kullback-Leibler divergence. This divergence is nothing but the generalized Bregman distance obtained by using the negative of Shannon's entropy as the underlying Bregman function. Recent studies about Bregman projections (Kiwiel[19]), Bregman/Legendre projections (Bauschke and Borwein[21]), and averaged entropic projections (Butnariu, Censor and Reich[26]) – and their uses for convex feasibility problems in R^n discussed therein – attest to the continued theoretical and practical interest in employing Bregman projections in projection methods for convex feasibility problems.

3. ALGORITHMIC STRUCTURES

Projection algorithmic schemes for the convex feasibility problem and for the best approximation problem are, in general, either *sequential* or *simultaneous* or *block-iterative* (see, e.g., Censor and Zenios[2] for a classification of projection algorithms into such classes, and the review paper of Bauschke and Borwein[9] for a variety of specific algorithms of these kinds). In the following subsections we explain and demonstrate these structures along with the recently proposed *string-averaging* structure. The philosophy behind these algorithms is that it is easier to calculate projections onto the individual sets C_i then onto the whole intersection of sets. Thus, these algorithms call for projections onto individual sets as they proceed sequentially, simultaneously or in the block-iterative or the string-averaging algorithmic modes.

3.1. Sequential Projections

The well-known "Projections Onto Convex Sets" (POCS) algorithm for the convex feasibility problem is a *sequential* projection algorithm, see Bregman,[27] Gubin, Polyak and Raik,[28] Youla[29] and the review papers by Combettes.[3,4] Starting from an arbitrary initial point $x^0 \in R^n$, the POCS algorithm's iterative step is

$$x^{k+1} = x^k + \lambda_k (P_{C_{i(k)}}(x^k) - x^k), \tag{5}$$

where $\{\lambda_k\}_{k\geq 0}$ are relaxation parameters and $\{i(k)\}_{k\geq 0}$ is a *control sequence*, $1 \leq i(k) \leq m$, for all $k \geq 0$, which determines the individual set $C_{i(k)}$ onto which the current iterate x^k is projected. A commonly used control is the *cyclic control* in which $i(k) = k \bmod m+1$, but other controls are also available.[2] Bregman's projection algorithm,[2,15] allowed originally only unrelaxed projections, i.e., its iterative step is of the form

$$x^{k+1} = P^f_{C_{i(k)}}(x^k), \quad \text{for all } k \geq 0. \tag{6}$$

For the Bregman function $f(x) = (1/2)\|x\|^2$ with zone $S = R^n$ and for unity relaxation ($\lambda_k = 1$, for all $k \geq 0$), (6) coincides with (5).

3.2. The String Averaging Algorithmic Structure

The *string-averaging* algorithmic scheme was proposed by Censor, Elfving and Herman.[30] For $t = 1, 2, \ldots, M$, let the *string* I_t be an ordered subset of $\{1, 2, \ldots, m\}$ of the form

$$I_t = (i_1^t, i_2^t, \ldots, i_{m(t)}^t), \tag{7}$$

with $m(t)$ denoting the number of elements in I_t. Suppose that there is a set $S \subseteq R^n$ such that there are operators $R_1, R_2, \ldots, R_m$ mapping S into S and an operator R which maps $S^M = S \times S \times \cdots \times S$ (M times) into S. Initializing the algorithm at an arbitrary $x^0 \in S$, the iterative step of the string-averaging algorithmic scheme is as follows. Given the current iterate x^k, calculate, for all $t = 1, 2, \ldots, M$,

$$T_t x^k = R_{i^t_{m(t)}} \ldots R_{i_2^t} R_{i_1^t} x^k, \tag{8}$$

and then calculate

$$x^{k+1} = R(T_1 x^k, T_2 x^k, \ldots, T_M x^k). \tag{9}$$

For every $t = 1, 2, \ldots, M$, this algorithmic scheme applies to x^k successively the operators whose indices belong to the t-th string. This can be done in parallel for all strings and then the operator R maps all end-points onto the next iterate x^{k+1}. This is indeed an algorithm provided that the operators $\{R_i\}_{i=1}^m$ and R all have algorithmic implementations. In this framework we get a *sequential* algorithm by the choice $M = 1$ and $I_1 = (1, 2, \ldots, m)$ and a *simultaneous* algorithm by the choice $M = m$ and $I_t = (t),\ t = 1, 2, \ldots, M$.

We demonstrate the underlying idea of the string-averaging algorithmic scheme with the aid of Figure 1. For simplicity, we take the convex sets to be hyperplanes, denoted by H_1, H_2, H_3, H_4, H_5, and H_6, and assume all operators R_i to be orthogonal projections onto the hyperplanes. The operator R is taken as a convex combination

$$R(x^1, x^2, \ldots, x^M) = \sum_{t=1}^{M} \omega_t x^t, \tag{10}$$

with $\omega_t > 0$, for all $t = 1, 2, \ldots, M$, and $\sum_{t=1}^{M} \omega_t = 1$.

Figure 1(a) depicts the purely sequential algorithm. This is the so-called POCS (Projections Onto Convex Sets) algorithm which coincides, for the case of hyperplanes, with the Kaczmarz algorithm, see, e.g., Algorithms 5.2.1 and 5.4.3, respectively, in Ref. 2. The fully simultaneous algorithm appears in Figure 1(b). With orthogonal reflections instead of orthogonal projections it was first proposed, by Cimmino,[31] for solving linear equations. Here the current iterate x^k is projected on all sets simultaneously and the next iterate x^{k+1} is a convex combination of the projected points. In Figure 1(c) we show how a simple averaging of *successive* projections (as opposed to averaging of parallel projections in Figure 1(b)) works. In this case $M = m$ and $I_t = (1, 2, \ldots, t)$, for $t = 1, 2, \ldots, M$. This scheme, appearing in Bauschke and Borwein,[9] inspired the formulation of the general string-averaging algorithmic scheme whose action is demonstrated in Figure 1(d). It averages, via convex combinations, the end-points obtained from strings of sequential projections and in this figure the strings are $I_1 = (1, 3, 5, 6)$, $I_2 = (2)$, $I_3 = (6, 4)$. Such schemes offer a variety of options for steering the iterates towards a solution of the convex feasibility problem. It is an *inherently parallel* scheme in that its mathematical formulation is parallel (like the fully simultaneous method mentioned above). We use this term to contrast such algorithms with others which are sequential in their mathematical formulation but can, sometimes, be implemented in a parallel fashion based on appropriate model decomposition (i.e., depending on the structure of the underlying problem). Being inherently parallel, this algorithmic scheme enables flexibility in the actual manner of implementation on a parallel machine.

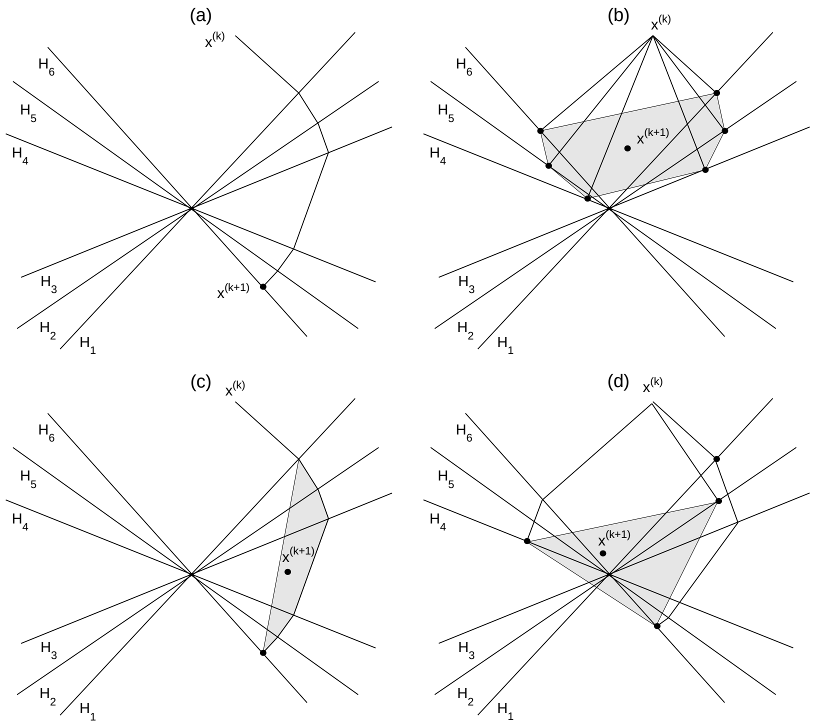

Figure 1. (a) Sequential projections. (b) Fully simultaneous projections. (c) Averaging of sequential projections. (d) String-averaging. (Reproduced from Censor, Elfving and Herman[30]).

At the extremes of the "spectrum" of possible specific algorithms, derivable from the string-averaging algorithmic scheme, are the generically sequential method, which uses one set at a time, and the fully simultaneous algorithm, which employs all sets at each iteration. The "block-iterative projections" (BIP) scheme of Aharoni and Censor[32] also has the sequential and the fully simultaneous methods as its extremes in terms of block structures (see also Butnariu and Censor,[33] Bauschke and Borwein,[9] Bauschke, Borwein and Lewis,[34] Elfving[35] and Eggermont, Herman and Lent[36]). The question whether there are any other relationships between the BIP and the string-averaging algorithmic schemes is of theoretical interest and is still open. However, the string-averaging algorithmic structure gives users a tool to design many new inherently parallel computational schemes.

The behavior of the string-averaging algorithmic scheme, or special instances of it, in the inconsistent case when the intersection $C = \cap_{i=1}^{m} C_i$ is empty is also not fully answered at this time. For results on the behavior of the fully simultaneous algorithm with orthogonal projections in the inconsistent case see, e.g., Combettes[37] or Iusem and De Pierro.[38] Another way to treat possible inconsistencies is to reformulate the constraints as $c \leq Ax \leq d$ or $||Ax - d||_2 \leq \epsilon$. Also, variable iteration-dependent relaxation parameters and variable iteration-dependent string constructions could be interesting future extensions. The practical performance of specific algorithms needs also to be evaluated in applications and on parallel machines.

3.3. The Block-Iterative Algorithmic Scheme With Underrelaxed Bregman Projections

In this subsection we briefly review the *block-iterative algorithmic scheme* with (underrelaxed) Bregman projections for the solution of the convex feasibility problem proposed by Censor and Herman.[39] By *block-iterative* we mean that, at the k-th iteration, the next iterate x^{k+1} is generated from the current iterate x^k by using a subset (called a block) of the family of sets $\{C_i\}_{i=1}^{m}$ of the convex feasibility problem.[2] We use the term *algorithmic scheme* to emphasize that different specific algorithms may be derived by different choices of Bregman functions, and by various block structures. For example, if all blocks consist of a single set C_i, then our scheme gives rise to a sequential *row-action*[46] type algorithm. Taking the other extreme, if we let every block contain all sets, then we obtain a fully simultaneous algorithm. Such a block-iterative scheme for the convex feasibility problem was first proposed by Aharoni and Censor,[32] using orthogonal projections onto convex sets. That block-iterative projections (BIP) method generalizes the sequential POCS method. The block-iterative scheme, described below, extends Aharoni and Censor's BIP method by employing underrelaxed Bregman projections which contain the underrelaxed orthogonal projections as a special case. The underrelaxed Bregman projection with Bregman function f and relaxation parameter $\lambda \in [0, 1]$ of a point z onto a closed convex set Ω, denoted by $P^f_{\Omega,\lambda}(z)$, is given by

$$\nabla f(P^f_{\Omega,\lambda}(z)) = (1 - \lambda)\nabla f(z) + \lambda \nabla f(P^f_{\Omega}(z)). \tag{11}$$

Appealing to the definition of a convex combination with respect to a Bregman function f as defined by Censor and Reich,[17] the natural formula for a block-iterative step using underrelaxed Bregman projections is

$$\nabla f(x^{k+1}) = \sum_{i=1}^{m} v_i^k \nabla f(P^f_{C_i,\lambda_i^k}(x^k)), \tag{12}$$

where x^k is the k-th iterate, $\lambda_i^k \in [0, 1]$ is the relaxation parameter used in the underrelaxed Bregman projection onto the set C_i during the k-th iterative step and the v_i^k are the weights of the convex combination for the k-th iterative step (i.e., $v_i^k \geq 0$ for $1 \leq i \leq m$ and $\sum_{i=1}^{m} v_i^k = 1$). Substituting (11) into (12), defining $w_i^k := v_i^k \lambda_i^k$, for $1 \leq i \leq m$, and introducing

$$w_{m+1}^k := 1 - \sum_{i=1}^{m} w_i^k \quad \text{and} \quad C_{m+1} := R^n, \tag{13}$$

we get the following alternative formulation of the block-iterative step (12)

$$\nabla f(x^{k+1}) = \sum_{i=1}^{m+1} w_i^k \nabla f(P^f_{C_i}(x^k)), \tag{14}$$

with $w_i^k \geq 0$ for $1 \leq i \leq m+1$ and $\sum_{i=1}^{m+1} w_i^k = 1$. The block-iterative nature of this scheme stems from the fact that for every iteration index k some of the parameters w_i^k can be set to zero. The set of those indices i for which $w_i^k \neq 0$ at the k-th iteration defines the "block" of active constraints at this iteration. These index sets might vary dynamically from iteration to iteration as long as some technical conditions are observed.[39]

Many other block-iterative algorithms were studied by Byrne[40–43] in reference to image reconstruction from projections, where such algorithmic schemes are sometimes termed *ordered subset methods.* See also the work of Combettes[44] and Section 6 of his paper on quasi-Fejérian methods.[45]

4. BICAV: BLOCK-ITERATIVE COMPONENT AVERAGING

A recent member of the powerful family of block-iterative projection algorithms is the *BICAV (block-iterative component averaging)* algorithm of Censor, Gordon and Gordon[47] which was applied to a problem of image reconstruction from projections. The BICAV algorithm is a block-iterative companion to the *CAV (Component averaging)* method for solving systems of linear equations.[48] In these methods the sparsity of the matrix is explicitly used when constructing the iteration formula. Using this new scaling considerable improvement was observed compared to traditionally scaled iteration methods.

In Cimmino's simultaneous projections method,[31] see also, e.g., Censor and Zenios[2] with relaxation parameters and with equal weights $w_i = 1/m$, the next iterate x^{k+1} is the average of the orthogonal projections of x^k onto the hyperplanes H_i defined by the i-th row of the linear system $Ax = b$ and has, for every component $j = 1, 2, \dots, n$, the form

$$x_j^{k+1} = x_j^k + \frac{\lambda_k}{m} \sum_{i=1}^{m} \frac{b_i - \langle a^i, x^k \rangle}{\|a^i\|_2^2} a_j^i \,, \tag{15}$$

where a^i is the i-th column of the transpose A^T of A and b_i is the i-th component of the vector b and λ_k are relaxation parameters. When the $m \times n$ system matrix $A = (a_j^i)$ is sparse, only a relatively small number of the elements $\{a_j^1, a_j^2, \dots, a_j^m\}$ of the j-th column of A are nonzero, but in (15) the sum of their contributions is divided by the relatively large m. This observation led to the replacement of the factor $1/m$ in (15) by a factor that depends only on the *nonzero* elements in the set $\{a_j^1, a_j^2, \dots, a_j^m\}$. For each $j = 1, 2, \dots, n$, denote by s_j the number of nonzero elements of column j of the matrix A, and replace (15) by

$$x_j^{k+1} = x_j^k + \frac{\lambda_k}{s_j} \sum_{i=1}^{m} \frac{b_i - \langle a^i, x^k \rangle}{\|a^i\|_2^2} a_j^i \,. \tag{16}$$

Certainly, if A is sparse then the s_j values will be much smaller than m. The iterative step (15) is a special case of

$$x^{k+1} = x^k + \lambda_k \sum_{i=1}^{m} w_i \frac{b_i - \langle a^i, x^k \rangle}{\|a^i\|_2^2} a^i \,, \tag{17}$$

where the fixed weights $\{w_i\}_{i=1}^m$ must be positive for all i and $\sum_{i=1}^m w_i = 1$. The attempt to use $1/s_j$ as weights in (16) does not fit into the scheme (17), unless one can prove convergence of the iterates of a fully simultaneous iterative scheme with component-dependent (i.e., j-dependent) weights of the form

$$x_j^{k+1} = x_j^k + \lambda_k \sum_{i=1}^{m} w_{ij} \frac{b_i - \langle a^i, x^k \rangle}{\|a^i\|_2^2} a_j^i \,, \tag{18}$$

for all $j = 1, 2, \dots, n$. To formalize this consider a set $\{G_i\}_{i=1}^m$ of real diagonal $n \times n$ matrices $G_i = \mathrm{diag}(g_{i1}, g_{i2}, \dots, g_{in})$ with $g_{ij} \geq 0$, for all $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n$, such that $\sum_{i=1}^m G_i = I$, where I is the unit matrix. Referring to the sparsity pattern of A one needs the following definition.[48]

DEFINITION 1. *A family $\{G_i\}_{i=1}^m$ of real diagonal $n \times n$ matrices with all diagonal elements $g_{ij} \geq 0$ and such that $\sum_{i=1}^m G_i = I$ is called* **sparsity pattern oriented** *(SPO, for short)* **with respect to an** *$m \times n$* **matrix** *A if, for every $i = 1, 2, \dots, m$, $g_{ij} = 0$ if and only if $a_j^i = 0$.*

The Component Averaging (CAV) algorithm combines three features: (i) Each orthogonal projection onto H_i in is replaced by a *generalized oblique projection with respect to* G_i, denoted below by $P_{H_i}^{G_i}$. (ii) The scalar weights $\{w_i\}$ in (17) are replaced by the diagonal weighting matrices $\{G_i\}$. (iii) The actual weights are set to be inversely proportional to the number of nonzero elements in each column, as motivated by the discussion preceding Equation (16). The iterative step resulting from the first two features has the form

$$x^{k+1} = x^k + \lambda_k \sum_{i=1}^{m} G_i \left(P_{H_i}^{G_i}(x^k) - x^k \right) . \tag{19}$$

The basic idea of the block-iterative CAV (BICAV) algorithm is to break up the system $Ax = b$ into "blocks" of equations and treat each block according to the CAV methodology, passing cyclically over all the blocks. Throughout the following, T will be the number of blocks and, for $t = 1, 2, \ldots, T$, let the block of indices $B_t \subseteq \{1, 2, \ldots, m\}$, be an ordered subset of the form $B_t = \{i_1^t, i_2^t, \ldots, i_{m(t)}^t\}$, where $m(t)$ is the number of elements in B_t, such that every element of $\{1, 2, \ldots, m\}$ appears in at least one of the sets B_t. For $t = 1, 2, \ldots, T$, let A_t denote the matrix formed by taking all the rows of A whose indices belong to the block of indices B_t, i.e.,

$$A_t := \begin{bmatrix} a^{i_1^t} \\ a^{i_2^t} \\ \vdots \\ a^{i_{m(t)}^t} \end{bmatrix}, \quad t = 1, 2, \ldots, T. \tag{20}$$

The iterative step of the BICAV algorithm, developed and experimentally tested by Censor, Gordon and Gordon,[47] uses, for every block index $t = 1, 2, \ldots, T$, generalized oblique projections with respect to a family $\{G_i^t\}_{i=1}^m$ of diagonal matrices which are SPO with respect to A_t. The same family is also used to perform the diagonal weighting. The resulting iterative step has the form

$$x^{k+1} \;=\; x^k + \lambda_k \sum_{i \in B_{t(k)}} G_i^{t(k)} \left(P_{H_i}^{G_i^{t(k)}}(x^k) - x^k \right), \tag{21}$$

where $\{t(k)\}_{k \geq 0}$ is a *control sequence* according to which the $t(k)$-th block is chosen by the algorithm to be acted upon at the k-th iteration, thus, $1 \leq t(k) \leq T$, for all $k \geq 0$. The real numbers $\{\lambda_k\}_{k \geq 0}$ are user-chosen *relaxation parameters.* Finally, in order to achieve the acceleration, the diagonal matrices $\{G_i^t\}_{i=1}^m$ are constructed with respect to each A_t. Let s_j^t be the number of nonzero elements $a_j^i \neq 0$ in the j-th column of A_t and define

$$g_{ij}^t := \begin{cases} \dfrac{1}{s_j^t}, & \text{if } a_j^i \neq 0, \\[2ex] 0, & \text{if } a_j^i = 0. \end{cases} \tag{22}$$

It is easy to verify that, for each $t = 1, 2, \ldots, T$, $\sum_{i=1}^m G_i^t = I$ holds for these matrices. With these particular SPO families of G_i^t's one obtains the block-iterative algorithm:

ALGORITHM 2. ***BICAV***

Initialization: *$x^0 \in R^n$ is arbitrary.*

Iterative Step: *Given x^k, compute x^{k+1} by using, for $j = 1, 2, \ldots, n$, the formula:*

$$x_j^{k+1} \;=\; x_j^k + \lambda_k \sum_{i \in B_{t(k)}} \frac{b_i - \langle a^i, x^k \rangle}{\sum_{l=1}^n s_l^{t(k)} (a_l^i)^2} a_j^i, \tag{23}$$

where λ_k are relaxation parameters, $\{s_l^t\}_{l=1}^n$ are as defined above, and the control sequence is cyclic, i.e., $t(k) = k \bmod T + 1$, for all $k \geq 0$.

ACKNOWLEDGMENTS

We thank our co-authors Tommy Elfving and Gabor Herman[30,39] and Dan Gordon and Rachel Gordon[47,48] for their collaboration on these papers from which some of the results and material presented here are taken. This work was supported by grant No. 592/00 of the Israel Science Foundation, founded by the Israel Academy of Sciences and Humanities.

REFERENCES

1. H. Stark and Y. Yang, *Vector Space Projections: A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics,* John Wiley & Sons, New York, NY, USA, 1998.
2. Y. Censor and S.A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications*, Oxford University Press, New York, NY, USA, 1997.
3. P.L. Combettes, "The foundations of set-theoretic estimation," *Proceedings of the IEEE* **81**, pp. 182–208, 1993.
4. P.L. Combettes, "The convex feasibility problem in image recovery," *Advances in Imaging and Electron Physics* **95**, pp. 155–270, 1996.
5. G.T. Herman, *Image Reconstruction From Projections: The Fundamentals of Computerized Tomography*, Academic Press, New York, NY, USA, 1980.
6. Y. Censor, M.D. Altschuler and W.D. Powlis, "On the use of Cimmino's simultaneous projections method for computing a solution of the inverse problem in radiation therapy treatment planning," *Inverse Problems* **4**, pp. 607–623, 1988.
7. Y. Censor, "Mathematical aspects of radiation therapy treatment planning: continuous inversion versus full discretization and optimization versus feasibility," in *Computational Radiology and Imaging: Therapy and Diagnostics,* C. Börgers and F. Natterer, eds., pp. 101–112, The IMA Volumes in Mathematics and its Applications, Vol. **110**, Springer-Verlag, New York, NY, USA, 1999.
8. L.D. Marks, W. Sinkler and E. Landree, "A feasible set approach to the crystallographic phase problem," *Acta Crystallographica* **A55**, pp. 601–612, 1999.
9. H.H. Bauschke and J.M. Borwein, "On projection algorithms for solving convex feasibility problems," *SIAM Review* **38**, pp. 367–426, 1996.
10. F. Deutsch, *Best Approximation in Inner product Spaces*, Springer-Verlag, New York, NY, USA, 2001.
11. L.M. Bregman, Y. Censor and S. Reich, "Dykstra's algorithm as the nonlinear extension of Bregman's optimization method," *Journal of Convex Analysis* **6**, pp. 319–334, 1999.
12. H.H. Bauschke and P.L. Combettes, "A weak-to-strong convergence principle for Fejér-monotone methods in Hilbert spaces," *Mathematics of Operations Research* **26**, pp. 248–264, 2001.
13. H.H. Bauschke, "The approximation of fixed points of compositions of nonexpansive mappings in Hilbert space," *Journal of Mathematical Analysis and Applications* **202**, pp. 150–159, 1996.
14. Y. Censor and A. Lent, "An iterative row-action method for interval convex programming," *Journal of Optimization Theory and Applications* **34**, pp. 321–353, 1981.
15. L.M. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR Computational Mathematics and Mathematical Physics* **7**, pp. 200–217, 1967.
16. Y. Censor and T. Elfving, "A multiprojections algorithm using Bregman projections in a product space," *Numerical Algorithms* **8**, pp. 221–239, 1994.
17. Y. Censor and S. Reich, "Iterations of paracontractions and firmly nonexpansive operators with applications to feasibility and optimization," *Optimization* **37**, pp. 323–339, 1996.
18. A.R. De Pierro and A.N. Iusem, "A relaxed version of Bregman's method for convex programming," *Journal of Optimization Theory and Applications* **51**, pp. 421–440, 1986.
19. K. Kiwiel, "Generalized Bregman projections in convex feasibility problems," *Journal of Optimization Theory and Applications* **96**, pp. 139–157, 1998.
20. K. Kiwiel, "Free–steering relaxation methods for problems with strictly convex costs and linear constraints," *Mathematics of Operations Research* **22**, pp. 326–349, 1997.
21. H.H. Bauschke and J.M. Borwein, "Legendre functions and the method of random Bregman projections," *Journal of Convex Analysis* **4**, pp. 27–67, 1997.
22. D. Butnariu and A.N. Iusem, *Totally Convex Functions for Fixed Points Computation and Infinite Dimensional Optimization*, Kluwer Academic Publishers, Dordrecht, the Netherlands, 2000.
23. T. Kotzer, N. Cohen and J. Shamir, "A projection-based algorithm for consistent and inconsistent constraints," *SIAM Journal on Optimization* **7**, pp. 527–546, 1997.
24. D. Lyszyk and J. Shamir, "Signal processing under uncertain conditions by parallel projections onto fuzzy sets," *Journal of the Optical Society of America A* **16**, pp. 1602–1611, 1999.

25. I. Csiszár and G. Tusnády, "Information geometry and alternating minimization procedures," *Statistics and Decisions* Supplement Issue No. **1**, pp. 205–237, 1984.
26. D. Butnariu, Y. Censor and S. Reich, "Iterative averaging of entropic projections for solving stochastic convex feasibility problems," *Computational Optimization and Applications* **8**, pp. 21–39, 1997.
27. L.M. Bregman, "The method of successive projections for finding a common point of convex sets," *Soviet Mathematics Doklady* **6**, pp. 688–692, 1965.
28. L. Gubin, B. Polyak and E. Raik, "The method of projections for finding the common point of convex sets," *USSR Computational Mathematics and Mathematical Physics* **7**, pp. 1–24, 1967.
29. D.C. Youla, "Mathematical theory of image restoration by the method of convex projections," in *Image Recovery: Theory and Applications*, H. Stark, ed., pp. 29–77, Academic Press, Orlando, Florida, USA, 1987.
30. Y. Censor, T. Elfving and G.T. Herman, "Averaging strings of sequential iterations for convex feasibility problems," in *Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications*, D. Butnariu, Y. Censor, and S. Reich, eds., pp. 101–114, Elsevier Science Publishers, Amsterdam, The Netherlands, 2001.
31. G. Cimmino, "Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari," *La Ricerca Scientifica* **XVI** Series II, Anno IX, **1,** pp. 326–333, 1938.
32. R. Aharoni and Y. Censor, "Block-iterative projection methods for parallel computation of solutions to convex feasibility problems," *Linear Algebra and Its Applications* **120**, pp. 165–175, 1989.
33. D. Butnariu and Y. Censor, "Strong convergence of almost simultaneous block-iterative projection methods in Hilbert spaces," *Journal of Computational and Applied Mathematics* **53**, pp. 33–42, 1994.
34. H.H. Bauschke, J.M. Borwein and A.S. Lewis, "The method of cyclic projections for closed convex sets in Hilbert space," *Contemporary Mathematics* **204**, pp. 1–38, 1997.
35. T. Elfving, "Block-iterative methods for consistent and inconsistent linear equations," *Numerische Mathematik* **35**, pp. 1–12, 1980.
36. P.P.B. Eggermont, G.T. Herman and A. Lent, "Iterative algorithms for large partitioned linear systems, with applications to image reconstruction," *Linear Algebra and Its Applications* **40**, pp. 37–67, 1981.
37. P.L. Combettes, "Inconsistent signal feasibility problems: least-squares solutions in a product space," *IEEE Transactions on Signal Processing* **SP-42**, pp. 2955–2966, 1994.
38. A.N. Iusem and A.R. De Pierro, "Convergence results for an accelerated nonlinear Cimmino algorithm," *Numerische Mathematik* **49**, pp. 367–378, 1986.
39. Y. Censor and G.T. Herman, "Block-iterative algorithms with underrelaxed Bregman projections," *Technical Report*, May 2001.
40. C.L. Byrne, "Block-iterative methods for image reconstruction from projections," *IEEE Transactions on Image Processing* **IP-5**, pp. 792–794, 1996.
41. C.L. Byrne, "Block-iterative interior point optimization methods for image reconstruction from limited data," *Inverse Problems* **16**, pp. 1405–1419, 2000.
42. C.L. Byrne, "Bregman-Legendre multidistance projection algorithms for convex feasibility and optimization," in *Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications*, D. Butnariu, Y. Censor, and S. Reich, eds., pp. 87–99, Elsevier Science Publishers, Amsterdam, The Netherlands, 2001.
43. C.L. Byrne, "Notes on block-iterative or ordered subset methods for image reconstruction," *Technical Report*, September 2000.
44. P.L. Combettes, "Strong convergence of block-iterative outer approximation methods for convex optimization," *SIAM Journal on Control and Optimization* **38**, pp. 538–565, 2000.
45. P.L. Combettes, "Quasi-Fejérian analysis of some optimization algorithms," in *Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications*, D. Butnariu, Y. Censor, and S. Reich, eds., pp. 115–152, Elsevier Science Publishers, Amsterdam, The Netherlands, 2001.
46. Y. Censor, "Row-action methods for huge and sparse systems and their applications," *SIAM Review* **23,** pp. 444-466, 1981.
47. Y. Censor, D. Gordon and R. Gordon, "BICAV: A block-iterative, parallel algorithm for sparse systems with pixel-dependent weighting," *IEEE Transactions on Medical Imaging,* to appear.
48. Y. Censor, D. Gordon and R. Gordon, "Component averaging: An efficient iterative parallel algorithm for large and sparse unstructured problems," *Parallel Computing* **27**, pp. 777–808, 2001.