# THE DYKSTRA ALGORITHM WITH BREGMAN PROJECTIONS


Yair Censor[1] and Simeon Reich[2]

[1]Department of Mathematics and Computer Science
University of Haifa, Mt. Carmel, 31905 Haifa, Israel.
E-mail: yair@mathcs2.haifa.ac.il

[2]Department of Mathematics
The Technion — Israel Institute of Technology
32000 Haifa, Israel.
E-mail: sreich@techunix.technion.ac.il



**ABSTRACT:** Let $\{C_i \mid 1 \leq i \leq m\}$ be a finite family of closed convex subsets of $\mathbf{R}^n$, and assume that their intersection $C = \cap\{C_i \mid 1 \leq i \leq m\}$ is not empty. In this paper we propose a general Dykstra-type sequential algorithm for finding the Bregman projection of a given point $r \in \mathbf{R}^n$ onto $C$ and show that it converges in several important special cases.



**Keywords**. Best approximation, Bregman projection, convex feasibility, Dykstra's algorithm, Hildreth's algorithm, $I$-projection.

**1991 Mathematics Subject Classification**. 47N10, 49M30, 90C25.


## 1. INTRODUCTION

The Dykstra algorithm is an iterative procedure which (asymptotically) finds the nearest point projection (also called the orthogonal projection) of any given point onto the intersection of a given finite family of closed convex sets. It iterates by passing sequentially over the individual sets and projecting onto each one a deflected version of the previous iterate. The algorithm was first proposed and analyzed by Dykstra in 1983 [21] for a family of closed convex cones in the Euclidean space $\mathbf{R}^n$ and, subsequently, by Boyle and Dykstra [4] for convex sets in a Hilbert space. In 1988 Han [26] rediscovered the algorithm, investigating its behavior in $\mathbf{R}^n$ in the framework of the duality theory of mathematical programming (see also the related work of Han and Lou [27]). Gaffke and Mathar [25] studied the Dykstra algorithm in Hilbert space from a duality standpoint and showed its relation to the method of componentwise cyclic minimization over a Cartesian product. They also proposed a fully simultaneous Dykstra algorithm. Iusem and De Pierro published in 1991 their study [32] in which they used Pierra's [35] product space formalism to show convergence of the simultaneus Dykstra algorithm in both the consistent and the inconsistent cases in $\mathbf{R}^n$. Crombez [16] did such an analysis in Hilbert space.

Combettes included the Dykstra algorithm in his short review [15]. Bauschke and Borwein [2] analyzed Dykstra's algorithm for two sets in Hilbert space and generalized the work of Iusem and De Pierro [32] to this setting (see also Bauschke's thesis [1]). Recently, Deutsch and Hundal published a rate of convergence study for the polyhedral case [20], and generalizations to an infinite family of sets and to random, rather then cyclic, order control [30]. Han [26], as well as Iusem and De Pierro [32], show that for linear inequality constraints and for linear interval inequalities constraints (the polyhedral case), the method of Dykstra becomes the Hildreth algorithm, first published in [29] and studied further by D'Esopo [23] and by Lent and Censor [34], and the ART4 algorithm of Herman and Lent [28], respectively.

In all the above mentioned investigations only orthogonal projections onto closed convex sets are discussed, and it is natural to ask whether or not the algorithmic framework of Dykstra can accomodate also different, non-orthogonal, projections. If so, what kind of other projections should be employed? How should the more general Dykstra algorithm look like? Which, if any, of the earlier results on the Dykstra method carry over?

In this report we propose an algorithmic structure, based on the theory of Bregman functions, distances and projections, which is a starting point for generalizing Dykstra's algorithm to non-orthogonal projections. We are unable, at this time, to furnish a complete convergence theory for our new scheme, but we do show below how it is related to the following methods:

Dykstra's original sequential method with orthogonal projections; (1.1)

Hildreth's and the ART4 methods in the polyhedral case; (1.2)

Dykstra's procedure with $I$-projections [22]; (1.3)

and

Bregman's primal-dual minimization methods in the polyhedral case (see Bregman [5] and Censor and Lent [11]). (1.4)

All these turn out to be special cases of our new scheme and as such, provide partial convergence results for it. We conjecture that a general convergence theory can be developed for our general, sequential non-orthogonal Dykstra algorithm.

Furthermore, it should be possible to develop a simultaneous non-orthogonal Dykstra algorithm along the lines of Iusem and De Pierro [32] based on Censor and Elfving [9].

Our presentation is organized as follows. In Section 2 we present the new non-orthogonal algorithmic scheme and show how the classical orthogonal projections algorithm is obtained from it. In Section 3 we prove the convergence of the new scheme with non-orthogonal projections for the polyhedral case by reduction to Bregman's method. The analysis in Section 4 makes precise the relationship of our scheme with Dykstra's procedure for $I$-projections. For the reader's convenience we attach an Appendix with a brief summary of definitions and relevant results from the theory of Bregman's distances and projections.

## 2. THE ALGORITHMIC SCHEME WITH NONORTHOGONAL PROJECTIONS

Let $C_i \subseteq \mathbf{R}^n$, $i = 1, 2, \ldots, m$, be a finite family of closed convex nonempty subsets of the $n$-dimensional Euclidean space, and assume that $C \triangleq \bigcap_{i=1}^m C_i \neq \emptyset$. Let $f \in \mathcal{B}(S)$ be a Bregman function with zone $S$ and let $D_f(x, y)$ be the generalized distance function associated with $f$ (consult the Appendix at the end of the paper for definitions and results about Bregman functions, generalized distances and projections). The problem under consideration is:

$$\begin{array}{l} \min D_f(x, r) \\ \text{s.t. } x \in C \cap \overline{S} \end{array} \tag{2.1}$$

where $r \in S$ is a given point. Our goal is to find the projection $P_C^f(r)$ of $r$ onto $C$ with respect to $f$. The proposed Dykstra-type sequential algorithmic scheme solves this problem by sequentially projecting w.r.t. $f$, onto each set $C_i$, a deflected version of the previous iterate.

In the sequel we denote the gradient of $f$ by $\nabla f$. If $f$ is also essentially smooth, then $y = \nabla f(x)$ is a one-to-one mapping with a continuous inverse $(\nabla f)^{-1}$, see, e.g., Rockafellar [37], Corollary 26.3.1. The applicability of the following Algorithm 2.1 depends on the ability to invert $\nabla f$ explicitly to get a workable formula in any given case.

**Algorithm 2.1**

**Initialization:** *Set the vectors* $\{y_{(i)}^{(0)}\}_{i=1}^m$ *so that*

$$\nabla f(y_{(i)}^{(0)}) = 0, \qquad i = 1, 2, \ldots, m, \tag{2.2}$$

*and set* $x_{(m)}^{(0)} \triangleq r$*, the given point.*

**Iterative step:** *Compute the* $2m$ *vectors* $\{x_{(i)}^{(k)}\}_{i=1}^m$, $\{y_{(i)}^{(k)}\}_{i=1}^m$ *as follows:*

*Set* $x^{(k)}_{(0)} = x^{(k-1)}_{(m)}$ *and, for* $i = 1, 2, \ldots, m$, *calculate* $z^{(k)}_{(i)}$ *from*

$$\nabla f(z^{(k)}_{(i)}) = \nabla f(x^{(k)}_{(i-1)}) + \nabla f(y^{(k-1)}_{(i)}); \tag{2.3}$$

*then project* $z^{(k)}_{(i)}$ *onto* $C_i$ *w.r.t.* $f$, *i.e.,*

$$x^{(k)}_{(i)} = P^f_i(z^{(k)}_{(i)}), \tag{2.4}$$

*where* $P^f_i = P^f_{C_i}$, *and finally update the "memory" vector by calculating* $y^{(k)}_{(i)}$ *such that*

$$\nabla f(y^{(k)}_{(i)}) = \nabla f(z^{(k)}_{(i)}) - \nabla f(P^f_i(z^{(k)}_{(i)})). \tag{2.5}$$

Observe that in practice $y^{(k)}_{(i)}$ need not be found explicitly from (2.5) since only the gradient $\nabla f(y^{(k-1)}_{(i)})$ is needed in (2.3).

If we choose the Bregman function $f(x) = \frac{1}{2}\|x\|^2$ with zone $S = \overline{S} = \mathbf{R}^n$, then $\nabla f(x) = x$, $D_f(x, y) = \frac{1}{2}\|x - y\|^2$ and $P^f_i$ is the orthogonal projection $P_i$ onto $C_i$. In this case Algorithm 2.1 becomes the original Dykstra algorithm with $y^{(0)}_{(i)} = 0$ for all $i = 1, 2, \ldots, m$, and with (2.3) and (2.5) replaced by

$$z^{(k)}_{(i)} = x^{(k)}_{(i-1)} + y^{(k-1)}_{(i)}, \tag{2.6}$$

and by

$$y^{(k)}_{(i)} = z^{(k)}_{(i)} - P_i(z^{(k)}_{(i)}), \tag{2.7}$$

respectively. See, e.g., Han [26].

## 3. CONVERGENCE IN THE POLYHEDRAL CASE

The polyhedral case occurs when the sets are half-spaces $C_i = \{x \in \mathbf{R}^n \mid \langle a^{(i)}, x\rangle \leq \alpha_i\}$, where $0 \neq a^{(i)} \in \mathbf{R}^n$ and $\alpha_i \in \mathbf{R}$, for all $i = 1, 2, \ldots, m$, are given. Han [26] and Iusem and De Pierro [32] have shown that in this case the original Dykstra algorithm coincides with Hildreth's algorithm; see Hildreth [29], D'Esopo [23], Lent and Censor [34], or Censor and Zenios [14] (Han [26] actually considers sets of the form $\{x \in \mathbf{R}^n \mid \beta_i \leq \langle a^{(i)}, x\rangle \leq \alpha_i\}$ in which case the Dykstra algorithm becomes identical with the algebraic reconstruction technique ART4 of Herman and Lent [28].) Deutsch and Hundal [20] studied the rate of convergence of the method in the polyhedral case.

Here we consider the polyhedral case of Algorithm 2.1 and prove the following convergence result (consult the Appendix for the meaning of strong zone consistency).

**Theorem 3.1** *Let* $f \in \mathcal{B}(S)$, $C_i \triangleq \{x \in \mathbf{R}^n \mid \langle a^{(i)}, x\rangle \leq \alpha_i\}$ *for* $i = 1, 2, \ldots, m$, $C \triangleq \cap_{i=1}^m C_i$, *and assume that* $C \cap \overline{S} \neq \emptyset$, *and that* $f$ *is strongly zone consistent w.r.t.*

*each* $H_i \triangleq \{x \in \mathbf{R}^n \mid \langle a^{(i)}, x\rangle = \alpha_i\}$. *Let* $r \in S$ *and assume that there exists a vector* $\pi \in \mathbf{R}_+^m$ *such that*

$$\nabla f(r) = -A^T \pi, \tag{3.1}$$

*where* $A^T$ *is the* $n \times m$ *matrix with* $a^{(i)}$ *in its* $i$*-th column. Then any sequence* $\{x^{(k)}\}_{k=0}^{\infty}$, *where* $x^{(k)} \triangleq x_{(m)}^{(k)}$, *generated by Algorithm 2.1, converges to the solution of (2.1).*

**Proof.** The idea of the proof is to show that under the assumptions of the theorem, Algorithm 2.1 reduces to Bregman's algorithm for the minimization of $f(x)$ over $C \cap \overline{S}$, see Algorithm A.1 in the Appendix. In the $k$-th iterative step of Algorithm 2.1, when the $i$-th half-space is iterated upon, one of the following two cases occurs.

**Case I.** $\langle a^{(i)}, z_{(i)}^{(k)}\rangle \leq \alpha_i$. Then, by (2.4), $x_{(i)}^{(k)} = P_i^f(z_{(i)}^{(k)}) = z_{(i)}^{(k)}$ and, by (2.5), $\nabla f(y_{(i)}^{(k)}) = 0$.

**Case II.** $\langle a^{(i)}, z_{(i)}^{(k)}\rangle > \alpha_i$. Then, according to the formula for Bregman projections onto a hyperplane (see, equations (2.14)–(2.15) of [5], Lemma 3.1 of [11], or Lemma 2.2.1 of [14]), there exists a unique real number $\lambda_i^k$ such that

$$\nabla f(x_{(i)}^{(k)}) = \nabla f(z_{(i)}^{(k)}) - \lambda_i^k a^{(i)}, \tag{3.2}$$

and

$$\langle a^{(i)}, x_{(i)}^{(k)}\rangle = \alpha_i. \tag{3.3}$$

¿From Lemma 3.2 of [11] we know that $\lambda_i^k(\alpha_i - \langle a^{(i)}, x_{(i)}^{(k)}\rangle) > 0$, and therefore $\lambda_i^k > 0$ must hold in Case II.

We can think of (3.2) as covering both Case I and Case II if we agree that $\lambda_i^k = 0$ for Case I and that $\lambda_i^k$ is determined form (3.2)–(3.3) in Case II.

Equation (3.2) implies, by (2.5), that

$$\nabla f(y_{(i)}^{(k)}) = \lambda_i^k a^{(i)}. \tag{3.4}$$

Using (3.2), (2.3) and (3.4) for $k-1$, we get

$$\nabla f(x_{(i)}^{(k)}) = \nabla f(x_{(i-1)}^{(k)}) + (\lambda_i^{k-1} - \lambda_i^k) a^{(i)}, \tag{3.5}$$

which can also be written as

$$\lambda_i^k = \lambda_i^{k-1} + \Big\langle \frac{a^{(i)}}{\|a^{(i)}\|^2}, \nabla f(x_{(i-1)}^{(k)})\Big\rangle - \Big\langle \frac{a^{(i)}}{\|a^{(i)}\|^2}, \nabla f(x_{(i)}^{(k)})\Big\rangle. \tag{3.6}$$

This enables us to say that, in either Case I or Case II, the iterative step of Algorithm 2.1 can be described in equivalent form by

$$\nabla f(x_{(i)}^{(k)}) = \nabla f(x_{(i-1)}^{(k)}) + \beta_i^k a^{(i)}, \tag{3.7}$$

with

$$\beta_i^k \triangleq \min\Big(\lambda_i^{k-1}, \frac{\langle a^{(i)}, \nabla f(x_{(i)}^{(k)}) - \nabla f(x_{(i-1)}^{(k)})\rangle}{\|a^{(i)}\|^2}\Big). \tag{3.8}$$

Indeed, for $\beta_i^k = \lambda_i^{k-1}$ (3.7) is identical to (3.5) with $\lambda_i^k = 0$, and using (3.4) for $k-1$, we obtain $\nabla f(x_{(i)}^{(k)}) = \nabla f(z_{(i)}^{(k)})$, which is Case I. For $\beta_i^k \neq \lambda_i^{k-1}$, (3.7) is identical to (3.5) via (3.6). The fact that in Case II, $\lambda_i^k > 0$ guarantees, by (3.6), that $\beta_i^k$ will not take the value $\lambda_i^{k-1}$.

Finally, to complete the identification of (3.7)–(3.8) with Bregman's algorithm, we replace the double index $(k,i)$ by a single index $\nu$ through

$$\nu \triangleq (k-1)m + i - 1 \tag{3.9}$$

and define

$$u^{(\nu)} \triangleq x_{(i-1)}^{(k)}, \quad \pi_i^{(\nu)} \triangleq \lambda_i^{k-1}, \text{ and } c_\nu \triangleq \beta_i^k. \tag{3.10}$$

With these replacements it is clear that Algorithm 2.1 becomes identical with Algorithm A.1 of the Appendix and Theorem A.1 applies. ∎

In a similar manner it is possible to handle the polyhedral case with interval linear inequality constraints and show that in this case Algorithm 2.1 becomes identical with Algorithm 5.1 of Censor and Lent [11], which in the special case of orthogonal projections is Herman and Lent's [28] ART4 algorithm.

## 4. THE CASE OF $I$-PROJECTIONS

Dykstra devised in [22] an algorithm which finds the $I$-projection of a point onto the nonempty intersection of closed convex sets.

**Definition 4.1** *Let $\Delta_n \triangleq \{x \in \mathbf{R}_+^n \mid \sum_{j=1}^n x_j = 1\}$, and for all $x, y \in \Delta_n$ let*

$$I(x,y) \triangleq \sum_{j=1}^{n} x_j \log\Big(\frac{x_j}{y_j}\Big),$$

*where* log *stands for the natural logarithm and $0 \log 0 \triangleq 0$. For a closed and convex subset $\Omega \subseteq \Delta_n$ and a point $r \in \Delta_n$, the $I$-projection of $r$ onto $\Omega$ is defined as the point $r^* \in \Delta_n$ for which*

$$r^* = \arg\min_{r \in \Omega} I(x,r). \tag{4.1}$$

Csiszár [17], [18] studied such $I$-projections (see also Robertson, Wright and Dykstra [36]).

In the spirit of the original Dykstra algorithm ([21], [4], [26]) with orthogonal projections (2.6)–(2.7), Dykstra developed in [22] the following algorithm.

**Algorithm 4.1**

**Initialization:** *Set* $y_{(i)}^{(0)} \triangleq \underset{\sim}{1}$, $i = 1, 2, \ldots, m$, *where* $\underset{\sim}{1}$ *is the vector all the coordinates of which are equal to* 1, *and set* $x_{(m)}^{(0)} \triangleq r$, *the given point.*

**Iterative Step:** *Set* $x_{(0)}^{(k)} = x_{(m)}^{(k-1)}$ *and, for* $i = 1, 2, \ldots, m$, *calculate* $z_{(i)}^{(k)}$ *from*

$$(z_{(i)}^{(k)})_j = \frac{(x_{(i-1)}^{(k)})_j}{(y_{(i)}^{(k-1)})_j}, \qquad j = 1, 2, \ldots, n; \tag{4.2}$$

*then* $I$*-project* $z_{(i)}^{(k)}$ *onto* $C_i$, *i.e.,*

$$x_{(i)}^{(k)} = P_i^I(z_{(i)}^{(k)}), \tag{4.3}$$

*where* $P_i^I$ *is the* $I$*-projection onto the set* $C_i$. *Finally, update the "memory" vector by calculating*

$$(y_{(i)}^{(k)})_j = \frac{(z_{(i)}^{(k)})_j}{(P_i^I(z_{(i)}^{(k)}))_j}, \qquad j = 1, 2, \ldots, n. \tag{4.4}$$

To recognize the identity of this algorithm with Dykstra's $I$-projections algorithm one has only to identify $x_{(i)}^{(k)} \equiv p_{n,i}$ and $z_{(i)}^{(k)} \equiv s_{n,i}$ where $p_{n,i}$ and $s_{n,i}$ are the symbols used in [22]. Dykstra conjectures that Algorithm 4.1 always converges to the $I$-projection of $r$ onto the intersection $C \triangleq \bigcap_{i=1}^m C_i$, but he is only able to prove this by requiring an extra condition ([22], Theorem 2.1).

Algorithm 4.1 can be viewed as a special instance of Algorithm 2.1 by taking the function $f(x) \triangleq \sum_{j=1}^n x_j \log x_j - x_j$, and the zone $S \triangleq \text{int } \Delta_n$. Then $f \in \mathcal{B}(S)$, $(\nabla f(x))_j = \log x_j$, $j = 1, 2, \ldots, n$, and $D_f(x, y) = KL(x, y)$ for all $x, y \in \Delta_n$, where $KL(x, y)$ is the Kullback-Leibler distance

$$KL(x, y) = \sum_{j=1}^n \left( x_j \log \left( \frac{x_j}{y_j} \right) - x_j + y_j \right), \tag{4.5}$$

which coincides with $I(x, y)$ over $\Delta_n \times \Delta_n$; see, e.g., Teboulle [38], Examples 3.1. Therefore, the result of Dykstra [22] applies to Algorithm 2.1 in this case.

The function $f(x) \triangleq \sum_{j=1}^n x_j \log x_j = -\text{ ent}(x)$, where $\text{ent}(x) \triangleq -\sum_{j=1}^n x_j \log x_j$ is Shannon's entropy, is a Bregman function over the larger zone $S = \text{int } \mathbf{R}_+^n$ and $D_f(x, y) = KL(x, y)$, see, e.g., Censor et al. [8], Lemma 5, but these distance functions are not necessarily equal to $I(x, y)$ outside $\Delta_n \times \Delta_n$. Denoting $P_i^f$, in this particular case, by $P_i^{\text{ent}}$, we obtain from Algorithm 2.1 another Dykstra entropic algorithm which is identical with Algorithm 4.1 except for the replacement of $P_i^I$ by $P_i^{\text{ent}}$ in (4.3) and (4.4). This slightly more general case in also well covered by Algorithm 2.1 because if $\Omega \subseteq \Delta_n$ is a closed convex set and $r \in \text{int } \mathbf{R}_+^n$ but $r \notin \Delta_n$ then it is not difficult to verify that

$$P_\Omega^{\text{ent}}(r) = P_\Omega^{\text{ent}}\left(P_{\Delta_n}^{\text{ent}}(r)\right) = P_\Omega^I(r') = P_\Omega^{I'}(r), \tag{4.6}$$

where $r' = P^{\text{ent}}_{\Delta_n}(r) = r/\sum_{j=1}^{n} r_j$ and $P^{I'}_{\Omega}$ is the projection onto $\Omega$ with respect to $I'(x,y) = \sum_{j=1}^{n}\left(x_j \log\left(\frac{x_j}{y_j}\right)\right)$ defined on $\Delta_n \times \mathbf{R}^n_+$.


## ACKNOWLEDGEMENTS

We gratefully acknowledge useful discussions with Dan Butnariu and Yael Zepkowitz-Malachi. The work of Y. Censor was partially supported by NIH grant HL-28438 at the Medical Image Processing Group (MIPG), Department of Radiology, University of Pensylvania, Philadelphia, PA, USA. S. Reich was partially supported by the Fund for the Promotion of Research at the Technion and by the Technion VPR Fund — M. and M. L. Bank Mathematics Research Fund.


## REFERENCES


[1] H. H. Bauschke (1996), Projection Algorithms and Monotone Operators, Ph.D. thesis, Dept. of Mathematics and Statistics, Simon Fraser University, Burnaby, BC, Canada.

[2] H. H. Bauschke and J. M. Borwein (1994), Dykstra's alternating projection algorithm for two sets, *Journal of Approximation Theory* **79**, 418-443.

[3] H. H. Bauschke, and J. M. Borwein (1996), Legendre functions and the method of random Bregman projections, Technical Report, Centre for Experimental & Constructive Mathematics (CECM), Simon Fraser University, Burnaby, BC, Canada, *Journal of Convex Analysis*, to appear.

[4] J. P. Boyle and R. L. Dykstra (1986), A method for finding projections onto the intersection of convex sets in Hilbert spaces, in: *Advances in Order Restricted Statistical Inference, Lecture Notes in Statistics*, Springer-Verlag, Berlin, Germany, Volume **37**, pp. 28–47.

[5] L. M. Bregman (1967), 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**, 200–217.

[6] C. Byrne and Y. Censor (1997), Proximity function minimization and the convex feasibility problem for jointly convex Bregman distances, Technical Report, February 1997.

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

[8] Y. Censor, A. R. De Pierro, T. Elfving, G. T. Herman, and A. N. Iusem (1990), On iterative methods for linearly constrained entropy optimization, in: A. Wakulicz (Editor), *Numerical Analysis and Mathematical Modelling*, Banach Center Publications, Vol. **24**, pp. 145–163, PWN — Polish Scientific Publishers, Warsaw, Poland.

[9] Y. Censor and T. Elfving (1994) A multiprojection algorithm using Bregman projections in a product space, *Numerical Algorithms* **8**, 221–239.

[10] Y. Censor, A. N. Iusem and S. A. Zenios (1997) An interior point method with Bregman functions for the variational inequality problem with paramonotone operators, Technical Report, Revised: October 1995, *Mathematical Programming*, to appear.

[11] Y. Censor and A. Lent (1981) An iterative row-action method for interval convex programming, *Journal of Optimization Theory and Applications* **34**, 321–353.

[12] Y. Censor and S. Reich (1996), Iterations of paracontractions and firmly nonexpansive operators with applications to feasibility and optimization, *Optimization* **37**, 323–339.

[13] Y. Censor and S. A. Zenios (1992), Proximal minimization algorithm with $D$-functions, *Journal of Optimization Theory and Applications* **73**, 451–464.

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

[15] P. L. Combettes (1993), Signal recovery by best feasible approximation, *IEEE Transactions on Image Processing* **2**, 269–271.

[16] G. Crombez (1995), Finding projections onto the intersection of convex sets in Hilbert spaces, *Numerical Functional Analysis and Optimization* **16**, 637–652.

[17] I. Csiszár (1975), $I$-divergence geometry of probability distributions and minimization problems, *The Annals of Probability* **3**, 146–158.

[18] I. Csiszár (1991), Why least squares and maximum entropy? An axiomatic approach to inference for linear inverse problems, *The Annals of Statistics* **19**, 2032–2066.

[19] A. R. De Pierro and A. N. Iusem (1986) A relaxed version of Bregman's method for convex programming, *Journal of Optimization Theory and Applications* **51**, 421–440.

[20] F. Deutsch and H. Hundal (1994), The rate of convergence of Dykstra's cyclic projections algorithm: The polyhedral case, *Numerical Functional Analysis and Optimization* **15**, 537–565.

[21] R. L. Dykstra (1983), An algorithm for restricted least squares regression, *Journal of the American Statistical Association* **78**, 837–842.

[22] R. L. Dykstra (1985), An iterative procedure for obtaining $I$-projections onto the intersection of convex sets, *The Annals of Probability* **13**, 975–984.

[23] D. A. D'Esopo (1959), A convex programming procedure, *Naval Research Logistics Quarterly* **6**, 33–42.

[24] J. Eckstein (1993), Nonlinear proximal point algorithms using Bregman functions with applications to convex programming, *Mathematics of Operations Research* **18**, 202–226.

[25] N. Gaffke and R. Mathar (1989), A cyclic projection algorithm via duality, *Metrika* **36**, 29–54.

[26] S. P. Han (1988), A successive projection method, *Mathematical Programming* **40**, 1–14.

[27] S. P. Han and G. Lou (1988), A parallel algorithm for a class of convex programs, *SIAM Journal on Control and Optimization* **26**, 345–355.

[28] G. T. Herman and A. Lent (1978), A Familiy of iterative quadratic optimization algorithms for pairs of inequalities with application in diagnostic radiology, *Mathematical Programming Study* **9**, 15–29.

[29] C. Hildreth (1957), A quadratic programming procedure, *Naval Research Logistics Quarterly* **4**, 79–85. Erratum, ibid., p. 361.

[30] H. Hundal and F. Deutsch (1997), Two generalizations of Dykstra's cyclic projection algorithm *Mathematical Programming*, to appear.

[31] A. N. Iusem (1991), On dual convergence and the rate of primal convergence of Bregman's convex programming method, *SIAM Journal on Optimization*, **1**, 401–423.

[32] A. N. Iusem and A. R. De Pierro (1991), On the convergence of Han's method for convex programming with quadratic objective, *Mathematical Programming* **52**, 265–284.

[33] K. C. Kiwiel (1997), Proximal minimization methods with generalized Bregman functions, *SIAM Journal on Control and Optimization* **35**.

[34] A. Lent and Y. Censor (1980), Extensions of Hildreth's row-action method for quadratic programming, *SIAM Journal on Control and Optimization* **18**, 444–454.

[35] G. Pierra (1984), Decomposition through formalization in a product space, *Mathematical Programming* **28**, 96–115.

[36] T. Robertson, F. T. Wright and R. L. Dykstra (1988), *Order Restricted Statistical Inference*, John Wiley & Sons, Chichester, UK.

[37] R. T. Rockafellar (1970), *Convex Analysis*, Princeton University Press, Princeton, NJ, USA.

[38] M. Teboulle (1992), Entropic proximal mappings with applications to nonlinear programming, *Mathematics of Operations Research* **17**, 670–690.

## APPENDIX: BREGMAN FUNCTIONS, DISTANCES AND PROJECTIONS

We recall here some basic facts on Bregman functions, $D_f$-functions, and projections. This material has its origin in Bregman's paper [5], Censor and Lent [11], and futher developments which appear in the works of Bauschke and Borwein [3], Censor and Zenios [13], Censor, Iusem and Zenios [10], De Pierro and Iusem [19], Eckstein [24], Iusem [31], Teboulle [38], and others (see Censor and Zenios [14]). Let $S$ be a nonempty, open, convex set such that its closure $\overline{S} \subseteq \Lambda$, where $\Lambda$ is the domain of a function $f: \Lambda \subseteq \mathbf{R}^n \to \mathbf{R}$. Assume that $f(x)$ has continuous first partial derivatives at every $x \in S$ and denote by $\nabla f(x)$ its gradient at $x$. From $f(x)$, construct the $D_f$-function $D_f: \overline{S} \times S \subseteq \mathbf{R}^{2n} \to \mathbf{R}$ by

$$D_f(x,z) \stackrel{\Delta}{=} f(x) - f(z) - \langle \nabla f(z), x - z \rangle, \tag{A.1}$$

where $\langle \cdot, \cdot \rangle$ is the standard inner product in $\mathbf{R}^n$. Denote, for $\alpha \in \mathbf{R}$, the partial level sets of $D_f(x,z)$ by

$$L_1^f(\alpha, z) = \{x \in \overline{S} \mid D_f(x,z) \leq \alpha\}, \tag{A.2}$$

$$L_2^f(x, \alpha) = \{z \in S \mid D_f(x,z) \leq \alpha\}. \tag{A.3}$$

**Definition A.1** *A function $f: \Lambda \subseteq \mathbf{R}^n \to \mathbf{R}$ is called a Bregman function if there exists a nonempty, open, convex set $S$, such that $\overline{S} \subseteq \Lambda$ and the following hold:*

*(i) $f$ is continuously differentiable on $S$;*

*(ii) $f$ is strictly convex on $\overline{S}$;*

*(iii) $f$ is continuous on $\overline{S}$;*

*(iv) for every $\alpha \in \mathbf{R}$, the partial level sets $L_1^f(\alpha, z)$ and $L_2^f(x, \alpha)$ are bounded, for every $z \in S$ and every $x \in \overline{S}$, respectively;*

*(v) if $z^{(k)} \in S$, for all $k \geq 0$, and $\lim_{k\to\infty} z^{(k)} = z^*$, then $\lim_{k\to\infty} D_f(z^*, z^{(k)}) = 0$;*

*(vi) if $x^{(k)} \in \overline{S}$ and $z^{(k)} \in S$, for all $k \geq 0$, $\lim_{k\to\infty} D_f(x^{(k)}, z^{(k)}) = 0$, $\lim_{k\to\infty} z^{(k)} = z^{(*)}$ and $\{x^{(k)}\}$ is bounded, then $\lim_{k\to\infty} x^{(k)} = z^*$.*

The set $S$ is called the *zone* and the function $f$ is a *Bregman function* with respect to $S$ and we denote these facts by $f \in \mathcal{B}(S)$. The reason for collecting the conditions $(i)$–$(vi)$ under one heading is that these are precisely the conditions needed to ensure the applicability of the primal-dual optimization algorithm of Bregman [5] and the algorithm for interval convex programming of Censor and Lent [11], when the function $D_f(x, z)$ has the form (A.1).

¿From the strict convexity assumption on $f$ it follows that, for $x \in \overline{S}$, $z \in S$,

$$D_f(x, z) \geq 0, \quad \text{and} \quad D_f(x, z) = 0 \iff x = z, \tag{A.4}$$

which makes $D_f$ a "measure of distance on $S$" according to the terminology of Csiszár [18], although, for a general Bregman function, $D_f$ is not a distance function. We remark however that $D_f$ equals the vertical distance, at the point $x$, between $(x, f(x))$ and the tangent hyperplane to epi $f$ at the point $(z, f(z))$, cf. Eckstein [24]. When $f(x) = \frac{1}{2}\|x\|^2$ one gets $D_f(x, z) = \frac{1}{2}\|x - z\|^2$.

With the aid of $D_f$ one defines projections as follows.

**Definition A.2** *Given a Bregman function with zone $S$, a set $\Omega \subseteq \mathbf{R}^n$ and some $z \in S$, the projection (called D-projection in [7]) of $z$ onto $\Omega$, denoted by $P^f_\Omega(z)$, is the point*

$$z^* = P^f_\Omega(z) \triangleq \arg \min_{x \in \Omega \cap \overline{S}} D_f(x, z). \tag{A.5}$$

Existence and uniqueness of such a projection when $\Omega$ is closed and convex and $\Omega \cap \overline{S} \neq \emptyset$ (which we hereafter assume) are established in Lemma 2.2 of Censor and Lent [11]. We will also need the following concept.

If $f(x) = \frac{1}{2}\|x\|^2$ and $S = \mathbf{R}^n$, then $P^f_\Omega(z)$ is the orthogonal projection. Further results on Bregman projections and additional examples can be found in Censor and Reich [12], De Pierro and Iusem [19], Eckstein [24], Teboulle [38], Censor and Elfving [9], Byrne and Censor [6] and Kiwiel [33].

The following basic result can be considered an extension of the Pythagorean theorem.

**Lemma A.1** *Let a function $f \in \mathcal{B}(S)$ and a closed convex set $\Omega \subseteq \mathbf{R}^n$ be given. Assume that $\Omega \cap \overline{S} \neq \emptyset$ and that $y \in S$ implies $P^f_\Omega(y) \in S$. If $y \in \Omega \cap \overline{S}$, then for any $y \in S$ the inequality*

$$D_f(P^f_\Omega(y), y) \leq D_f(z, y) - D_f(z, P^f_\Omega(y)) \tag{A.6}$$

*holds.*

**Proof:** This is a specialization of Lemma 1 of Bregman [5] for the case of $D_f$-functions; see Example 2 on p. 205 of Bregman [5]. ∎

Bregman's algorithm for the linearly constrained minimization of a Bregman function $f \in \mathcal{B}(S)$ is designed to solve the problem

$$\text{Minimize } f(x) \tag{A.7}$$

$$\text{s.t. } \langle a^{(i)}, x\rangle \leq b_i, \qquad i \in I \triangleq \{1, 2, \ldots, m\}, \tag{A.8}$$

$$x \in \overline{S}. \tag{A.9}$$

Let $H_i \triangleq \{x \mid \langle a^{(i)}, x\rangle = b_i\}$ and $C_i \triangleq \{x \mid \langle a^{(i)}, x\rangle \leq b_i\}$; denote also $H = \cap_{i=1}^m H_i$, $C = \cap_{i=1}^m C_i$, and assume that $C \cap \overline{S} \neq \emptyset$. $A = (a_{ij})$ is the $m \times n$ matrix the $i$-th row of which is $(a^{(i)})^T$, and $b = (b_i) \in \mathbf{R}^m$. Assume that all $a^{(i)} \neq 0$. Assume also that $f \in \mathcal{B}(S)$ is strongly zone consistent with respect to every $H_i$. This means ([11], Definition 3.1) that $y \in S$ implies $P^f_{\hat{H}_i}(y) \in S$ for $\hat{H}_i = H_i$, and for any other hyperplane $\hat{H}_i$ parallel to $H_i$ and lying between $y$ and $H_i$. Define the following sets:

$$Z \triangleq \{x \in S \mid \exists \pi \in \mathbf{R}^m \text{ such that } \nabla f(x) = -A^T\pi\}, \tag{A.10}$$

$$Z_0 \triangleq \{x \in S \mid \exists \pi \in \mathbf{R}^m_+ \text{ such that } \nabla f(x) = -A^T\pi\}, \tag{A.11}$$

which are assumed to be nonempty.

The following algorithm can be found in [5], [11], [14].

**Algorithm A.1**

**Initialization:** $u^{(0)} \in Z_0$ *is arbitrary, and* $\pi^{(0)}$ *is such that*

$$\nabla f(u^{(0)}) = -A^T\pi^{(0)}. \tag{A.12}$$

**Iterative step:** *Given* $u^{(\nu)}$ *and* $\pi^{(\nu)}$*, calculate* $u^{(\nu+1)}$*, and* $\pi^{(\nu+1)}$ *from*

$$\begin{aligned} \nabla f(u^{(\nu+1)}) &= \nabla f(u^{(\nu)}) + c_\nu a^{(i(\nu))}, \\ \pi^{(\nu+1)} &= \pi^{(\nu)} - c_\nu e^{(i(\nu))}, \\ c_\nu &\triangleq \min(\pi^{(\nu)}_{i(\nu)}, \theta_\nu), \end{aligned} \tag{A.13}$$

*where* $\theta_\nu$ *is the parameter associated with the generalized projection of* $u^{(\nu)}$ *onto* $H_{i(v)}$*. We assume throughout that the representation of every hyperplane is fixed during the whole iteration process, so that the values of* $\theta_\nu$ *are well defined.*

**Control:** *The sequence* $\{i(\nu)\}_{\nu=0}^\infty$ *is cyclic on the index set* $I$*.*

Recall that $\theta_\nu$, the parameter associated with the generalized projection $\hat{u}^{(\nu+1)}$ of $u^{(\nu)}$ onto $H_{i(\nu)}$, is obtained by solving the system

$$\begin{aligned} \nabla f(\hat{u}^{(\nu+1)}) &= \nabla f(u^{(\nu)}) + \theta_\nu a^{(i(\nu))}, \\ \langle a^{(i(\nu))}, \hat{u}^{(\nu+1)}\rangle &= b_{i(\nu)}. \end{aligned} \tag{A.14}$$

Here we use $e^{(t)}$ to denote the $t$-th standard basis vector with a 1 in its $t$-th coordinate and zeros elsewhere. The cyclicality means that $i(\nu) = \nu \pmod{m} + 1$ (a more general control sequence called *almost cyclic* may also be employed, see [11]).

The following convergence theorem ([5], [11], [14]) holds.

**Theorem A.1** *Assume the following:*

*(i)* $f \in \mathcal{B}(S)$*,*

*(ii)* $f$ *is strongly zone consistent with respect to each* $H_i$*,* $i \in I$*,*

*(iii)* $\{i(\nu)\}_{\nu=0}^{\infty}$ *is cyclic,*

*(iv)* $Z_0 \neq \emptyset$*.*

*Then, any sequence* $\{x^{(\nu)}\}$ *produced by Algorithm A.1 converges to the point* $x^*$*, which is the solution of (A.7)–(A.9).*