Sequential Unconstrained Minimization: A Survey[*]

Charles L. Byrne[†]

February 21, 2013

Abstract

The problem is to minimize a function $f : X \to (-\infty, \infty]$, over a non-empty subset C of X, where X is an arbitrary set.

At the kth step of a sequential unconstrained minimization algorithm we minimize a function $G_k(x) = f(x) + g_k(x)$ to get x^k. The auxiliary functions g_k are typically selected to impose the constraint that x be in C, or to penalize any violation of that constraint. The g_k may also be employed to permit the x^k to be calculated in closed form. Barrier-function and penalty-function algorithms are the most well known sequential unconstrained minimization methods. Our main objective is to find $g_k(x)$ so that the infinite sequence $\{x^k\}$ generated by our algorithm converges to a solution of the problem; this, of course, requires some topology on the set X. Failing that, we want the sequence $\{f(x^k)\}$ to converge to d, where

$$d = \inf\{f(x)|x \in C\} \geq -\infty,$$

or, at the very least, for the sequence $\{f(x^k)\}$ to be non-increasing.

A sequential unconstrained minimization algorithm is in the SUMMA class if, for all $x \in X$,

$$G_k(x) - G_k(x^k) \geq g_{k+1}(x) \geq 0.$$

If $\{x^k\}$ is generated by an algorithm in the SUMMA class, then the sequence $\{f(x^k)\}$ converges to d.

A wide variety of iterative methods, including barrier-function and penalty-function methods, can be shown to be members of the SUMMA class. Other members of the SUMMA class include proximal minimization algorithms using Bregman distances, forward-backward splitting methods, the CQ algorithm for the split feasibility problem, the simultaneous MART algorithm, alternating minimization methods, and the expectation maximization maximum likelihood (EM) algorithms.

[*] The latest version is available at http://faculty.uml.edu/cbyrne/cbyrne.html

[†] Charles_Byrne@uml.edu, Department of Mathematical Sciences, University of Massachusetts Lowell, Lowell, MA 01854

1 Introduction

Minimizing a real-valued function subject to constraints on the independent variable can be a difficult problem to solve; typically, iterative algorithms are required. Sequential unconstrained minimization techniques replace the single constrained optimization problem with an infinite sequence of unconstrained minimization problems, each one easier to solve than the original problem. In the best of cases, the sequence of minimizers will converge to a solution of the original constrained minimization problem, or, failing that, their function values will converge to the constrained minimum, or, at least, will be non-decreasing.

Even when there are no constraints, the problem of minimizing a real-valued function may require iteration; the formalism of sequential unconstrained minimization techniques can be useful in deriving such iterative algorithms, as well as in proving convergence.

2 Preliminaries

In this section we look briefly at several of the topics to be discussed in more detail later.

2.1 The Basic Problem

Let X be an arbitrary non-empty set. We are concerned here with the minimization of a real-valued function $f : X \to \mathbb{R}$, possibly subject to constraints on the independent variable, and with the use of iterative sequential unconstrained minimization algorithms to solve such problems.

The primary problem in convex programming is to minimize a convex function $f : \mathbb{R}^J \to \mathbb{R}$, subject to the constraints $h_i(x) \leq 0$, where, for $i = 1, 2, ..., I$, h_i is convex. If the problem is super-consistent, that is, there are points x with $h_i(x) < 0$, for all i, then the Karush-Kuhn-Tucker necessary and sufficient conditions for x^* to solve the problem are that there are $\lambda_i^* \geq 0$, with $\lambda_i^* h_i(x^*) = 0$, for all i, and x^* minimizes

$$f(x) + \sum_{i=1}^{I} \lambda_i^* h_i(x). \tag{2.1}$$

Of course, finding the λ_i^* is part of the problem, since we are not given the λ_i^* initially. Some terms in Equation (2.1) are non-negative whenever x fails to satisfy all the constraints, so the summation acts as a kind of penalty function, penalizing violation of the constraints. Penalty-function methods for constrained minimization exploit this idea.

2.2 Penalty-Function Methods

Suppose that our goal is to minimize a function $f : \mathbb{R}^J \to \mathbb{R}$, subject to the constraint that $x \in C$, where C is a non-empty closed subset of $\mathbb{R}^J$. We select a non-negative function $p : \mathbb{R}^J \to \mathbb{R}$ with the property that $p(x) = 0$ if and only if x is in C and then, for each positive integer k, we minimize

$$G_k(x) = f(x) + kp(x), \tag{2.2}$$

to get x^k. We then want the sequence $\{x^k\}$ to converge to some $x^* \in C$ that solves the original problem. In order for this iterative algorithm to be useful, each x^k should be relatively easy to calculate.

If, for example, we should select $p(x) = +\infty$ for x not in C and $p(x) = 0$ for x in C, then minimizing $G_k(x)$ is equivalent to the original problem and we have achieved nothing.

Suppose that we want to minimize the function $f(x) = (x+1)^2$, subject to $x \geq 0$. Let us select $p(x) = x^2$, for $x \leq 0$, and $p(x) = 0$ otherwise. Then $x^k = \frac{-1}{k+1}$, which converges to the right answer, $x^* = 0$, as $k \to \infty$.

2.3 Barrier-Function Methods

Suppose now that $b : C \to \mathbb{R}$ is a barrier function for C, that is, b has the property that $b(x) \to +\infty$ as x approaches the boundary of C. At the kth step of the iteration we minimize

$$G_k(x) = f(x) + \frac{1}{k}b(x) \tag{2.3}$$

to get x^k. Then each x^k is in C. We want the sequence $\{x^k\}$ to converge to some x^* in C that solves the original problem.

Suppose that we want to minimize the function $f(x) = f(x_1, x_2) = x_1^2 + x_2^2$, subject to the constraint that $x_1 + x_2 \geq 1$. The constraint is then written $g(x_1, x_2) = 1 - (x_1 + x_2) \leq 0$. We use the logarithmic barrier function $b(x) = -\log(x_1 + x_2 - 1)$. For each positive integer k, the vector $x^k = (x_1^k, x_2^k)$ minimizing the function

$$G_k(x) = x_1^2 + x_2^2 - \frac{1}{k}\log(x_1 + x_2 - 1) = f(x) + \frac{1}{k}b(x)$$

has entries

$$x_1^k = x_2^k = \frac{1}{4} + \frac{1}{4}\sqrt{1 + \frac{4}{k}}.$$

Notice that $x_1^k + x_2^k > 1$, so each x^k satisfies the constraint. As $k \to +\infty$, x^k converges to $(\frac{1}{2}, \frac{1}{2})$, which is the solution to the original problem. The use of the logarithmic barrier function forces $x_1 + x_2 - 1$ to be positive, thereby enforcing the constraint on $x = (x_1, x_2)$.

Penalty-function methods are called exterior-point methods since, typically, none of the x^k satisfies the constraints. Barrier-function methods are called interior-point methods because each x^k satisfies the constraints.

2.4 Sequential Unconstrained Minimization

Penalty-function and barrier-function methods are the most well known of sequential unconstrained minimization techniques. At the kth step of a sequential unconstrained minimization algorithm we minimize

$$G_k(x) = f(x) + g_k(x) \tag{2.4}$$

to obtain x^k, where the added function $g_k(x)$ is chosen by us. The goal is to have the sequence $\{x^k\}$ converge to a solution x^* of the original problem. Failing that, we want the function values $f(x^k)$ to converge to the constrained minimum.

As is clear with penalty-function and barrier-function methods, the added functions $g_k(x)$ may serve to incorporate the constraints. There may also be other reasons for selecting the g_k, however.

2.5 Projected Gradient Descent

The problem now is to minimize $f : \mathbb{R}^J \to \mathbb{R}$, over the closed, non-empty convex set C, where f is convex and differentiable on $\mathbb{R}^J$. In most cases there is no closed-form algebraic solution and we need an iterative method. We can use the idea of sequential unconstrained minimization to derive an iterative algorithm to solve this problem. As we shall see, although we still call it a sequential unconstrained minimization method, it is not unconstrained; our goal here is to derive a closed-form iterative method, not to incorporate the constraint in the objective function being minimized.

The Bregman distance $D_f(x, y)$ associated with the function f is

$$D_f(x, y) = f(x) - f(y) - \langle \nabla f(y), x - y \rangle. \tag{2.5}$$

Since f is convex, we know that $D_f(x, y)$ is non-negative, for all x and y.

At the kth step we minimize

$$G_k(x) = f(x) + \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_f(x, x^{k-1}), \tag{2.6}$$

over $x \in C$, obtaining

$$x^k = P_C(x^{k-1} - \gamma\nabla f(x^{k-1})); \tag{2.7}$$

here P_C denotes the orthogonal projection onto C. This is the projected gradient descent algorithm. For convergence we must require that f have certain additional properties to be discussed later. Note that the added function

$$g_k(x) = \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_f(x, x^{k-1}) \tag{2.8}$$

is unrelated to the set C, so is not used here to incorporate the constraint; it is used to provide a closed-form iterative scheme.

When $C = \mathbb{R}^J$ we have no constraint and the problem is simply to minimize f. Then the iterative algorithm becomes

$$x^k = x^{k-1} - \gamma\nabla f(x^{k-1}); \tag{2.9}$$

this is the gradient descent algorithm.

2.6 Relaxed Gradient Descent

In the gradient descent method we move away from the current x^{k-1} by the vector $\gamma\nabla f(x^{k-1})$. In relaxed gradient descent, the magnitude of the movement is reduced by α, where $\alpha \in (0,1)$. Such relaxation methods are sometimes used to accelerate convergence. The relaxed gradient descent method can also be formulated as a sequential unconstrained minimization method.

At the kth step we minimize

$$G_k(x) = \alpha[f(x) + \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_f(x, x^{k-1})] + \frac{1-\alpha}{2\gamma}\|x - x^{k-1}\|^2, \tag{2.10}$$

obtaining

$$x^k = (x^{k-1} - \alpha\gamma\nabla f(x^{k-1})). \tag{2.11}$$

2.7 Regularized Gradient Descent

In many applications the function to be minimized involves measured data, which is typically noisy, as well as some less than perfect model of how the measured data was obtained. In such cases, we may not want to minimize $f(x)$ exactly. In regularization methods we add to $f(x)$ another function that is designed to reduce sensitivity to noise and model error.

For example, suppose that we want to minimize

$$\alpha f(x) + \frac{1-\alpha}{2}\|x-p\|^2, \tag{2.12}$$

where p is chosen a priori. The regularized gradient descent algorithm for this problem can be put in the framework of a sequential unconstrained minimization problem.

At the kth step we minimize

$$G_k(x) = \alpha[f(x) + \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_f(x, x^{k-1})] + \frac{1-\alpha}{2\gamma}\|x-p\|^2, \tag{2.13}$$

obtaining

$$x^k = \alpha(x^{k-1} - \gamma\nabla f(x^{k-1})) + (1-\alpha)p. \tag{2.14}$$

If we select $p = 0$ the iterative step becomes

$$x^k = \alpha(x^{k-1} - \gamma\nabla f(x^{k-1})). \tag{2.15}$$

2.8 Proximal Minimization

Let $f : \mathbb{R}^J \to (-\infty, +\infty]$ be a closed, proper, and convex function. Let h be a closed proper convex function, with effective domain D, that is differentiable on the nonempty open convex set int D. Assume that $f(x)$ is finite on $C = \overline{D}$ and attains its minimum value on C at $\hat{x}$. The corresponding *Bregman distance* $D_h(x, z)$ is defined for x in D and z in int D by

$$D_h(x, z) = h(x) - h(z) - \langle \nabla h(z), x - z\rangle. \tag{2.16}$$

Note that $D_h(x, z) \geq 0$ always. If h is essentially strictly convex, then $D_h(x, z) = 0$ implies that $x = z$. Our objective is to minimize $f(x)$ over x in $C = \overline{D}$.

At the kth step of the *proximal minimization algorithm* (PMA) [13], we minimize the function

$$G_k(x) = f(x) + D_h(x, x^{k-1}), \tag{2.17}$$

to get x^k. The function

$$g_k(x) = D_h(x, x^{k-1}) \tag{2.18}$$

is nonnegative and $g_k(x^{k-1}) = 0$. We assume that each x^k lies in int D.

We note that the proximal minimization framework has already been used several times in the previous discussion, when we added

$$g_k(x) = \frac{1}{2\gamma}\|x - x^{k-1}\|^2 - D_f(x, x^{k-1}) = D_h(x, x^{k-1}),$$

for

$$h(x) = \frac{1}{2\gamma}\|x\|^2 - f(x).$$

As we shall see, in the proximal minimization approach we have the inequality

$$G_k(x) - G_k(x^k) \geq g_{k+1}(x) \geq 0, \tag{2.19}$$

for all x. The SUMMA class of sequential unconstrained minimization approaches are those for which this inequality holds. All of the methods discussed so far fall into the SUMMA class.

2.9 Majorization Minimization

Majorization minimization (MM) is a technique for converting a hard optimization problem into a sequence of simpler ones [45, 6, 37]. The MM method requires that we majorize the objective function $f(x)$ with $g(x|y)$, such that $g(x|y) \geq f(x)$, for all x, and $g(y|y) = f(y)$. At the kth step of the iterative algorithm we minimize the function $g(x|x^{k-1})$ to get x^k. Said another way, we minimize

$$f(x) + [g(x|x^{k-1}) - f(x)] = f(x) + h(x|x^{k-1}), \tag{2.20}$$

where, for each y, $h(x|y) \geq 0$ for all x, and $h(y|y) = 0$. The MM method fits into the sequential unconstrained minimization format when we set $G_k(x) = g(x|x^{k-1})$. Now we have $g_k(x) = h(x|x^{k-1})$, so that $g_k(x) \geq 0$ and $g_k(x^{k-1}) = 0$; it then follows that the sequence $\{f(x^k)\}$ is non-increasing.

2.10 A Convergence Theorem

So far, we haven't discussed the restrictions necessary to prove convergence of these iterative algorithms. The framework of sequential unconstrained minimization can be helpful in this regard, as we illustrate now.

We say that the gradient operator ∇f is L-Lipschitz continuous if, for all x and y, we have

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \tag{2.21}$$

The following theorem concerns convergence of the gradient descent algorithm with iterative step given by Equation (2.9).

Theorem 2.1 *Let $f : \mathbb{R}^J \to \mathbb{R}$ be differentiable, with L-Lipschitz continuous gradient. For γ in the interval $(0, \frac{1}{L})$ the sequence $\{x^k\}$ given by Equation (2.9) converges to a minimizer of f, whenever minimizers exist.*

Proof:

The added function

$$g_k(x) = \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_f(x, x^{k-1}) \tag{2.22}$$

can be rewritten as

$$g_k(x) = D_h(x, x^{k-1}), \tag{2.23}$$

where

$$h(x) = \frac{1}{2\gamma}\|x\|_2^2 - f(x). \tag{2.24}$$

Therefore, $g_k(x) \geq 0$ whenever $h(x)$ is a convex function.

We know that $h(x)$ is convex if and only if

$$\langle \nabla h(x) - \nabla h(y), x - y \rangle \geq 0, \tag{2.25}$$

for all x and y. This is equivalent to

$$\frac{1}{\gamma}\|x - y\|_2^2 - \langle \nabla f(x) - \nabla f(y), x - y \rangle \geq 0. \tag{2.26}$$

Since ∇f is L-Lipschitz, the inequality (2.26) holds whenever $0 < \gamma < \frac{1}{L}$.

A relatively simple calculation shows that

$$G_k(x) - G_k(x^k) = \frac{1}{2\gamma}\|x - x^k\|_2^2 + \frac{1}{\gamma}\langle x^k - (x^{k-1} - \gamma \nabla f(x^{k-1})), x - x^k \rangle. \tag{2.27}$$

From Equation (2.9) it follows that

$$G_k(x) - G_k(x^k) \geq \frac{1}{2\gamma}\|x - x^k\|_2^2, \tag{2.28}$$

for all $x \in C$, so that

$$G_k(x) - G_k(x^k) \geq \frac{1}{2\gamma}\|x - x^k\|_2^2 - D_f(x, x^k) = g_{k+1}(x). \tag{2.29}$$

Now let $\hat{x}$ minimize $f(x)$ over all x. Then

$$G_k(\hat{x}) - G_k(x^k) = f(\hat{x}) + g_k(\hat{x}) - f(x^k) - g_k(x^k)$$

$$\leq f(\hat{x}) + G_{k-1}(\hat{x}) - G_{k-1}(x^{k-1}) - f(x^k) - g_k(x^k),$$

so that

$$\Big(G_{k-1}(\hat{x}) - G_{k-1}(x^{k-1})\Big) - \Big(G_k(\hat{x}) - G_k(x^k)\Big) \geq f(x^k) - f(\hat{x}) + g_k(x^k) \geq 0.$$

Therefore, the sequence $\{G_k(\hat{x}) - G_k(x^k)\}$ is decreasing and the sequences $\{g_k(x^k)\}$ and $\{f(x^k) - f(\hat{x})\}$ converge to zero.

From

$$G_k(\hat{x}) - G_k(x^k) \geq \frac{1}{2\gamma}\|\hat{x} - x^k\|_2^2,$$

it follows that the sequence $\{x^k\}$ is bounded. Let $\{x^{k_n}\}$ converge to $x^* \in C$ with $\{x^{k_n+1}\}$ converging to $x^{**} \in C$; we then have $f(x^*) = f(x^{**}) = f(\hat{x})$.

Replacing the generic $\hat{x}$ with x^{**}, we find that $\{G_{k_n+1}(x^{**}) - G_{k_n+1}(x^{k_n+1})\}$ is decreasing. By Equation (2.27), this subsequence converges to zero; therefore, the entire sequence $\{G_k(x^{**}) - G_k(x^k)\}$ converges to zero. From the inequality in (2.28), we conclude that the sequence $\{\|x^{**} - x^k\|_2^2\}$ converges to zero, and so $\{x^k\}$ converges to x^{**}. This completes the proof of the theorem. ▮

3 The SUMMA Class

The problem is to minimize a function $f : X \to (-\infty, \infty]$, over a subset P of X, where X is an arbitrary non-empty set. As we mentioned previously, a sequential unconstrained minimization technique is in the SUMMA class if each x^k in the iterative sequence can be obtained by minimizing

$$G_k(x) = f(x) + g_k(x) \tag{3.1}$$

over $x \in P$, where the $g_k(x)$ can be chosen so that the inequality in (2.19) is satisfied. In practice, of course, this minimization may need to be performed iteratively; we shall not address this issue here, and shall assume that x^k can be computed. We make the following additional assumptions.

Assumption 1: The functions $g_k(x)$ are finite-valued on the subset P.

Assumption 2: The functions $g_k(x)$ satisfy the inequality in (2.19), for $k = 1, 2, ...$ and all $x \in P$. Consequently,

$$g_{k+1}(x^k) = 0.$$

Assumption 3: There is a real number α with

$$\alpha \leq f(x),$$

for all x in X.

Assumption 4: Each x^k is in P.

Using these assumptions, we can conclude several things about the sequence $\{x^k\}$.

Proposition 3.1 *The sequence $\{f(x^k)\}$ is decreasing, and the sequence $\{g_k(x^k)\}$ converges to zero.*

Proof: We have

$$f(x^{k+1}) + g_{k+1}(x^{k+1}) = G_{k+1}(x^{k+1}) \leq G_{k+1}(x^k) = f(x^k) + g_{k+1}(x^k) = f(x^k).$$

Therefore,

$$f(x^k) - f(x^{k+1}) \geq g_{k+1}(x^{k+1}) \geq 0.$$

Since the sequence $\{f(x^k)\}$ is decreasing and bounded below by d, the difference sequence must converge to zero. Therefore, the sequence $\{g_k(x^k)\}$ converges to zero. ∎

Let

$$d = \inf\{f(x)|x \in P\} \geq -\infty.$$

Then we have the following theorem.

Theorem 3.1 *The sequence $\{f(x^k)\}$ converges to d.*

Proof: Suppose that there is $d^* > d$ with

$$f(x^k) \geq d^*,$$

for all k. Then there is z in P with

$$f(x^k) \geq d^* > f(z) \geq d,$$

for all k. From

$$g_{k+1}(z) \leq G_k(z) - G_k(x^k),$$

we have

$$g_k(z) - g_{k+1}(z) \geq f(x^k) + g_k(x^k) - f(z) \geq f(x^k) - f(z) \geq d^* - f(z) > 0,$$

for all k. This says that the nonnegative sequence $\{g_k(z)\}$ is decreasing, but that successive differences remain bounded away from zero, which cannot happen. ▮

Definition 3.1 *Let X be a complete metric space. A real-valued function $p(x)$ on X has* compact level sets *if, for all real γ, the level set $\{x|p(x) \leq \gamma\}$ is compact.*

Theorem 3.2 *Let X be a complete metric space, $f(x)$ be a continuous function, $d > -\infty$, and the restriction of $f(x)$ to x in P have compact level sets. Then the sequence $\{x^k\}$ is bounded and has convergent subsequences. Furthermore, $f(x^*) = d$, for any subsequential limit point $x^* \in X$. If $\hat{x}$ is the unique minimizer of $f(x)$ for $x \in P$, then $x^* = \hat{x}$ and $\{x^k\} \to \hat{x}$.*

Proof: From the previous theorem we have $f(x^*) = d$, for all subsequential limit points x^*. But, by uniqueness, $x^* = \hat{x}$, and so $\{x^k\} \to \hat{x}$. ▮

Corollary 3.1 *Let $C \subseteq \mathbb{R}^J$ be closed and convex. Let $f(x) : \mathbb{R}^J \to \mathbb{R}$ be closed, proper and convex. If $\hat{x}$ is the unique minimizer of $f(x)$ over $x \in C$, the sequence $\{x^k\}$ converges to $\hat{x}$.*

Proof: Let $\iota_C(x)$ be the indicator function of the set C, that is, $\iota_C(x) = 0$, for all x in C, and $\iota_C(x) = +\infty$, otherwise. Then the function $g(x) = f(x) + \iota_C(x)$ is closed, proper and convex. If $\hat{x}$ is unique, then we have

$$\{x|f(x) + \iota_C(x) \leq f(\hat{x})\} = \{\hat{x}\}.$$

Therefore, one of the level sets of $g(x)$ is bounded and nonempty. It follows from Corollary 8.7.1 of [47] that every level set of $g(x)$ is bounded, so that the sequence $\{x^k\}$ is bounded. ▮

If $\hat{x}$ is not unique, we may still be able to prove convergence of the sequence $\{x^k\}$, for particular cases of SUMMA, as we shall see shortly.

4 Barrier-function Methods

Let $b(x) : \mathbb{R}^J \to (-\infty, +\infty]$ be continuous, with effective domain the set

$$D = \{x | \, b(x) < +\infty\}.$$

The goal is to minimize the objective function $f(x)$, over x in C, the closure of D. We assume that there is $\hat{x} \in C$ with $f(\hat{x}) \leq f(x)$, for all x in C.

In the barrier-function method, we minimize

$$f(x) + \frac{1}{k} b(x) \tag{4.1}$$

over x in D to get x^k. Each x^k lies within D, so the method is an interior-point algorithm. If the sequence $\{x^k\}$ converges, the limit vector x^* will be in C and $f(x^*) = f(\hat{x})$.

Barrier functions typically have the property that $b(x) \to +\infty$ as x approaches the boundary of D, so not only is x^k prevented from leaving D, it is discouraged from approaching the boundary.

4.1 Examples of Barrier Functions

Consider the convex programming (CP) problem of minimizing the convex function $f : \mathbb{R}^J \to \mathbb{R}$, subject to $g_i(x) \leq 0$, where each $g_i : \mathbb{R}^J \to \mathbb{R}$ is convex, for $i = 1, ..., I$. Let $D = \{x | g_i(x) < 0, i = 1, ..., I\}$; then D is open. We consider two barrier functions appropriate for this problem.

4.1.1 The Logarithmic Barrier Function

A suitable barrier function is the *logarithmic barrier function*

$$b(x) = \Big(- \sum_{i=1}^{I} \log(-g_i(x)) \Big). \tag{4.2}$$

The function $-\log(-g_i(x))$ is defined only for those x in D, and is positive for $g_i(x) > -1$. If $g_i(x)$ is near zero, then so is $-g_i(x)$ and $b(x)$ will be large.

4.1.2 The Inverse Barrier Function

Another suitable barrier function is the *inverse barrier function*

$$b(x) = \sum_{i=1}^{I} \frac{-1}{g_i(x)}, \tag{4.3}$$

defined for those x in D.

In both examples, when k is small, the minimization pays more attention to $b(x)$, and less to $f(x)$, forcing the $g_i(x)$ to be large negative numbers. But, as k grows larger, more attention is paid to minimizing $f(x)$ and the $g_i(x)$ are allowed to be smaller negative numbers. By letting $k \to \infty$, we obtain an iterative method for solving the constrained minimization problem.

Barrier-function methods are particular cases of the SUMMA. The iterative step of the barrier-function method can be formulated as follows: minimize

$$f(x) + [(k-1)f(x) + b(x)] \tag{4.4}$$

to get x^k. Since, for $k = 2, 3, ...$, the function

$$(k-1)f(x) + b(x) \tag{4.5}$$

is minimized by x^{k-1}, the function

$$g_k(x) = (k-1)f(x) + b(x) - (k-1)f(x^{k-1}) - b(x^{k-1}) \tag{4.6}$$

is nonnegative, and x^k minimizes the function

$$G_k(x) = f(x) + g_k(x). \tag{4.7}$$

From

$$G_k(x) = f(x) + (k-1)f(x) + b(x) - (k-1)f(x^{k-1}) - b(x^{k-1}),$$

it follows that

$$G_k(x) - G_k(x^k) = kf(x) + b(x) - kf(x^k) - b(x^k) = g_{k+1}(x),$$

so that $g_{k+1}(x)$ satisfies the condition in (2.19). This shows that the barrier-function method is a particular case of SUMMA.

From the properties of SUMMA algorithms, we conclude that $\{f(x^k)\}$ is decreasing to $f(\hat{x})$, and that $\{g_k(x^k)\}$ converges to zero. From the nonnegativity of $g_k(x^k)$ we have that

$$(k-1)(f(x^k) - f(x^{k-1})) \geq b(x^{k-1}) - b(x^k).$$

Since the sequence $\{f(x^k)\}$ is decreasing, the sequence $\{b(x^k)\}$ must be increasing, but might not be bounded above.

If $\hat{x}$ is unique, and $f(x)$ has bounded level sets, then it follows, from our discussion of SUMMA, that $\{x^k\} \to \hat{x}$. Suppose now that $\hat{x}$ is not known to be unique, but can be chosen in D, so that $G_k(\hat{x})$ is finite for each k. From

$$f(\hat{x}) + \frac{1}{k}b(\hat{x}) \geq f(x^k) + \frac{1}{k}b(x^k)$$

we have

$$\frac{1}{k}\Big(b(\hat{x}) - b(x^k)\Big) \geq f(x^k) - f(\hat{x}) \geq 0,$$

so that

$$b(\hat{x}) - b(x^k) \geq 0,$$

for all k. If either f or b has bounded level sets, then the sequence $\{x^k\}$ is bounded and has a cluster point, x^* in C. It follows that $b(x^*) \leq b(\hat{x}) < +\infty$, so that x^* is in D. If we assume that $f(x)$ is convex and $b(x)$ is strictly convex on D, then we can show that x^* is unique in D, so that $x^* = \hat{x}$ and $\{x^k\} \to \hat{x}$.

To see this, assume, to the contrary, that there are two distinct cluster points x^* and x^{**} in D, with

$$\{x^{k_n}\} \to x^*,$$

and

$$\{x^{j_n}\} \to x^{**}.$$

Without loss of generality, we assume that

$$0 < k_n < j_n < k_{n+1},$$

for all n, so that

$$b(x^{k_n}) \leq b(x^{j_n}) \leq b(x^{k_{n+1}}).$$

Therefore,

$$b(x^*) = b(x^{**}) \leq b(\hat{x}).$$

From the strict convexity of $b(x)$ on the set D, and the convexity of $f(x)$, we conclude that, for $0 < \lambda < 1$ and $y = (1-\lambda)x^* + \lambda x^{**}$, we have $b(y) < b(x^*)$ and $f(y) \leq f(x^*)$. But, we must then have $f(y) = f(x^*)$. There must then be some k_n such that

$$G_{k_n}(y) = f(y) + \frac{1}{k_n}b(y) < f(x_{k_n}) + \frac{1}{k_n}b(x_{k_n}) = G_{k_n}(x^{k_n}).$$

But, this is a contradiction. ∎

The following theorem summarizes what we have shown with regard to the barrier-function method.

Theorem 4.1 *Let $f : \mathbb{R}^J \to (-\infty, +\infty]$ be a continuous function. Let $b(x) : \mathbb{R}^J \to (0, +\infty]$ be a continuous function, with effective domain the nonempty set D. Let $\hat{x}$ minimize $f(x)$ over all x in $C = \overline{D}$. For each positive integer k, let x^k minimize the function $f(x) + \frac{1}{k}b(x)$. Then the sequence $\{f(x^k)\}$ is monotonically decreasing to the limit $f(\hat{x})$, and the sequence $\{b(x^k)\}$ is increasing. If $\hat{x}$ is unique, and $f(x)$ has* *bounded level sets, then the sequence $\{x^k\}$ converges to $\hat{x}$. In particular, if $\hat{x}$ can be chosen in D, if either $f(x)$ or $b(x)$ has bounded level sets, if $f(x)$ is convex and if $b(x)$ is strictly convex on D, then $\hat{x}$ is unique in D and $\{x^k\}$ converges to $\hat{x}$.*

At the kth step of the barrier method we must minimize the function $f(x) + \frac{1}{k}b(x)$. In practice, this must also be performed iteratively, with, say, the Newton-Raphson algorithm. It is important, therefore, that barrier functions be selected so that relatively few Newton-Raphson steps are needed to produce acceptable solutions to the main problem. For more on these issues see Renegar [46] and Nesterov and Nemirovski [44].

5 Penalty-function Methods

When we add a barrier function to $f(x)$ we restrict the domain. When the barrier function is used in a sequential unconstrained minimization algorithm, the vector x^k that minimizes the function $f(x) + \frac{1}{k}b(x)$ lies in the effective domain D of $b(x)$, and we proved that, under certain conditions, the sequence $\{x^k\}$ converges to a minimizer of the function $f(x)$ over the closure of D. The constraint of lying within the set $\overline{D}$ is satisfied at every step of the algorithm; for that reason such algorithms are called interior-point methods. Constraints may also be imposed using a penalty function. In this case, violations of the constraints are discouraged, but not forbidden. When a penalty function is used in a sequential unconstrained minimization algorithm, the x^k need not satisfy the constraints; only the limit vector need be feasible.

5.1 Examples of Penalty Functions

Consider the convex programming problem. We wish to minimize the convex function $f(x)$ over all x for which the convex functions $g_i(x) \leq 0$, for $i = 1, ..., I$.

5.1.1 The Absolute-Value Penalty Function

We let $g_i^+(x) = \max\{g_i(x), 0\}$, and

$$p(x) = \sum_{i=1}^{I} g_i^+(x). \tag{5.1}$$

This is the *Absolute-Value* penalty function; it penalizes violations of the constraints $g_i(x) \leq 0$, but does not forbid such violations. Then, for $k = 1, 2, ...$, we minimize

$$f(x) + kp(x), \tag{5.2}$$

to get x^k. As $k \to +\infty$, the penalty function becomes more heavily weighted, so that, in the limit, the constraints $g_i(x) \leq 0$ should hold. Because only the limit vector satisfies the constraints, and the x^k are allowed to violate them, such a method is called an *exterior-point* method.

5.1.2 The Courant-Beltrami Penalty Function

The *Courant-Beltrami* penalty-function method is similar, but uses

$$p(x) = \sum_{i=1}^{I} [g_i^+(x)]^2. \tag{5.3}$$

5.1.3 The Quadratic-Loss Penalty Function

Penalty methods can also be used with equality constraints. Consider the problem of minimizing the convex function $f(x)$, subject to the constraints $g_i(x) = 0$, $i = 1, ..., I$. The *quadratic-loss* penalty function is

$$p(x) = \frac{1}{2} \sum_{i=1}^{I} (g_i(x))^2. \tag{5.4}$$

The inclusion of a penalty term can serve purposes other than to impose constraints on the location of the limit vector. In image processing, it is often desirable to obtain a reconstructed image that is locally smooth, but with well defined edges. Penalty functions that favor such images can then be used in the iterative reconstruction [29]. We survey several instances in which we would want to use a penalized objective function.

5.1.4 Regularized Least-Squares

Suppose we want to solve the system of equations $Ax = b$. The problem may have no exact solution, precisely one solution, or there may be infinitely many solutions. If we minimize the function

$$f(x) = \frac{1}{2}\|Ax - b\|_2^2,$$

we get a *least-squares* solution, generally, and an exact solution, whenever exact solutions exist. When the matrix A is ill-conditioned, small changes in the vector b can lead to large changes in the solution. When the vector b comes from measured data, the entries of b may include measurement errors, so that an exact solution of $Ax = b$ may be undesirable, even when such exact solutions exist; exact solutions may correspond to x with unacceptably large norm, for example. In such cases, we may, instead, wish to minimize a function such as

$$\frac{1}{2}\|Ax - b\|_2^2 + \frac{\epsilon}{2}\|x - z\|_2^2, \tag{5.5}$$

for some vector z. If $z = 0$, the minimizing vector x_ϵ is then a *norm-constrained* least-squares solution. We then say that the least-squares problem has been *regularized.* In the limit, as $\epsilon \to 0$, these regularized solutions x_ϵ converge to the least-squares solution closest to z.

Suppose the system $Ax = b$ has infinitely many exact solutions. Our problem is to select one. Let us select z that incorporates features of the desired solution, to the extent that we know them *a priori.* Then, as $\epsilon \to 0$, the vectors x_ϵ converge to the exact solution closest to z. For example, taking $z = 0$ leads to the *minimum-norm solution.*

5.1.5 Minimizing Cross-Entropy

In image processing, it is common to encounter systems $Px = y$ in which all the terms are non-negative. In such cases, it may be desirable to solve the system $Px = y$, approximately, perhaps, by minimizing the *cross-entropy* or *Kullback-Leibler distance*

$$KL(y, Px) = \sum_{i=1}^{I} \Big(y_i \log \frac{y_i}{(Px)_i} + (Px)_i - y_i \Big), \tag{5.6}$$

over vectors $x \geq 0$. When the vector y is noisy, the resulting solution, viewed as an image, can be unacceptable. It is wise, therefore, to add a penalty term, such as $p(x) = \epsilon KL(z, x)$, where $z > 0$ is a prior estimate of the desired x [35, 49, 36, 11].

A similar problem involves minimizing the function $KL(Px, y)$. Once again, noisy results can be avoided by including a penalty term, such as $p(x) = \epsilon KL(x, z)$ [11].

5.1.6 The Lagrangian in Convex Programming

When there is a sensitivity vector λ for the CP problem, minimizing $f(x)$ is equivalent to minimizing the Lagrangian,

$$f(x) + \sum_{i=1}^{I} \lambda_i g_i(x) = f(x) + p(x); \tag{5.7}$$

in this case, the addition of the second term, $p(x)$, serves to incorporate the constraints $g_i(x) \leq 0$ in the function to be minimized, turning a constrained minimization problem into an unconstrained one. The problem of minimizing the Lagrangian still remains, though. We may have to solve that problem using an iterative algorithm.

5.1.7 Infimal Convolution

The *infimal convolution* of the functions f and g is defined as

$$(f \oplus g)(z) = \inf_x \Big\{ f(x) + g(z-x) \Big\}.$$

The *infimal deconvolution* of f and g is defined as

$$(f \ominus g)(z) = \sup_x \Big\{ f(z-x) - g(x) \Big\}.$$

5.1.8 Moreau's Proximity-Function Method

The Moreau envelope of the function f is the function

$$m_f(z) = \inf_x \Big\{ f(x) + \frac{1}{2}\|x-z\|_2^2 \Big\}, \tag{5.8}$$

which is also the *infimal convolution* of the functions $f(x)$ and $\frac{1}{2}\|x\|_2^2$. It can be shown that the infimum is uniquely attained at the point denoted $x = \text{prox}_f z$ (see [47]). In similar fashion, we can define $m_{f^*}z$ and $\text{prox}_{f^*}z$, where $f^*(z)$ denotes the function conjugate to f.

Proposition 5.1 *The infimum of $m_f(z)$, over all z, is the same as the infimum of $f(x)$, over all x.*

Proof: We have

$$\inf_z m_f(z) = \inf_z \inf_x \{f(x) + \frac{1}{2}\|x-z\|_2^2\}$$

$$= \inf_x \inf_z \{f(x) + \frac{1}{2}\|x-z\|_2^2\} = \inf_x \{f(x) + \frac{1}{2}\inf_z \|x-z\|_2^2\} = \inf_x f(x).$$

∎

The minimizers of $m_f(z)$ and $f(x)$ are the same, as well. Therefore, one way to use Moreau's method is to replace the original problem of minimizing the possibly non-smooth function $f(x)$ with the problem of minimizing the smooth function $m_f(z)$. Another way is to convert Moreau's method into a sequential minimization algorithm, replacing z with x^{k-1} and minimizing with respect to x to get x^k. As we shall see, this leads to the proximal minimization algorithm.

5.2 The Roles Penalty Functions Play

From the examples just surveyed, we can distinguish several distinct roles that penalty functions can play.

5.2.1 Impose Constraints

The first role is to penalize violations of constraints, as part of sequential minimization, or even to turn a constrained minimization into an equivalent unconstrained one: the Absolute-Value and Courant-Beltrami penalty functions penalize violations of the constraints $g_i(x) \leq 0$, while Quadratic-Loss penalty function penalizes violations of the constraints $g_i(x) = 0$. The augmented objective functions $f(x) + kp(x)$ now become part of a sequential unconstrained minimization method. It is sometimes possible for $f(x)$ and $f(x)+p(x)$ to have the same minimizers, or for constrained minimizers of $f(x)$ to be the same as unconstrained minimizers of $f(x)+p(x)$, as happens with the Lagrangian in the CP problem.

5.2.2 Regularization

The second role is regularization: in the least-squares problem, the main purpose for adding the norm-squared penalty function in Equation (5.5) is to reduce sensitivity to noise in the entries of the vector b. Also, regularization will usually turn a problem with multiple solutions into one with a unique solution.

5.2.3 Incorporate Prior Information

The third role is to incorporate prior information: when $Ax = b$ is under-determined, using the penalty function $\epsilon\|x - z\|_2^2$ and letting $\epsilon \to 0$ encourages the solution to be close to the prior estimate z.

5.2.4 Simplify Calculations

A fourth role that penalty functions can play is to simplify calculation: in the case of cross-entropy minimization, adding the penalty functions $KL(z, x)$ and $KL(x, z)$ to the objective functions $KL(y, Px)$ and $KL(Px, y)$, respectively, regularizes the minimization problem. But, as we shall see later, the SMART algorithm minimizes $KL(Px, y)$ by using a sequential approach, in which each minimizer x^k can be calculated in closed form.

5.2.5 Sequential Unconstrained Minimization

More generally, a fifth role for penalty functions is as part of sequential minimization. Here the goal is to replace one computationally difficult minimization with a sequence of simpler ones. Clearly, one reason for the difficulty can be that the original problem is constrained, and the sequential approach uses a series of unconstrained minimizations, penalizing violations of the constraints through the penalty function. However, there are other instances in which the sequential approach serves to simplify the calculations, not to remove constraints, but, perhaps, to replace a non-differentiable objective function with a differentiable one, or a sequence of differentiable ones, as in Moreau's method.

Once again, our objective is to find a sequence $\{x^k\}$ such that $\{f(x^k)\} \to d$. We select a penalty function $p(x)$ with $p(x) \geq 0$ and $p(x) = 0$ if and only if x is in P. For $k = 1, 2, ...$, let x^k be a minimizer of the function $f(x) + kp(x)$. As we shall see, we can formulate this penalty-function algorithm as a barrier-function iteration.

In order to relate penalty-function methods to barrier-function methods, we note that minimizing $T_k(x) = f(x)+kp(x)$ is equivalent to minimizing $p(x)+\frac{1}{k}f(x)$. This is the form of the barrier-function iteration, with $p(x)$ now in the role previously played by $f(x)$, and $f(x)$ now in the role previously played by $b(x)$. We are not concerned here with the effective domain of $f(x)$. Therefore, we can now mimic most, but not all, of what we did for barrier-function methods.

5.3 Basic Facts

Lemma 5.1 *The sequence $\{T_k(x^k)\}$ is increasing, bounded above by d and converges to some $\gamma \leq d$.*

Proof: We have

$$T_k(x^k) \leq T_k(x^{k+1}) \leq T_k(x^{k+1}) + p(x^{k+1}) = T_{k+1}(x^{k+1}).$$

Also, for any $z \in P$, and for each k, we have

$$f(z) = f(z) + kp(z) = T_k(z) \geq T_k(x^k);$$

therefore $d \geq \gamma$. ▮

Lemma 5.2 *The sequence $\{p(x^k)\}$ is decreasing to zero, the sequence $\{f(x^k)\}$ is increasing and converging to some $\beta \leq d$.*

Proof: Since x^k minimizes $T_k(x)$ and x^{k+1} minimizes $T_{k+1}(x)$, we have

$$f(x^k) + kp(x^k) \leq f(x^{k+1}) + kp(x^{k+1}),$$

and

$$f(x^{k+1}) + (k+1)p(x^{k+1}) \leq f(x^k) + (k+1)p(x^k).$$

Consequently, we have

$$(k+1)[p(x^k)-p(x^{k+1})] \geq f(x^{k+1})-f(x^k) \geq k[p(x^k)-p(x^{k+1})].$$

Therefore,

$$p(x^k)-p(x^{k+1}) \geq 0,$$

and

$$f(x^{k+1})-f(x^k) \geq 0.$$

From

$$f(x^k) \leq f(x^k)+kp(x^k)=T_k(x^k) \leq \gamma \leq d,$$

it follows that the sequence $\{f(x^k)\}$ is increasing and converges to some $\beta \leq \gamma$. Since

$$\alpha+kp(x^k) \leq f(x^k)+kp(x^k)=T_k(x^k) \leq \gamma$$

for all k, we have $0 \leq kp(x^k) \leq \gamma-\alpha$. Therefore, the sequence $\{p(x^k)\}$ converges to zero. ▮

We want $\beta = d$. To obtain this result, it appears that we need to make more assumptions: we assume, therefore, that X is a complete metric space, P is closed in X, the functions f and p are continuous and f has compact level sets. From these assumptions, we are able to assert that the sequence $\{x^k\}$ is bounded, so that there is a convergent subsequence; let $\{x^{k_n}\} \to x^*$. It follows that $p(x^*)=0$, so that x^* is in P. Then

$$f(x^*)=f(x^*)+p(x^*)=\lim_{n\to+\infty}(f(x^{k_n})+p(x^{k_n})) \leq \lim_{n\to+\infty}T_{k_n}(x^{k_n})=\gamma \leq d.$$

But $x^* \in P$, so $f(x^*) \geq d$. Therefore, $f(x^*)=d$.

It may seem odd that we are trying to minimize $f(x)$ over the set P using a sequence $\{x^k\}$ with $\{f(x^k)\}$ increasing, but remember that these x^k are not in P.

6 Proximity-function Minimization

Let $f:\mathbb{R}^J \to (-\infty,+\infty]$ be a closed, proper, and convex function. Let h be a closed proper convex function, with effective domain D, that is differentiable on the nonempty open convex set int D. Assume that $f(x)$ is finite on $C=\overline{D}$ and attains its minimum value on C at $\hat{x}$. The corresponding *Bregman distance* $D_h(x,z)$ is defined for x in D and z in int D by

$$D_h(x,z)=h(x)-h(z)-\langle\nabla h(z),x-z\rangle. \tag{6.1}$$

Note that $D_h(x,z) \geq 0$ always. If h is essentially strictly convex, then $D_h(x,z)=0$ implies that $x=z$. Our objective is to minimize $f(x)$ over x in $C=\overline{D}$.

6.1 Proximal Minimization Algorithms

At the kth step of the *proximal minimization algorithm* (PMA) [13], we minimize the function

$$G_k(x) = f(x) + D_h(x, x^{k-1}), \tag{6.2}$$

to get x^k. The function

$$g_k(x) = D_h(x, x^{k-1}) \tag{6.3}$$

is nonnegative and $g_k(x^{k-1}) = 0$. We assume that each x^k lies in int D.

We show now that the PMA is a particular case of the SUMMA. We remind the reader that $f(x)$ is now assumed to be convex.

Lemma 6.1 *For each k we have*

$$G_k(x) - G_k(x^k) \geq D_h(x, x^k) = g_{k+1}(x). \tag{6.4}$$

Proof: Since x^k minimizes $G_k(x)$ within the set D, we have

$$0 \in \partial f(x^k) + \nabla h(x^k) - \nabla h(x^{k-1}), \tag{6.5}$$

so that

$$\nabla h(x^{k-1}) = u^k + \nabla h(x^k), \tag{6.6}$$

for some u^k in $\partial f(x^k)$. Then

$$G_k(x) - G_k(x^k) = f(x) - f(x^k) + h(x) - h(x^k) - \langle \nabla h(x^{k-1}), x - x^k \rangle.$$

Now substitute, using Equation (6.6), to get

$$G_k(x) - G_k(x^k) = f(x) - f(x^k) - \langle u^k, x - x^k \rangle + D_h(x, x^k). \tag{6.7}$$

Therefore,

$$G_k(x) - G_k(x^k) \geq D_h(x, x^k),$$

since u^k is in $\partial f(x^k)$. ∎

From the discussion of the SUMMA we know that $\{f(x^k)\}$ is monotonically decreasing to $f(\hat{x})$. As we noted previously, if the sequence $\{x^k\}$ is bounded, and $\hat{x}$ is unique, we can conclude that $\{x^k\} \to \hat{x}$.

Suppose that $\hat{x}$ is not known to be unique, but can be chosen in D; this will be the case, of course, whenever D is closed. Then $G_k(\hat{x})$ is finite for each k. From the definition of $G_k(x)$ we have

$$G_k(\hat{x}) = f(\hat{x}) + D_h(\hat{x}, x^{k-1}). \tag{6.8}$$

From Equation (6.7) we have

$$G_k(\hat{x}) = G_k(x^k) + f(\hat{x}) - f(x^k) - \langle u^k, \hat{x} - x^k \rangle + D_h(\hat{x}, x^k), \tag{6.9}$$

so that

$$G_k(\hat{x}) = f(x^k) + D_h(x^k, x^{k-1}) + f(\hat{x}) - f(x^k) - \langle u^k, \hat{x} - x^k \rangle + D_h(\hat{x}, x^k). \tag{6.10}$$

Therefore,

$$D_h(\hat{x}, x^{k-1}) - D_h(\hat{x}, x^k) =$$

$$f(x^k) - f(\hat{x}) + D_h(x^k, x^{k-1}) + f(\hat{x}) - f(x^k) - \langle u^k, \hat{x} - x^k \rangle. \tag{6.11}$$

It follows that the sequence $\{D_h(\hat{x}, x^k)\}$ is decreasing and that $\{f(x^k)\}$ converges to $f(\hat{x})$. If either the function $f(x)$ or the function $D_h(\hat{x}, \cdot)$ has bounded level sets, then the sequence $\{x^k\}$ is bounded, has cluster points x^* in C, and $f(x^*) = f(\hat{x})$, for every x^*. We now show that $\hat{x}$ in D implies that x^* is also in D, whenever h is a Bregman -Legendre function.

Let x^* be an arbitrary cluster point, with $\{x^{k_n}\} \to x^*$. If $\hat{x}$ is not in the interior of D, then, by Property B2 of Bregman-Legendre functions, we know that

$$D_h(x^*, x^{k_n}) \to 0,$$

so x^* is in D. Then the sequence $\{D_h(x^*, x^k)\}$ is decreasing. Since a subsequence converges to zero, we have $\{D_h(x^*, x^k)\} \to 0$. From Property R5, we conclude that $\{x^k\} \to x^*$.

If $\hat{x}$ is in int D, but x^* is not, then $\{D_h(\hat{x}, x^k)\} \to +\infty$, by Property R2. But, this is a contradiction; therefore x^* is in D. Once again, we conclude that $\{x^k\} \to x^*$.

Now we summarize our results for the PMA. Let $f : \mathbb{R}^J \to (-\infty, +\infty]$ be closed, proper, convex and differentiable. Let h be a closed proper convex function, with effective domain D, that is differentiable on the nonempty open convex set int D. Assume that $f(x)$ is finite on $C = \overline{D}$ and attains its minimum value on C at $\hat{x}$. For each positive integer k, let x^k minimize the function $f(x) + D_h(x, x^{k-1})$. Assume that each x^k is in the interior of D.

Theorem 6.1 *If the restriction of $f(x)$ to x in C has bounded level sets and $\hat{x}$ is unique, and then the sequence $\{x^k\}$ converges to $\hat{x}$.*

Theorem 6.2 *If $h(x)$ is a Bregman-Legendre function and $\hat{x}$ can be chosen in D, then $\{x^k\} \to x^*$, x^* in D, with $f(x^*) = f(\hat{x})$.*

7 The Forward-Backward Splitting Algorithm

The *forward-backward splitting* methods form a quite large subclass of the SUMMA algorithms.

7.1 Moreau's Proximity Operators

Let $f : \mathbb{R}^J \to \mathbb{R}$ be convex. For each $z \in \mathbb{R}^J$ the function

$$m_f(z) := \min_x \{f(x) + \frac{1}{2}\|x - z\|_2^2\} \tag{7.1}$$

is minimized by a unique x [47]. The operator that associates with each z the minimizing x is Moreau's proximity operator, and we write $x = \text{prox}_f(z)$. The operator prox_f extends the notion of orthogonal projection onto a closed convex set [40, 41, 42]. We have $x = \text{prox}_f(z)$ if and only if $z - x \in \partial f(x)$, where the set $\partial f(x)$ is the subdifferential of f at x, given by

$$\partial f(x) := \{u | \langle u, y - x\rangle \leq f(y) - f(x), \text{for all}\, y\}. \tag{7.2}$$

Proximity operators are also firmly non-expansive [25]; indeed, the proximity operator prox_f is the resolvent of the maximal monotone operator $B(x) = \partial f(x)$ and all such resolvent operators are firmly non-expansive [9].

7.2 The Forward-Backward Splitting Algorithm

Our objective here is to provide an elementary proof of convergence for the forward-backward splitting (FBS) algorithm; a detailed discussion of this algorithm and its history is given by Combettes and Wajs in [25].

Let $f : \mathbb{R}^J \to \mathbb{R}$ be convex, with $f = f_1 + f_2$, both convex, f_2 differentiable, and ∇f_2 L-Lipschitz continuous. The iterative step of the FBS algorithm is

$$x^k = \text{prox}_{\gamma f_1}\Big(x^{k-1} - \gamma \nabla f_2(x^{k-1})\Big). \tag{7.3}$$

As we shall show, convergence of the sequence $\{x^k\}$ to a solution can be established, if γ is chosen to lie within the interval $(0, 1/L]$.

7.3 Convergence of the FBS algorithm

Let $f : \mathbb{R}^J \to \mathbb{R}$ be convex, with $f = f_1 + f_2$, both convex, f_2 differentiable, and ∇f_2 L-Lipschitz continuous. Let $\{x^k\}$ be defined by Equation (7.3) and let $0 < \gamma \leq 1/L$.

For each $k = 1, 2, ...$ let

$$G_k(x) = f(x) + \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_{f_2}(x, x^{k-1}), \tag{7.4}$$

where

$$D_{f_2}(x, x^{k-1}) = f_2(x) - f_2(x^{k-1}) - \langle \nabla f_2(x^{k-1}), x - x^{k-1}\rangle. \tag{7.5}$$

Since $f_2(x)$ is convex, $D_{f_2}(x, y) \geq 0$ for all x and y and is the Bregman distance formed from the function f_2 [8].

The auxiliary function

$$g_k(x) = \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_{f_2}(x, x^{k-1}) \tag{7.6}$$

can be rewritten as

$$g_k(x) = D_h(x, x^{k-1}), \tag{7.7}$$

where

$$h(x) = \frac{1}{2\gamma}\|x\|_2^2 - f_2(x). \tag{7.8}$$

Therefore, $g_k(x) \geq 0$ whenever $h(x)$ is a convex function.

We know that $h(x)$ is convex if and only if

$$\langle \nabla h(x) - \nabla h(y), x - y\rangle \geq 0, \tag{7.9}$$

for all x and y. This is equivalent to

$$\frac{1}{\gamma}\|x - y\|_2^2 - \langle \nabla f_2(x) - \nabla f_2(y), x - y\rangle \geq 0. \tag{7.10}$$

Since ∇f_2 is L-Lipschitz, the inequality (7.10) holds for $0 < \gamma \leq 1/L$.

Lemma 7.1 *The x^k that minimizes $G_k(x)$ over x is given by Equation (7.3).*

Proof: We know that x^k minimizes $G_k(x)$ if and only if

$$0 \in \nabla f_2(x^k) + \frac{1}{\gamma}(x^k - x^{k-1}) - \nabla f_2(x^k) + \nabla f_2(x^{k-1}) + \partial f_1(x^k),$$

or, equivalently,

$$\Big(x^{k-1} - \gamma\nabla f_2(x^{k-1})\Big) - x^k \in \partial(\gamma f_1)(x^k).$$

Consequently,

$$x^k = \text{prox}_{\gamma f_1}(x^{k-1} - \gamma\nabla f_2(x^{k-1})).$$

▮

Theorem 7.1 *The sequence $\{x^k\}$ converges to a minimizer of the function $f(x)$, whenever minimizers exist.*

Proof: A relatively simple calculation shows that

$$G_k(x) - G_k(x^k) = \frac{1}{2\gamma}\|x - x^k\|_2^2 +$$

$$\Big(f_1(x) - f_1(x^k) - \frac{1}{\gamma}\langle(x^{k-1} - \gamma\nabla f_2(x^{k-1})) - x^k, x - x^k\rangle\Big). \tag{7.11}$$

Since

$$(x^{k-1} - \gamma\nabla f_2(x^{k-1})) - x^k \in \partial(\gamma f_1)(x^k),$$

it follows that

$$\Big(f_1(x) - f_1(x^k) - \frac{1}{\gamma}\langle(x^{k-1} - \gamma\nabla f_2(x^{k-1})) - x^k, x - x^k\rangle\Big) \geq 0.$$

Therefore,

$$G_k(x) - G_k(x^k) \geq \frac{1}{2\gamma}\|x - x^k\|_2^2 \geq g_{k+1}(x). \tag{7.12}$$

Therefore, the inequality in (2.19) holds and the iteration fits into the SUMMA class.

Now let $\hat{x}$ minimize $f(x)$ over all x. Then

$$G_k(\hat{x}) - G_k(x^k) = f(\hat{x}) + g_k(\hat{x}) - f(x^k) - g_k(x^k)$$

$$\leq f(\hat{x}) + G_{k-1}(\hat{x}) - G_{k-1}(x^{k-1}) - f(x^k) - g_k(x^k),$$

so that

$$\Big(G_{k-1}(\hat{x}) - G_{k-1}(x^{k-1})\Big) - \Big(G_k(\hat{x}) - G_k(x^k)\Big) \geq f(x^k) - f(\hat{x}) + g_k(x^k) \geq 0.$$

Therefore, the sequence $\{G_k(\hat{x}) - G_k(x^k)\}$ is decreasing and the sequences $\{g_k(x^k)\}$ and $\{f(x^k) - f(\hat{x})\}$ converge to zero.

From

$$G_k(\hat{x}) - G_k(x^k) \geq \frac{1}{2\gamma}\|\hat{x} - x^k\|_2^2,$$

it follows that the sequence $\{x^k\}$ is bounded. Therefore, we may select a subsequence $\{x^{k_n}\}$ converging to some x^{**}, with $\{x^{k_n-1}\}$ converging to some x^*, and therefore $f(x^*) = f(x^{**}) = f(\hat{x})$.

Replacing the generic $\hat{x}$ with x^{**}, we find that $\{G_k(x^{**}) - G_k(x^k)\}$ is decreasing to zero. From the inequality in (7.12), we conclude that the sequence $\{\|x^* - x^k\|_2^2\}$ converges to zero, and so $\{x^k\}$ converges to x^*. This completes the proof of the theorem. ▮

7.4 Some Examples

We present some examples to illustrate the application of the convergence theorem.

7.4.1 Projected Gradient Descent

Let C be a non-empty, closed convex subset of $\mathbb{R}^J$ and $f_1(x) = \iota_C(x)$, the function that is $+\infty$ for x not in C and zero for x in C. Then $\iota_C(x)$ is convex, but not differentiable. We have $\text{prox}_{\gamma f_1} = P_C$, the orthogonal projection onto C. The iteration in Equation (7.3) becomes

$$x^k = P_C\Big(x^{k-1} - \gamma \nabla f_2(x^{k-1})\Big). \tag{7.13}$$

The sequence $\{x^k\}$ converges to a minimizer of f_2 over $x \in C$, whenever such minimizers exist, for $0 < \gamma \leq 1/L$.

7.4.2 The CQ Algorithm

Let A be a real I by J matrix, $C \subseteq \mathbb{R}^J$, and $Q \subseteq \mathbb{R}^I$, both closed convex sets. The split feasibility problem (SFP) is to find x in C such that Ax is in Q. The function

$$f_2(x) = \frac{1}{2}\|P_Q Ax - Ax\|_2^2 \tag{7.14}$$

is convex, differentiable and ∇f_2 is L-Lipschitz for $L = \rho(A^T A)$, the spectral radius of $A^T A$. The gradient of f_2 is

$$\nabla f_2(x) = A^T(I - P_Q)Ax. \tag{7.15}$$

We want to minimize the function $f_2(x)$ over x in C, or, equivalently, to minimize the function $f(x) = \iota_C(x) + f_2(x)$. The projected gradient descent algorithm has the iterative step

$$x^k = P_C\Big(x^{k-1} - \gamma A^T(I - P_Q)Ax^{k-1}\Big); \tag{7.16}$$

this iterative method was called the CQ-algorithm in [14, 15]. The sequence $\{x^k\}$ converges to a solution whenever f_2 has a minimum on the set C, for $0 < \gamma \leq 1/L$.

In [23, 22] the CQ algorithm was extended to a multiple-sets algorithm and applied to the design of protocols for intensity-modulated radiation therapy.

7.4.3 The Projected Landweber Algorithm

The problem is to minimize the function

$$f_2(x) = \frac{1}{2}\|Ax - b\|_2^2,$$

over $x \in C$. This is a special case of the SFP and we can use the CQ-algorithm, with $Q = \{b\}$. The resulting iteration is the projected Landweber algorithm [7]; when $C = \mathbb{R}^J$ it becomes the Landweber algorithm [34].

7.5 Minimizing f_2 over a Linear Manifold

Suppose that we want to minimize f_2 over x in the linear manifold $M = S + p$, where S is a subspace of $\mathbb{R}^J$ of dimension $I < J$ and p is a fixed vector. Let A be an I by J matrix such that the I columns of A^T form a basis for S. For each $z \in \mathbb{R}^I$ let

$$d(z) = f_2(A^T z + p),$$

so that d is convex, differentiable, and its gradient,

$$\nabla d(z) = A\nabla f_2(A^T z + p),$$

is K-Lipschitz continuous, for $K = \rho(A^T A)L$. The sequence $\{z^k\}$ defined by

$$z^k = z^{k-1} - \gamma\nabla d(z^{k-1}) \tag{7.17}$$

converges to a minimizer of d over all z in $\mathbb{R}^I$, whenever minimizers exist, for $0 < \gamma \leq 1/K$.

From Equation (7.17) we get

$$x^k = x^{k-1} - \gamma A^T A\nabla f_2(x^{k-1}), \tag{7.18}$$

with $x^k = A^T z^k + p$. The sequence $\{x^k\}$ converges to a minimizer of f_2 over all x in M.

Suppose now that we begin with an algorithm having the iterative step

$$x^k = x^{k-1} - \gamma A^T A \nabla f_2(x^{k-1}), \tag{7.19}$$

where A is any real I by J matrix having rank I. Let x^0 be in the range of A^T, so that $x^0 = A^T z^0$, for some $z^0 \in \mathbb{R}^I$. Then each $x^k = A^T z^k$ is again in the range of A^T, and we have

$$A^T z^k = A^T z^{k-1} - \gamma A^T A \nabla f_2(A^T z^{k-1}). \tag{7.20}$$

With $d(z) = f_2(A^T z)$, we can write Equation (7.20) as

$$A^T \Big(z^k - (z^{k-1} - \gamma \nabla d(z^{k-1})) \Big) = 0. \tag{7.21}$$

Since A has rank I, A^T is one-to-one, so that

$$z^k - z^{k-1} - \gamma \nabla d(z^{k-1}) = 0. \tag{7.22}$$

The sequence $\{z^k\}$ converges to a minimizer of d, over all $z \in \mathbb{R}^I$, whenever such minimizers exist, for $0 < \gamma \leq 1/K$. Therefore, the sequence $\{x^k\}$ converges to a minimizer of f_2 over all x in the range of A^T.

7.6 Feasible-Point Algorithms

Suppose that we want to minimize a convex differentiable function $f(x)$ over x such that $Ax = b$, where A is an I by J full-rank matrix, with $I < J$. If $Ax^k = b$ for each of the vectors $\{x^k\}$ generated by the iterative algorithm, we say that the algorithm is a feasible-point method.

7.6.1 The Projected Gradient Algorithm

Let C be the feasible set of all x in $\mathbb{R}^J$ such that $Ax = b$. For every z in $\mathbb{R}^J$, we have

$$P_C z = P_{NS(A)} z + A^T (AA^T)^{-1} b, \tag{7.23}$$

where $NS(A)$ is the null space of A. Using

$$P_{NS(A)} z = z - A^T (AA^T)^{-1} Az, \tag{7.24}$$

we have

$$P_C z = z + A^T (AA^T)^{-1} (b - Az). \tag{7.25}$$

Using Equation (7.3), we get the iteration step for the projected gradient algorithm:

$$x^k = x^{k-1} - \gamma P_{NS(A)} \nabla f(x^{k-1}), \tag{7.26}$$

which converges to a solution for $0 < \gamma \leq 1/L$, whenever solutions exist.

Next we present a somewhat simpler approach.

7.6.2 The Reduced Gradient Algorithm

Let x^0 be a feasible point, that is, $Ax^0 = b$. Then $x = x^0 + p$ is also feasible if p is in the null space of A, that is, $Ap = 0$. Let Z be a J by $J - I$ matrix whose columns form a basis for the null space of A. We want $p = Zv$ for some v. The best v will be the one for which the function

$$\phi(v) = f(x^0 + Zv)$$

is minimized. We can apply to the function $\phi(v)$ the steepest descent method, or the Newton-Raphson method, or any other minimization technique.

The steepest descent method, applied to $\phi(v)$, is called the reduced steepest descent algorithm [43]. The gradient of $\phi(v)$, also called the reduced gradient, is

$$\nabla\phi(v) = Z^T \nabla f(x),$$

where $x = x^0 + Zv$; the gradient operator $\nabla\phi$ is then K-Lipschitz, for $K = \rho(A^T A)L$.

Let x^0 be feasible. The iteration in Equation (7.3) now becomes

$$v^k = v^{k-1} - \gamma\nabla\phi(v^{k-1}), \tag{7.27}$$

so that the iteration for $x^k = x^0 + Zv^k$ is

$$x^k = x^{k-1} - \gamma Z Z^T \nabla f(x^{k-1}). \tag{7.28}$$

The vectors x^k are feasible and the sequence $\{x^k\}$ converges to a solution, whenever solutions exist, for any $0 < \gamma < \frac{1}{K}$.

7.6.3 The Reduced Newton-Raphson Method

The same idea can be applied to the Newton-Raphson method. The Newton-Raphson method, applied to $\phi(v)$, is called the reduced Newton-Raphson method [43]. The Hessian matrix of $\phi(v)$, also called the reduced Hessian matrix, is

$$\nabla^2\phi(v) = Z^T \nabla^2 f(c) Z,$$

so that the reduced Newton-Raphson iteration becomes

$$x^k = x^{k-1} - Z\Big(Z^T \nabla^2 f(x^{k-1}) Z\Big)^{-1} Z^T \nabla f(x^{k-1}). \tag{7.29}$$

Let c^0 be feasible. Then each x^k is feasible. The sequence $\{x^k\}$ is not guaranteed to converge.

8 The SMART and EMML Algorithms

Our next examples are the simultaneous multiplicative algebraic reconstruction technique (SMART) and the expectation maximization maximum likelihood (EMML) algorithms. For $a > 0$ and $b > 0$, the Kullback-Leibler distance, $KL(a, b)$, is defined as

$$KL(a, b) = a \log \frac{a}{b} + b - a. \tag{8.1}$$

In addition, $KL(0, 0) = 0$, $KL(a, 0) = +\infty$ and $KL(0, b) = b$. The KL distance is then extended to nonnegative vectors coordinate-wise.

8.1 The SMART Iteration

The SMART minimizes the function $f(x) = KL(Px, y)$, over nonnegative vectors x. Here y is a vector with positive entries, and P is a matrix with nonnegative entries, such that $s_j = \sum_{i=1}^{I} P_{ij} > 0$. Denote by $\mathcal{X}$ the set of all nonnegative x for which the vector Px has only positive entries.

Having found the vector x^{k-1}, the next vector in the SMART sequence is x^k, with entries given by

$$x_j^k = x_j^{k-1} \exp s_j^{-1} \Big(\sum_{i=1}^{I} P_{ij} \log(y_i/(Px^{k-1})_i) \Big). \tag{8.2}$$

8.2 The EMML Iteration

The EMML algorithm minimizes the function $f(x) = KL(y, Px)$, over nonnegative vectors x. Having found the vector x^{k-1}, the next vector in the EMML sequence is x^k, with entries given by

$$x_j^k = x_j^{k-1} s_j^{-1} \Big(\sum_{i=1}^{I} P_{ij} (y_i/(Px^{k-1})_i) \Big). \tag{8.3}$$

8.3 The EMML and the SMART as Alternating Minimization

In [11] the SMART was derived using the following alternating minimization approach.

For each $x \in \mathcal{X}$, let $r(x)$ and $q(x)$ be the I by J arrays with entries

$$r(x)_{ij} = x_j P_{ij} y_i/(Px)_i, \tag{8.4}$$

and

$$q(x)_{ij} = x_j P_{ij}. \tag{8.5}$$

In the iterative step of the SMART we get x^k by minimizing the function

$$KL(q(x), r(x^{k-1})) = \sum_{i=1}^{I} \sum_{j=1}^{J} KL(q(x)_{ij}, r(x^{k-1})_{ij})$$

over $x \geq 0$. Note that $KL(Px, y) = KL(q(x), r(x))$.

Similarly, the iterative step of the EMML is to minimize the function $KL(r(x^{k-1}), q(x))$ to get $x = x^k$. Note that $KL(y, Px) = KL(r(x), q(x))$. It follows from the identities established in [11] that the SMART can also be formulated as a particular case of the SUMMA.

8.4 The SMART as a Case of SUMMA

We show now that the SMART is a particular case of the SUMMA. The following lemma is helpful in that regard.

Lemma 8.1 *For any non-negative vectors x and z, with $z_+ = \sum_{j=1}^{J} z_j > 0$, we have*

$$KL(x, z) = KL(x_+, z_+) + KL(x, \frac{x_+}{z_+} z). \tag{8.6}$$

For notational convenience, we assume, for the remainder of this chapter, that $s_j = 1$ for all j. From the identities established for the SMART in [11], we know that the iterative step of SMART can be expressed as follows: minimize the function

$$G_k(x) = KL(Px, y) + KL(x, x^{k-1}) - KL(Px, Px^{k-1}) \tag{8.7}$$

to get x^k. According to Lemma 8.1, the quantity

$$g_k(x) = KL(x, x^{k-1}) - KL(Px, Px^{k-1})$$

is nonnegative, since $s_j = 1$. The $g_k(x)$ are defined for all nonnegative x; that is, the set D is the closed nonnegative orthant in $\mathbb{R}^J$. Each x^k is a positive vector.

It was shown in [11] that

$$G_k(x) = G_k(x^k) + KL(x, x^k), \tag{8.8}$$

from which it follows immediately that Assumption 2 holds for the SMART, so that the SMART is in the SUMMA class.

Because the SMART is a particular case of the SUMMA, we know that the sequence $\{f(x^k)\}$ is monotonically decreasing to $f(\hat{x})$. It was shown in [11] that if $y = Px$ has no nonnegative solution and the matrix P and every submatrix obtained from P by removing columns has full rank, then $\hat{x}$ is unique; in that case, the sequence $\{x^k\}$ converges to $\hat{x}$. As we shall see, the SMART sequence always converges to a nonnegative minimizer of $f(x)$. To establish this, we reformulate the SMART as a particular case of the PMA.

8.5 The SMART as a Case of the PMA

We take $F(x)$ to be the function

$$F(x) = \sum_{j=1}^{J} x_j \log x_j. \tag{8.9}$$

Then

$$D_F(x, z) = KL(x, z). \tag{8.10}$$

For nonnegative x and z in $\mathcal{X}$, we have

$$D_f(x, z) = KL(Px, Pz). \tag{8.11}$$

Lemma 8.2 $D_F(x, z) \geq D_f(x, z)$.

Proof: We have

$$\begin{aligned} D_F(x, z) \geq \sum_{j=1}^{J} KL(x_j, z_j) &\geq \sum_{j=1}^{J} \sum_{i=1}^{I} KL(P_{ij}x_j, P_{ij}z_j) \\ &\geq \sum_{i=1}^{I} KL((Px)_i, (Pz)_i) = KL(Px, Pz). \end{aligned} \tag{8.12}$$

∎

We let $h(x) = F(x) - f(x)$; then $D_h(x, z) \geq 0$ for nonnegative x and z in $\mathcal{X}$. The iterative step of the SMART is to minimize the function

$$f(x) + D_h(x, x^{k-1}). \tag{8.13}$$

So the SMART is a particular case of the PMA.

The function $h(x) = F(x) - f(x)$ is finite on D the nonnegative orthant of $\mathbb{R}^J$, and differentiable on the interior, so $C = D$ is closed in this example. Consequently,

$\hat{x}$ is necessarily in D. From our earlier discussion of the PMA, we can conclude that the sequence $\{D_h(\hat{x}, x^k)\}$ is decreasing and the sequence $\{D_f(\hat{x}, x^k)\} \to 0$. Since the function $KL(\hat{x}, \cdot)$ has bounded level sets, the sequence $\{x^k\}$ is bounded, and $f(x^*) = f(\hat{x})$, for every cluster point. Therefore, the sequence $\{D_h(x^*, x^k)\}$ is decreasing. Since a subsequence converges to zero, the entire sequence converges to zero. The convergence of $\{x^k\}$ to x^* follows from basic properties of the KL distance.

From the fact that $\{D_f(\hat{x}, x^k)\} \to 0$, we conclude that $P\hat{x} = Px^*$. Equation (6.11) now tells us that the difference $D_h(\hat{x}, x^{k-1}) - D_h(\hat{x}, x^k)$ depends on only on $P\hat{x}$, and not directly on $\hat{x}$. Therefore, the difference $D_h(\hat{x}, x^0) - D_h(\hat{x}, x^*)$ also depends only on $P\hat{x}$ and not directly on $\hat{x}$. Minimizing $D_h(\hat{x}, x^0)$ over nonnegative minimizers $\hat{x}$ of $f(x)$ is therefore equivalent to minimizing $D_h(\hat{x}, x^*)$ over the same vectors. But the solution to the latter problem is obviously $\hat{x} = x^*$. Thus we have shown that the limit of the SMART is the nonnegative minimizer of $KL(Px, y)$ for which the distance $KL(x, x^0)$ is minimized.

The following theorem summarizes the situation with regard to the SMART.

Theorem 8.1 *In the consistent case the SMART converges to the unique nonnegative solution of $y = Px$ for which the distance $\sum_{j=1}^{J} s_j KL(x_j, x_j^0)$ is minimized. In the inconsistent case it converges to the unique nonnegative minimizer of the distance $KL(Px, y)$ for which $\sum_{j=1}^{J} s_j KL(x_j, x_j^0)$ is minimized; if P and every matrix derived from P by deleting columns has full rank then there is a unique nonnegative minimizer of $KL(Px, y)$ and at most $I - 1$ of its entries are nonzero.*

9 Alternating Minimization

As we have seen, the SMART is best derived as an alternating minimization (AM) algorithm. The main reference for alternating minimization is the paper [26] of Csiszár and Tusnády. As the authors of [49] remark, the geometric argument in [26] is “deep, though hard to follow”. As we shall see, all AM methods for which the five-point property of [26] holds fall into the SUMMA class (see [18]).

9.1 Alternating Minimization

The alternating minimization (AM) iteration of Csiszár and Tusnády [26] provides a useful framework for the derivation of iterative optimization algorithms. In this section we discuss their five-point property and use it to obtain a somewhat simpler proof of convergence for their AM algorithm. We then show that all AM algorithms with the five-point property are in the SUMMA class.

9.1.1 The AM Framework

Suppose that P and Q are arbitrary non-empty sets and the function $\Theta(p,q)$ satisfies $-\infty < \Theta(p,q) \leq +\infty$, for each $p \in P$ and $q \in Q$. We assume that, for each $p \in P$, there is $q \in Q$ with $\Theta(p,q) < +\infty$. Therefore, $b = \inf_{p\in P,\, q\in Q} \Theta(p,q) < +\infty$. We assume also that $b > -\infty$; in many applications, the function $\Theta(p,q)$ is non-negative, so this additional assumption is unnecessary. We do not always assume there are $\hat{p} \in P$ and $\hat{q} \in Q$ such that $\Theta(\hat{p},\hat{q}) = b$; when we do assume that such a $\hat{p}$ and $\hat{q}$ exist, we will not assume that $\hat{p}$ and $\hat{q}$ are unique with that property. The objective is to generate a sequence $\{(p^n, q^n)\}$ such that $\Theta(p^n, q^n) \to b$.

9.1.2 The AM Iteration

The general AM method proceeds in two steps: we begin with some q^0, and, having found q^n, we

- **1.** minimize $\Theta(p, q^n)$ over $p \in P$ to get $p = p^{n+1}$, and then
- **2.** minimize $\Theta(p^{n+1}, q)$ over $q \in Q$ to get $q = q^{n+1}$.

In certain applications we consider the special case of alternating cross-entropy minimization. In that case, the vectors p and q are non-negative, and the function $\Theta(p,q)$ will have the value $+\infty$ whenever there is an index j such that $p_j > 0$, but $q_j = 0$. It is important for those particular applications that we select q^0 with all positive entries. We therefore assume, for the general case, that we have selected q^0 so that $\Theta(p, q^0)$ is finite for all p.

The sequence $\{\Theta(p^n, q^n)\}$ is decreasing and bounded below by b, since we have

$$\Theta(p^n, q^n) \geq \Theta(p^{n+1}, q^n) \geq \Theta(p^{n+1}, q^{n+1}). \tag{9.1}$$

Therefore, the sequence $\{\Theta(p^n, q^n)\}$ converges to some $B \geq b$. Without additional assumptions, we can say little more.

We know two things:

$$\Theta(p^{n+1}, q^n) - \Theta(p^{n+1}, q^{n+1}) \geq 0, \tag{9.2}$$

and

$$\Theta(p^n, q^n) - \Theta(p^{n+1}, q^n) \geq 0. \tag{9.3}$$

Equation 9.3 can be strengthened to

$$\Theta(p, q^n) - \Theta(p^{n+1}, q^n) \geq 0. \tag{9.4}$$

We need to make these inequalities more precise.

9.1.3 The Five-Point Property for AM

The five-point property is the following: for all $p \in P$ and $q \in Q$ and $n = 1, 2, ...$

The Five-Point Property

$$\Theta(p, q) + \Theta(p, q^{n-1}) \geq \Theta(p, q^n) + \Theta(p^n, q^{n-1}). \tag{9.5}$$

9.1.4 The Main Theorem for AM

We want to find sufficient conditions for the sequence $\{\Theta(p^n, q^n)\}$ to converge to b, that is, for $B = b$. The following is the main result of [26].

Theorem 9.1 *If the five-point property holds then $B = b$.*

Proof: Suppose that $B > b$. Then there are p' and q' such that $B > \Theta(p', q') \geq b$. From the five-point property we have

$$\Theta(p', q^{n-1}) - \Theta(p^n, q^{n-1}) \geq \Theta(p', q^n) - \Theta(p', q'), \tag{9.6}$$

so that

$$\Theta(p', q^{n-1}) - \Theta(p', q^n) \geq \Theta(p^n, q^{n-1}) - \Theta(p', q') \geq 0. \tag{9.7}$$

All the terms being subtracted can be shown to be finite. It follows that the sequence $\{\Theta(p', q^{n-1})\}$ is decreasing, bounded below, and therefore convergent. The right side of Equation (9.7) must therefore converge to zero, which is a contradiction. We conclude that $B = b$ whenever the five-point property holds in AM. ▮

9.1.5 The Three- and Four-Point Properties

In [26] the five-point property is related to two other properties, the three- and four-point properties. This is a bit peculiar for two reasons: first, as we have just seen, the five-point property is sufficient to prove the main theorem; and second, these other properties involve a second function, $\Delta : P \times P \to [0, +\infty]$, with $\Delta(p, p) = 0$ for all $p \in P$. The three- and four-point properties jointly imply the five-point property, but to get the converse, we need to use the five-point property to define this second function; it can be done, however.

The three-point property is the following:

The Three-Point Property

$$\Theta(p, q^n) - \Theta(p^{n+1}, q^n) \geq \Delta(p, p^{n+1}), \tag{9.8}$$

for all p. The four-point property is the following:

The Four-Point Property

$$\Delta(p,p^{n+1}) + \Theta(p,q) \geq \Theta(p,q^{n+1}), \tag{9.9}$$

for all p and q.

It is clear that the three- and four-point properties together imply the five-point property. We show now that the three-point property and the four-point property are implied by the five-point property. For that purpose we need to define a suitable $\Delta(p,\tilde{p})$. For any p and $\tilde{p}$ in P define

$$\Delta(p,\tilde{p}) = \Theta(p,q(\tilde{p})) - \Theta(p,q(p)), \tag{9.10}$$

where $q(p)$ denotes a member of Q satisfying $\Theta(p,q(p)) \leq \Theta(p,q)$, for all q in Q. Clearly, $\Delta(p,\tilde{p}) \geq 0$ and $\Delta(p,p) = 0$. The four-point property holds automatically from this definition, while the three-point property follows from the five-point property. Therefore, it is sufficient to discuss only the five-point property when speaking of the AM method.

9.2 Alternating Bregman Distance Minimization

The general problem of minimizing $\Theta(p,q)$ is simply a minimization of a real-valued function of two variables, $p \in P$ and $q \in Q$. In many cases the function $\Theta(p,q)$ is a distance between p and q, either $\|p-q\|_2^2$ or $KL(p,q)$. In the case of $\Theta(p,q) = \|p-q\|_2^2$, each step of the alternating minimization algorithm involves an orthogonal projection onto a closed convex set; both projections are with respect to the same Euclidean distance function. In the case of cross-entropy minimization, we first project q^n onto the set P by minimizing the distance $KL(p,q^n)$ over all $p \in P$, and then project p^{n+1} onto the set Q by minimizing the distance function $KL(p^{n+1},q)$. This suggests the possibility of using alternating minimization with respect to more general distance functions. We shall focus on Bregman distances.

9.2.1 Bregman Distances

Let $f : \mathbb{R}^N \to \mathbb{R}$ be a Bregman function [8, 24, 10], and so $f(x)$ is convex on its domain and differentiable in the interior of its domain. Then, for x in the domain and z in the interior, we define the Bregman distance $D_f(x,z)$ by

$$D_f(x,z) = f(x) - f(z) - \langle \nabla f(z), x - z \rangle. \tag{9.11}$$

For example, the KL distance is a Bregman distance with associated Bregman function

$$f(x) = \sum_{j=1}^{J} x_j \log x_j \; - x_j. \tag{9.12}$$

Suppose now that $f(x)$ is a Bregman function and P and Q are closed convex subsets of the interior of the domain of $f(x)$. Let p^{n+1} minimize $D_f(p, q^n)$ over all $p \in P$. It follows then that

$$\langle \nabla f(p^{n+1}) - \nabla f(q^n), p - p^{n+1} \rangle \geq 0, \tag{9.13}$$

for all $p \in P$. Since

$$D_f(p, q^n) - D_f(p^{n+1}, q^n) =$$

$$D_f(p, p^{n+1}) + \langle \nabla f(p^{n+1}) - \nabla f(q^n), p - p^{n+1} \rangle, \tag{9.14}$$

it follows that the three-point property holds, with

$$\Theta(p, q) = D_f(p, q), \tag{9.15}$$

and

$$\Delta(p, \hat{p}) = D_f(p, \tilde{p}). \tag{9.16}$$

To get the four-point property we need to restrict D_f somewhat; we assume from now on that $D_f(p, q)$ is jointly convex, that is, it is convex in the combined vector variable (p, q) (see [3]). Now we can invoke a lemma due to Eggermont and LaRiccia [27].

9.2.2 The Eggermont-LaRiccia Lemma

Lemma 9.1 *Suppose that the Bregman distance $D_f(p, q)$ is jointly convex. Then it has the four-point property.*

Proof: By joint convexity we have

$$D_f(p, q) - D_f(p^n, q^n) \geq$$

$$\langle \nabla_1 D_f(p^n, q^n), p - p^n \rangle + \langle \nabla_2 D_f(p^n, q^n), q - q^n \rangle,$$

where ∇_1 denotes the gradient with respect to the first vector variable. Since q^n minimizes $D_f(p^n, q)$ over all $q \in Q$, we have

$$\langle \nabla_2 D_f(p^n, q^n), q - q^n \rangle \geq 0,$$

for all q. Also,

$$\langle \nabla_1(p^n, q^n), p - p^n \rangle = \langle \nabla f(p^n) - \nabla f(q^n), p - p^n \rangle.$$

It follows that

$$D_f(p, q^n) - D_f(p, p^n) = D_f(p^n, q^n) + \langle \nabla_1(p^n, q^n), p - p^n \rangle$$

$$\leq D_f(p, q) - \langle \nabla_2 D_f(p^n, q^n), q - q^n \rangle \leq D_f(p, q).$$

Therefore, we have

$$D_f(p, p^n) + D_f(p, q) \geq D_f(p, q^n).$$

This is the four-point property. ∎

We now know that the alternating minimization method works for any Bregman distance that is jointly convex. This includes the Euclidean and the KL distances.

9.3 Minimizing a Proximity Function

We present now an example of alternating Bregman distance minimization taken from [19]. The problem is the *convex feasibility problem* (CFP), to find a member of the intersection $C \subseteq \mathbb{R}^J$ of finitely many closed convex sets C_i, $i = 1, ..., I$, or, failing that, to minimize the proximity function

$$F(x) = \sum_{i=1}^{I} D_i(\overleftarrow{P}_i x, x), \tag{9.17}$$

where f_i are Bregman functions for which D_i, the associated Bregman distance, is jointly convex, and $\overleftarrow{P}_i x$ are the *left* Bregman projection of x onto the set C_i, that is, $\overleftarrow{P}_i x \in C_i$ and $D_i(\overleftarrow{P}_i x, x) \leq D_i(z, x)$, for all $z \in C_i$. Because each D_i is jointly convex, the function $F(x)$ is convex.

The problem can be formulated as an alternating minimization, where $P \subseteq \mathbb{R}^{IJ}$ is the product set $P = C_1 \times C_2 \times ... \times C_I$. A typical member of P has the form $p = (c^1, c^2, ..., c^I)$, where $c^i \in C_i$, and $Q \subseteq \mathbb{R}^{IJ}$ is the *diagonal* subset, meaning that the elements of Q are the I-fold product of a single x; that is $Q = \{d(x) = (x, x, ..., x) \in \mathbb{R}^{IJ}\}$. We then take

$$\Theta(p, q) = \sum_{i=1}^{I} D_i(c^i, x), \tag{9.18}$$

and $\Delta(p, \tilde{p}) = \Theta(p, \tilde{p})$.

In [21] a similar iterative algorithm was developed for solving the CFP, using the same sets P and Q, but using alternating projection, rather than alternating minimization. Now it is not necessary that the Bregman distances be jointly convex. Each iteration of their algorithm involves two steps:

- 1. minimize $\sum_{i=1}^{I} D_i(c^i, x^n)$ over $c^i \in C_i$, obtaining $c^i = \overleftarrow{P}_i x^n$, and then
- 2. minimize $\sum_{i=1}^{I} D_i(x, \overleftarrow{P}_i x^n)$.

Because this method is an alternating projection approach, it converges only when the CFP has a solution, whereas the previous alternating minimization method minimizes $F(x)$, even when the CFP has no solution.

9.4 Right and Left Projections

Because Bregman distances D_f are not generally symmetric, we can speak of *right* and *left* Bregman projections onto a closed convex set. For any allowable vector x, the *left* Bregman projection of x onto C, if it exists, is the vector $\overleftarrow{P}_C x \in C$ satisfying the inequality $D_f(\overleftarrow{P}_C x, x) \leq D_f(c, x)$, for all $c \in C$. Similarly, the *right* Bregman projection is the vector $\overrightarrow{P}_C x \in C$ satisfying the inequality $D_f(x, \overrightarrow{P}_C x) \leq D_f(x, c)$, for any $c \in C$.

The alternating minimization approach described above to minimize the proximity function

$$F(x) = \sum_{i=1}^{I} D_i(\overleftarrow{P}_i x, x) \tag{9.19}$$

can be viewed as an alternating projection method, but employing both right and left Bregman projections.

Consider the problem of finding a member of the intersection of two closed convex sets C and D. We could proceed as follows: having found x^n, minimize $D_f(x^n, d)$ over all $d \in D$, obtaining $d = \overrightarrow{P}_D x^n$, and then minimize $D_f(c, \overrightarrow{P}_D x^n)$ over all $c \in C$, obtaining $c = x^{n+1} = \overleftarrow{P}_C \overrightarrow{P}_D x^n$. The objective of this algorithm is to minimize $D_f(c, d)$ over all $c \in C$ and $d \in D$; such a minimum may not exist, of course.

In [4] the authors note that the alternating minimization algorithm of [19] involves right and left Bregman projections, which suggests to them iterative methods involving a wider class of operators that they call "Bregman retractions".

9.5 More Proximity Function Minimization

Proximity function minimization and right and left Bregman projections play a role in a variety of iterative algorithms. We survey several of them in this section.

9.5.1 Cimmino's Algorithm

Our objective here is to find an exact or approximate solution of the system of I linear equations in J unknowns, written $Ax = b$. For each i let

$$C_i = \{z|(Az)_i = b_i\}, \tag{9.20}$$

and $P_i x$ be the orthogonal projection of x onto C_i. Then

$$(P_i x)_j = x_j + \alpha_i A_{ij}(b_i - (Ax)_i), \tag{9.21}$$

where

$$(\alpha_i)^{-1} = \sum_{j=1}^{J} A_{ij}^2. \tag{9.22}$$

Let

$$F(x) = \sum_{i=1}^{I} \|P_i x - x\|_2^2. \tag{9.23}$$

Using alternating minimization on this proximity function gives Cimmino's algorithm, with the iterative step

$$x_j^{n+1} = x_j^n + \frac{1}{I}\sum_{i=1}^{I} \alpha_i A_{ij}(b_i - (Ax^n)_i). \tag{9.24}$$

9.5.2 Simultaneous Projection for Convex Feasibility

Now we let C_i be any closed convex subsets of $\mathbb{R}^J$ and define $F(x)$ as in the previous section. Again, we apply alternating minimization. The iterative step of the resulting algorithm is

$$x^{n+1} = \frac{1}{I}\sum_{i=1}^{I} P_i x^n. \tag{9.25}$$

The objective here is to minimize $F(x)$, if there is a minimum.

9.5.3 The Bauschke-Combettes-Noll Problem

In [5] Bauschke, Combettes and Noll consider the following problem: minimize the function

$$\Theta(p,q) = \Lambda(p,q) = \phi(p) + \psi(q) + D_f(p,q), \tag{9.26}$$

where ϕ and ψ are convex on $\mathbb{R}^J$, $D = D_f$ is a Bregman distance, and $P = Q$ is the interior of the domain of f. They assume that

$$b = \inf_{(p,q)} \Lambda(p,q) > -\infty, \tag{9.27}$$

and seek a sequence $\{(p^n, q^n)\}$ such that $\{\Lambda(p^n, q^n)\}$ converges to b. The sequence is obtained by the AM method, as in our previous discussion. They prove that, if the Bregman distance is jointly convex, then $\{\Lambda(p^n, q^n)\} \downarrow b$. In this subsection we obtain this result by showing that $\Lambda(p,q)$ has the five-point property whenever $D = D_f$ is jointly convex. Our proof is loosely based on the proof of the Eggermont-LaRiccia lemma.

The five-point property for $\Lambda(p,q)$ is

$$\Lambda(p, q^{n-1}) - \Lambda(p^n, q^{n-1}) \geq \Lambda(p, q^n) - \Lambda(p,q). \tag{9.28}$$

A simple calculation shows that the inequality in (9.28) is equivalent to

$$\Lambda(p,q) - \Lambda(p^n, q^n) \geq$$

$$D(p, q^n) + D(p^n, q^{n-1}) - D(p, q^{n-1}) - D(p^n, q^n). \tag{9.29}$$

By the joint convexity of $D(p,q)$ and the convexity of ϕ and ψ we have

$$\Lambda(p,q) - \Lambda(p^n, q^n) \geq$$

$$\langle \nabla_p \Lambda(p^n, q^n), p - p^n \rangle + \langle \nabla_q \Lambda(p^n, q^n), q - q^n \rangle, \tag{9.30}$$

where $\nabla_p \Lambda(p^n, q^n)$ denotes the gradient of $\Lambda(p,q)$, with respect to p, evaluated at (p^n, q^n).

Since q^n minimizes $\Lambda(p^n, q)$, it follows that

$$\langle \nabla_q \Lambda(p^n, q^n), q - q^n \rangle = 0, \tag{9.31}$$

for all q. Therefore,

$$\Lambda(p,q) - \Lambda(p^n,q^n) \geq \left\langle \nabla_p \Lambda(p^n,q^n), p - p^n \right\rangle . \tag{9.32}$$

We have

$$\langle \nabla_p \Lambda(p^n,q^n), p - p^n \rangle =$$

$$\langle \nabla f(p^n) - \nabla f(q^n), p - p^n \rangle + \langle \nabla \phi(p^n), p - p^n \rangle. \tag{9.33}$$

Since p^n minimizes $\Lambda(p,q^{n-1})$, we have

$$\nabla_p \Lambda(p^n,q^{n-1}) = 0, \tag{9.34}$$

or

$$\nabla \phi(p^n) = \nabla f(q^{n-1}) - \nabla f(p^n), \tag{9.35}$$

so that

$$\langle \nabla_p \Lambda(p^n,q^n), p - p^n \rangle = \langle \nabla f(q^{n-1}) - \nabla f(q^n), p - p^n \rangle \tag{9.36}$$

$$= D(p,q^n) + D(p^n,q^{n-1}) - D(p,q^{n-1}) - D(p^n,q^n). \tag{9.37}$$

Using (9.32) we obtain the inequality in (9.29). This shows that $\Lambda(p,q)$ has the five-point property whenever the Bregman distance $D = D_f$ is jointly convex.

From our previous discussion of AM, we conclude that the sequence $\{\Lambda(p^n,q^n)\}$ converges to b; this is Corollary 4.3 of [5].

In [20] it was shown that, in certain cases, the expectation maximization maximum likelihood (EM) method involves alternating minimization of a function of the form $\Lambda(p,q)$.

If $\psi = 0$, then $\{\Lambda(p^n,q^n)\}$ converges to b, even without the assumption that the distance D_f is jointly convex. In such cases, $\Lambda(p,q)$ has the form of the objective function in proximal minimization and therefore the problem falls into the SUMMA class (see Lemma 6.1).

9.6 AM as SUMMA

We show now that the SUMMA class of sequential unconstrained minimization methods includes all the AM methods for which the five-point property holds.

9.7 Reformulating AM as SUMMA

For each p in the set P, define $q(p)$ in Q as a member of Q for which $\Theta(p,q(p)) \leq \Theta(p,q)$, for all $q \in Q$. Let $f(p) = \Theta(p,q(p))$.

At the nth step of AM we minimize

$$G_n(p) = \Theta(p,q^{n-1}) = \Theta(p,q(p)) + \Big(\Theta(p,q^{n-1}) - \Theta(p,q(p))\Big) \tag{9.38}$$

to get p^n. With

$$g_n(p) = \Big(\Theta(p,q^{n-1}) - \Theta(p,q(p))\Big) \geq 0, \tag{9.39}$$

we can write

$$G_n(p) = f(p) + g_n(p). \tag{9.40}$$

According to the five-point property, we have

$$G_n(p) - G_n(p^n) \geq \Theta(p,q^n) - \Theta(p,q(p)) = g_{n+1}(p). \tag{9.41}$$

It follows that AM is a member of the SUMMA class.

10 Appendix One: Theorem 2.1 Revisited

10.1 Improving Theorem 2.1

The proof of Theorem 2.1 made use of the restriction that γ be in the interval $(0, \frac{1}{L})$. For convergence, we need only that γ be in the interval $(0, \frac{2}{L})$, as the following theorem asserts.

Theorem 10.1 *Let $f : \mathbb{R}^J \to \mathbb{R}$ be differentiable, with L-Lipschitz continuous gradient. For γ in the interval $(0, \frac{2}{L})$ the sequence $\{x^k\}$ given by Equation (2.9) converges to a minimizer of f, whenever minimizers exist.*

10.2 Properties of the Gradient

Theorem 10.2 *Let $g : \mathbb{R}^J \to \mathbb{R}$ be differentiable. The following are equivalent:*

- **1)** *$g(x)$ is convex;*
- **2)** *for all a and b we have*

$$g(b) \geq g(a) + \langle \nabla g(a), b - a \rangle\,; \tag{10.1}$$

- **3)** *for all a and b we have*

$$\langle \nabla g(b) - \nabla g(a), b-a\rangle \geq 0. \tag{10.2}$$

Because the operator ∇f is L-Lipschitz continuous, the gradient of the function $g(x) = \frac{1}{L}f(x)$ is non-expansive, that is,

$$\|\nabla g(x) - \nabla g(y)\| \leq \|x-y\|, \tag{10.3}$$

for all x and y.

10.3 Non-expansive gradients

In [31] Golshtein and Tretyakov prove the following theorem.

Theorem 10.3 *Let $g : \mathbb{R}^J \to \mathbb{R}$ be convex and differentiable. The following are equivalent:*

- **1)**

$$||\nabla g(x) - \nabla g(y)||_2 \leq ||x-y||_2; \tag{10.4}$$

- **2)**

$$g(x) \geq g(y) + \langle \nabla g(y), x-y\rangle + \frac{1}{2}||\nabla g(x) - \nabla g(y)||_2^2; \tag{10.5}$$

and

- **3)**

$$\langle \nabla g(x) - \nabla g(y), x-y\rangle \geq ||\nabla g(x) - \nabla g(y)||_2^2. \tag{10.6}$$

Proof: The only non-trivial step in the proof is showing that Inequality (10.4) implies Inequality (10.5). From Theorem 10.2 we see that Inequality (10.4) implies that the function $h(x) = \frac{1}{2}\|x\|^2 - g(x)$ is convex, and that

$$\frac{1}{2}\|x-y\|^2 \geq g(x) - g(y) - \langle \nabla g(y), x-y\rangle\,,$$

for all x and y. Now fix y and define

$$d(z) = D_g(z,y) = g(z) - g(y) - \langle \nabla g(y), z-y\rangle,$$

for all z. Since the function $g(z)$ is convex, so is $d(z)$. Since

$$\nabla d(z) = \nabla g(z) - \nabla g(y),$$

it follows from Inequality (10.4) that

$$\|\nabla d(z) - \nabla d(x)\| \leq \|z - x\|,$$

for all x and z. Then, from our previous calculations, we may conclude that

$$\frac{1}{2}\|z - x\|^2 \geq d(z) - d(x) - \langle \nabla d(x), z - x \rangle\,,$$

for all z and x.

Now let x be arbitrary and

$$z = x - \nabla g(x) + \nabla g(y).$$

Then

$$0 \leq d(z) \leq d(x) - \frac{1}{2}\|\nabla g(x) - \nabla g(y)\|^2.$$

This completes the proof. ▮

Now we can prove Theorem 10.1.

10.4 Proof of Theorem 10.1

Let $f(z) \leq f(x)$, for all x; then $\nabla f(z) = 0$. Then

$$\|z - x^k\|^2 = \|z - x^{k-1} + \gamma \nabla f(x^{k-1})\|^2 =$$

$$\|z - x^{k-1}\|^2 - 2\gamma\langle \nabla f(z) - \nabla f(x^{k-1}), z - x^{k-1}\rangle \ + \gamma^2\|\nabla f(z) - \nabla f(x^{k-1})\|^2.$$

Therefore,

$$\|z-x^{k-1}\|^2 - \|z-x^k\|^2 = 2\gamma L\langle \nabla g(z) - \nabla g(x^{k-1}), z - x^{k-1}\rangle - \gamma^2 L^2\|\nabla g(z) - \nabla g(x^{k-1})\|^2 \geq$$

$$(2\gamma L - \gamma^2 L^2)\|\nabla g(z) - \nabla g(x^{k-1})\|^2.$$

Since $0 < \gamma < \frac{2}{L}$, the sequence $\{\|z - x^k\|\}$ is decreasing and the sequence $\{\|\nabla f(z) - \nabla f(x^k)\|\}$ converges to zero. There is then a subsequence of $\{x^k\}$ converging to some x^* with $\nabla f(x^*) = 0$, so that x^* is a minimizer of f. Replacing the generic z with x^*, we find that the sequence $\{x^k\}$ converges to x^*. This completes the proof. ▮

We can interpret Theorem 10.3 as saying that, if g is convex and differentiable, and its gradient is non-expansive in the 2-norm, then the gradient of g is a firmly non-expansive operator [15].

If $f : \mathbb{R}^J \rightarrow \mathbb{R}$ is convex and differentiable, and its gradient is L-Lipschitz continuous, that is,

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2,$$

then the gradient of $g(x) = \frac{1}{L}f(x)$ is a firmly non-expansive operator. It then follows that the operator $I - \gamma\nabla f$ is an averaged operator, for any γ in the interval $(0, \frac{2}{L})$ [15].

11 Appendix Two: Bregman-Legendre Functions

In [2] Bauschke and Borwein show convincingly that the Bregman-Legendre functions provide the proper context for the discussion of Bregman projections onto closed convex sets. The summary here follows closely the discussion given in [2].

11.1 Essential Smoothness and Essential Strict Convexity

Following [47] we say that a closed proper convex function f is *essentially smooth* if int$D$ is not empty, f is differentiable on int$D$ and $x^n \in \text{int}D$, with $x^n \rightarrow x \in \text{bd}D$, implies that $||\nabla f(x^n)||_2 \rightarrow +\infty$. Here int$D$ and bd$D$ denote the interior and boundary of the set D. A closed proper convex function f is *essentially strictly convex* if f is strictly convex on every convex subset of dom ∂f.

The closed proper convex function f is essentially smooth if and only if the subdifferential $\partial f(x)$ is empty for $x \in \text{bd}D$ and is $\{\nabla f(x)\}$ for $x \in \text{int}D$ (so f is differentiable on int$D$) if and only if the function f^* is essentially strictly convex.

Definition 11.1 *A closed proper convex function f is said to be a* Legendre function *if it is both essentially smooth and essentialy strictly convex.*

So f is Legendre if and only if its conjugate function is Legendre, in which case the gradient operator ∇f is a topological isomorphism with ∇f^* as its inverse. The gradient operator ∇f maps int dom f onto int dom f^*. If int dom $f^* = \mathbb{R}^J$ then the range of ∇f is $\mathbb{R}^J$ and the equation $\nabla f(x) = y$ can be solved for every $y \in \mathbb{R}^J$. In order for int dom $f^* = \mathbb{R}^J$ it is necessary and sufficient that the Legendre function f be *super-coercive*, that is,

$$\lim_{||x||_2 \rightarrow +\infty} \frac{f(x)}{||x||_2} = +\infty. \tag{11.1}$$

If the effective domain of f is bounded, then f is super-coercive and its gradient operator is a mapping onto the space $\mathbb{R}^J$.

11.2 Bregman Projections onto Closed Convex Sets

Let f be a closed proper convex function that is differentiable on the nonempty set $\mathrm{int}D$. The corresponding *Bregman distance* $D_f(x,z)$ is defined for $x \in \mathbb{R}^J$ and $z \in \mathrm{int}D$ by

$$D_f(x,z) = f(x) - f(z) - \langle \nabla f(z), x - z\rangle. \tag{11.2}$$

Note that $D_f(x,z) \geq 0$ always and that $D_f(x,z) = +\infty$ is possible. If f is essentially strictly convex then $D_f(x,z) = 0$ implies that $x = z$.

Let K be a nonempty closed convex set with $K \cap \mathrm{int}D \neq \emptyset$. Pick $z \in \mathrm{int}D$. The *Bregman projection* of z onto K, with respect to f, is

$$P_K^f(z) = \mathrm{argmin}_{x\in K\cap D} D_f(x,z). \tag{11.3}$$

If f is essentially strictly convex, then $P_K^f(z)$ exists. If f is strictly convex on D then $P_K^f(z)$ is unique. If f is Legendre, then $P_K^f(z)$ is uniquely defined and is in $\mathrm{int}D$; this last condition is sometimes called *zone consistency*.

Example: Let $J = 2$ and $f(x)$ be the function that is equal to one-half the norm squared on D, the nonnegative quadrant, $+\infty$ elsewhere. Let K be the set $K = \{(x_1,x_2)|x_1 + x_2 = 1\}$. The Bregman projection of $(2,1)$ onto K is $(1,0)$, which is not in $\mathrm{int}D$. The function f is not essentially smooth, although it is essentially strictly convex. Its conjugate is the function f^* that is equal to one-half the norm squared on D and equal to zero elsewhere; it is essentially smooth, but not essentially strictly convex.

If f is Legendre, then $P_K^f(z)$ is the unique member of $K \cap \mathrm{int}D$ satisfying the inequality

$$\langle \nabla f(P_K^f(z)) - \nabla f(z), P_K^f(z) - c\rangle \geq 0, \tag{11.4}$$

for all $c \in K$. From this we obtain the *Bregman Inequality*:

$$D_f(c,z) \geq D_f(c, P_K^f(z)) + D_f(P_K^f(z), z), \tag{11.5}$$

for all $c \in K$.

11.3 Bregman-Legendre Functions

Following Bauschke and Borwein [2], we say that a Legendre function f is a *Bregman-Legendre* function if the following properties hold:

B1: for x in D and any $a > 0$ the set $\{z|D_f(x, z) \leq a\}$ is bounded.
B2: if x is in D but not in int$D$, for each positive integer n, y^n is in int$D$ with $y^n \to y \in$ bd$D$ and if $\{D_f(x, y^n)\}$ remains bounded, then $D_f(y, y^n) \to 0$, so that $y \in D$.
B3: if x^n and y^n are in int$D$, with $x^n \to x$ and $y^n \to y$, where x and y are in D but not in int$D$, and if $D_f(x^n, y^n) \to 0$ then $x = y$.

Bauschke and Borwein then prove that Bregman's SGP method converges to a member of K provided that one of the following holds: 1) f is Bregman-Legendre; 2) $K \cap \text{int}D \neq \emptyset$ and dom f^* is open; or 3) dom f and dom f^* are both open.

The Bregman functions form a class closely related to the Bregman-Legendre functions. For details see [10].

11.4 Useful Results about Bregman-Legendre Functions

The following results are proved in somewhat more generality in [2].
R1: If $y^n \in$ int dom f and $y^n \to y \in$ int dom f, then $D_f(y, y^n) \to 0$.
R2: If x and $y^n \in$ int dom f and $y^n \to y \in$ bd dom f, then $D_f(x, y^n) \to +\infty$.
R3: If $x^n \in D$, $x^n \to x \in D$, $y^n \in$ int D, $y^n \to y \in D$, $\{x, y\} \cap$ int $D \neq \emptyset$ and $D_f(x^n, y^n) \to 0$, then $x = y$ and $y \in$ int D.
R4: If x and y are in D, but are not in int D, $y^n \in$ int D, $y^n \to y$ and $D_f(x, y^n) \to 0$, then $x = y$.
As a consequence of these results we have the following.
R5: If $\{D_f(x, y^n)\} \to 0$, for $y^n \in$ int D and $x \in \mathbb{R}^J$, then $\{y^n\} \to x$.

Proof of R5: Since $\{D_f(x, y^n)\}$ is eventually finite, we have $x \in D$. By Property B1 above it follows that the sequence $\{y^n\}$ is bounded; without loss of generality, we assume that $\{y^n\} \to y$, for some $y \in \overline{D}$. If x is in int D, then, by result R2 above, we know that y is also in int D. Applying result R3, with $x^n = x$, for all n, we conclude that $x = y$. If, on the other hand, x is in D, but not in int D, then y is in D, by result R2. There are two cases to consider: 1) y is in int D; 2) y is not in int D. In case 1) we have $D_f(x, y^n) \to D_f(x, y) = 0$, from which it follows that $x = y$. In case 2) we apply result R4 to conclude that $x = y$. ∎

References

1. Bauschke, H., and Borwein, J. (1996) "On projection algorithms for solving convex feasibility problems." *SIAM Review*, **38 (3)**, pp. 367–426.

2. Bauschke, H., and Borwein, J. (1997) "Legendre functions and the method of random Bregman projections." *Journal of Convex Analysis*, **4**, pp. 27–67.

3. Bauschke, H., and Borwein, J. (2001) "Joint and separate convexity of the Bregman distance." in *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications*, edited by D. Butnariu, Y. Censor and S. Reich, pp. 23–36, Studies in Computational Mathematics 8. Amsterdam: Elsevier Publ.

4. Bauschke, H., and Combettes, P. (2003) "Iterating Bregman retractions." *SIAM Journal on Optimization*, **13**, pp. 1159–1173.

5. Bauschke, H., Combettes, P., and Noll, D. (2006) "Joint minimization with alternating Bregman proximity operators." *Pacific Journal of Optimization*, **2**, pp. 401–424.

6. Becker, M., Yang, I., and Lange, K. (1997) "EM algorithms without missing data." *Stat. Methods Med. Res.*, **6**, pp. 38–54.

7. Bertero, M., and Boccacci, P. (1998) *Introduction to Inverse Problems in Imaging* Bristol, UK: Institute of Physics Publishing.

8. 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 Computational Mathematics and Mathematical Physics* **7**, pp. 200–217.

9. Bruck, R., and Reich, S. (1977) "Nonexpansive projections and resolvents of accretive operators in Banach spaces." *Houston J. of Mathematics* **3**, pp. 459–470.

10. Butnariu, D., Byrne, C., and Censor, Y. (2003) "Redundant axioms in the definition of Bregman functions." *Journal of Convex Analysis*, **10**, pp. 245–254.

11. Byrne, C. (1993) "Iterative image reconstruction algorithms based on cross-entropy minimization."*IEEE Transactions on Image Processing* **IP-2**, pp. 96–103.

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

13. Byrne, C. (2001) "Bregman-Legendre multi-distance projection algorithms for convex feasibility and optimization." in *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications*, edited by D. Butnariu, Y. Censor and S. Reich, pp. 87-100, Studies in Computational Mathematics 8. Amsterdam: Elsevier Publ.

14. Byrne, C. (2002) "Iterative oblique projection onto convex sets and the split feasibility problem."*Inverse Problems* **18**, pp. 441–453.

15. Byrne, C. (2004) "A unified treatment of some iterative algorithms in signal processing and image reconstruction."*Inverse Problems* **20**, pp. 103–120.

16. Byrne, C. (2008) "Sequential unconstrained minimization algorithms for constrained optimization." *Inverse Problems*, **24(1)**, article no. 015013.

17. Byrne, C. (2011) *A First Course in Optimization*, available as a pdf file at my web site.

18. Byrne, C. (2012) "Alternating and sequential unconstrained minimization algorithms." *Journal of Optimization Theory and Applications*, **156(2)**, and DOI 10.1007/s1090134-2.

19. Byrne, C., and Censor, Y. (2001) "Proximity function minimization using multiple Bregman projections, with applications to split feasibility and Kullback-Leibler distance minimization." *Annals of Operations Research*, **105**, pp. 77–98.

20. Byrne, C., and Eggermont, P. (2011) "EM Algorithms." in *Handbook of Mathematical Methods in Imaging*, Otmar Scherzer, ed., Springer-Science.

21. Censor, Y. and Elfving, T. (1994) "A multi-projection algorithm using Bregman projections in a product space." *Numerical Algorithms*, **8** 221–239.

22. Censor, Y., Bortfeld, T., Martin, B., and Trofimov, A. "A unified approach for inversion problems in intensity-modulated radiation therapy." *Physics in Medicine and Biology* 51 (2006), 2353-2365.

23. Censor, Y., Elfving, T., Kopf, N., and Bortfeld, T. (2005) “The multiple-sets split feasibility problem and its application for inverse problems.” *Inverse Problems*, **21** , pp. 2071-2084.

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

25. Combettes, P., and Wajs, V. (2005) “Signal recovery by proximal forward-backward splitting.” *Multiscale Modeling and Simulation*, **4(4)**, pp. 1168–1200.

26. Csiszár, I. and Tusnády, G. (1984) “Information geometry and alternating minimization procedures.” *Statistics and Decisions* **Supp. 1**, pp. 205–237.

27. Eggermont, P., and LaRiccia, V. (2001) *Maximum Penalized Likelihood Estimation.* New York: Springer.

28. Fiacco, A., and McCormick, G. (1990) *Nonlinear Programming: Sequential Unconstrained Minimization Techniques.* Philadelphia, PA: SIAM Classics in Mathematics (reissue).

29. Geman, S., and Geman, D. (1984) “Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images.”*IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-6**, pp. 721–741.

30. Goebel, K., and Reich, S. (1984) *Uniform Convexity, Hyperbolic Geometry, and Nonexpansive Mappings*, New York: Dekker.

31. Golshtein, E., and Tretyakov, N. (1996) *Modified Lagrangians and Monotone Maps in Optimization.* New York: John Wiley and Sons, Inc.

32. Krasnosel’skiĭ, M. (1955) “Two remarks on the method of successive approximations” (in Russian). *Uspekhi Mathematicheskikh Nauk*, **10**, pp. 123–127.

33. Kullback, S. and Leibler, R. (1951) “On information and sufficiency.”*Annals of Mathematical Statistics* **22**, pp. 79–86.

34. Landweber, L. (1951) “An iterative formula for Fredholm integral equations of the first kind.”*Amer. J. of Math.* **73**, pp. 615–624.

35. Lange, K. and Carson, R. (1984) “EM reconstruction algorithms for emission and transmission tomography.”*Journal of Computer Assisted Tomography* **8**, pp. 306–316.

36. Lange, K., Bahn, M. and Little, R. (1987) "A theoretical study of some maximum likelihood algorithms for emission and transmission tomography."*IEEE Trans. Med. Imag.* **MI-6(2)**, pp. 106–114.

37. Lange, K., Hunter, D., and Yang, I. (2000) "Optimization transfer using surrogate objective functions (with discussion)." *J. Comput. Graph. Statist.*, **9**, pp. 1–20.

38. Mann, W. (1953) "Mean value methods in iteration." *Proceedings of the American Mathematical Society*, **4**, pp. 506–510.

39. Masad, E., and Reich, S. (2007) "A note on the multiple-set split convex feasibility problem in Hilbert space." *J. Nonlinear Convex Analysis*, **8**, pp. 367–371.

40. Moreau, J.-J. (1962) "Fonctions convexes duales et points proximaux dans un espace hilbertien." *C.R. Acad. Sci. Paris Sér. A Math.*, **255**, pp. 2897–2899.

41. Moreau, J.-J. (1963) "Propriétés des applications 'prox'." *C.R. Acad. Sci. Paris Sér. A Math.*, **256**, pp. 1069–1071.

42. Moreau, J.-J. (1965) "Proximité et dualité dans un espace hilbertien." *Bull. Soc. Math. France*, **93**, pp. 273–299.

43. Nash, S. and Sofer, A. (1996) *Linear and Nonlinear Programming.* New York: McGraw-Hill.

44. Nesterov, Y., and Nemirovski, A. (1994) *Interior-Point Polynomial Algorithms in Convex Programming.* Philadelphia, PA: SIAM Studies in Applied Mathematics.

45. Ortega, J., and Rheinboldt, W. (2000) *Iterative Solution of Nonlinear Equations in Several Variables*, Classics in Applied Mathematics, 30. Philadelphia, PA: SIAM, 2000

46. Renegar, J. (2001) *A Mathematical View of Interior-Point Methods in Convex Optimization.* Philadelphia, PA: SIAM (MPS-SIAM Series on Optimization).

47. Rockafellar, R. (1970) *Convex Analysis.* Princeton, NJ: Princeton University Press.

48. Shepp, L., and Vardi, Y. (1982) "Maximum likelihood reconstruction for emission tomography." *IEEE Transactions on Medical Imaging*, **MI-1**, pp. 113–122.

49. Vardi, Y., Shepp, L.A. and Kaufman, L. (1985) “A statistical model for positron emission tomography.” *Journal of the American Statistical Association* **80**, pp. 8–20.