A Note on Iterative Optimization

Charles L. Byrne
Department of Mathematical Sciences
University of Massachusetts Lowell
Lowell, MA 01854

March 18, 2012

Abstract

An elementary proof is given for the following theorem.

Theorem: Let $f : R^J \to R$ be convex and differentiable, with ∇f L-Lipschitz. Let C be any closed, convex subset of R^J. For $0 < \gamma < \frac{1}{L}$, let $T = P_C(I - \gamma\nabla f)$. If T has fixed points, then the sequence $\{x^k\}$ given by $x^k = Tx^{k-1}$ converges to a fixed point of T.

Any fixed point of T minimizes the function $f(x)$ over the set C. It is a consequence of the Krasnoselskii-Mann Theorem for averaged operators that convergence holds for $0 < \gamma < \frac{2}{L}$. The proof given here employs sequential unconstrained minimization and avoids using the non-trivial results that, because the operator $\frac{1}{L}\nabla f$ is non-expansive, it is firmly non-expansive, and that the product of averaged operators is averaged.

Several applications of the theorem are given, including the proof of convergence of two interior-point algorithms for minimizing $f(x)$ over x with $Ax = b$.

1 The Convergence Theorem

We provide an elementary proof of the following theorem:

Theorem 1.1 *Let $f : R^J \to R$ be convex and differentiable, with ∇f L-Lipschitz. Let C be any closed, convex subset of R^J. For $0 < \gamma < \frac{1}{L}$, let $T = P_C(I - \gamma\nabla f)$. If T has fixed points, then the sequence $\{x^k\}$ given by $x^k = Tx^{k-1}$ converges to a fixed point of T.*

The iterative step is given by

$$x^k = P_C(x^{k-1} - \gamma\nabla f(x^{k-1})). \tag{1.1}$$

It is a consequence of the Krasnoselskii-Mann Theorem for averaged operators that convergence holds for $0 < \gamma < \frac{2}{L}$. The proof given here employs sequential unconstrained minimization and avoids using the non-trivial results that, because the operator $\frac{1}{L}\nabla f$ is non-expansive, it is firmly non-expansive [8], and that the product of averaged operators is again averaged [1].

2 Sequential Unconstrained Optimization

Sequential unconstrained optimization algorithms can be used to minimize a function $f : R^J \to (-\infty, \infty]$ over a (not necessarily proper) subset C of R^J [7]. At the kth step of a *sequential unconstrained minimization* method we obtain x^k by minimizing the function

$$G_k(x) = f(x) + g_k(x), \tag{2.1}$$

where the auxiliary function $g_k(x)$ is appropriately chosen. If C is a proper subset of R^J we may force $g_k(x) = +\infty$ for x not in C, as in the barrier-function methods; then each x^k will lie in C. The objective is then to select the $g_k(x)$ so that the sequence $\{x^k\}$ converges to a solution of the problem, or failing that, at least to have the sequence $\{f(x^k)\}$ converging to the infimum of $f(x)$ over x in C.

Our main focus in this paper is the use of sequential unconstrained optimization algorithms to obtain iterative methods in which each iterate can be obtained in closed form. Now the $g_k(x)$ are selected not to impose a constraint, but to facilitate computation.

3 SUMMA

In [5] we presented a particular class of sequential unconstrained minimization methods called SUMMA. As we showed in that paper, this class is broad enough to contain barrier-function methods, proximal minimization methods, and the simultaneous multiplicative algebraic reconstruction technique (SMART). By reformulating the problem, the penalty-function methods can also be shown to be members of the SUMMA class. Any alternating minimization (AM) problem with the five-point property [6] can be reformulated as a SUMMA problem; therefore the *expectation maximization maximum likelihood* (EMML) algorithm for Poisson data, which is such an AM algorithm, must also be a SUMMA algorithm.

For a method to be in the SUMMA class we require that $x^k \in C$ for each k and that each auxiliary function $g_k(x)$ satisfy the inequality

$$0 \leq g_k(x) \leq G_{k-1}(x) - G_{k-1}(x^{k-1}), \tag{3.1}$$

for all x. Note that it follows that $g_k(x^{k-1}) = 0$, for all k. For this note we require that $f(x)$ be convex and differentiable, and that the gradient operator, ∇f, be L-Lipschitz.

We assume, throughout this section, that the inequality in (3.1) holds for each k. We also assume that $\inf_{x \in C} f(x) = b > -\infty$. The next two results are taken from [5].

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

Proof: We have

$$f(x^{k+1}) + g_{k+1}(x^{k+1}) = G_{k+1}(x^{k+1}) \leq G_{k+1}(x^k) = f(x^k). \tag{3.2}$$

∎

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

Proof: Suppose that there is $\delta > 0$ such that $f(x^k) \geq b + 2\delta$, for all k. Then there is $z \in C$ such that $f(x^k) \geq f(z) + \delta$, for all k. From the inequality in (3.1) we have

$$g_k(z) - g_{k+1}(z) \geq f(x^k) + g_k(x^k) - f(z) \geq f(x^k) - f(z) \geq \delta, \tag{3.3}$$

for all k. But this cannot happen; the successive differences of a non-increasing sequence of non-negative terms must converge to zero. ∎

4 Using Sequential Unconstrained Optimization

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

$$G_k(x) = f(x) + \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_f(x, x^{k-1}), \tag{4.1}$$

where

$$D_f(x, x^{k-1}) = f(x) - f(x^{k-1}) - \langle \nabla f(x^{k-1}), x - x^{k-1} \rangle. \tag{4.2}$$

Since $f(x)$ is convex, $D_f(x, y) \geq 0$ for all x and y and is the Bregman distance formed from the function f [2].

Lemma 4.1 *The c^k that minimizes $G_k(x)$ over $x \in C$ is given by Equation (1.1).*

Proof: We know that

$$\langle \nabla G_k(c^k), c - c^k \rangle \geq 0,$$

for all $c \in C$. With

$$\nabla G_k(c^k) = \frac{1}{\gamma}(c^k - c^{k-1}) + \nabla f(c^{k-1}),$$

we have

$$\langle c^k - (c^{k-1} - \gamma \nabla f(c^{k-1})), c - c^k \rangle \geq 0,$$

for all $c \in C$. We then conclude that

$$c^k = P_C(c^{k-1} - \gamma \nabla f(c^{k-1})).$$

∎

The auxiliary function

$$g_k(x) = \frac{1}{2\gamma}\|x - x^{k-1}\|_2^2 - D_f(x, x^{k-1}) \tag{4.3}$$

can be rewritten as

$$g_k(x) = D_h(x, x^{k-1}), \tag{4.4}$$

where

$$h(x) = \frac{1}{2\gamma}\|x\|_2^2 - f(x). \tag{4.5}$$

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

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

$$\langle \nabla h(x) - \nabla h(y), x - y \rangle \geq 0, \tag{4.6}$$

for all x and y. This is equivalent to

$$\frac{1}{\gamma}\|x - y\|_2^2 - \langle \nabla f(x) - \nabla f(y), x - y \rangle \geq 0. \tag{4.7}$$

Since ∇f is L-Lipschitz, the inequality (4.7) holds whenever $0 < \gamma < \frac{1}{L}$.

A relatively simple calculation shows that

$$G_k(c) - G_k(c^k) = \frac{1}{2\gamma}\|c - c^k\|_2^2 + \frac{1}{\gamma}\langle c^k - (c^{k-1} - \gamma \nabla f(c^{k-1})), c - c^k \rangle. \tag{4.8}$$

From Equation (1.1) it follows that

$$G_k(c) - G_k(c^k) \geq \frac{1}{2\gamma}\|c - c^k\|_2^2, \tag{4.9}$$

for all $c \in C$, so that

$$G_k(c) - G_k(c^k) \geq \frac{1}{2\gamma}\|c - c^k\|_2^2 - D_f(c, c^k) = g_{k+1}(c). \tag{4.10}$$

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

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

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

so that

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

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

From

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

it follows that the sequence $\{c^k\}$ is bounded and that a subsequence converges to some $c^* \in C$ with $f(c^*) = f(\hat{c})$.

Replacing the generic $\hat{c}$ with c^*, we find that $\{G_k(c^*) - G_k(c^k)\}$ is decreasing. By Equation (4.8), it therefore converges to the limit

$$\frac{1}{2\gamma}\|c^* - P_C(c^* - \gamma\nabla f(c^*))\|_2^2 + \frac{1}{\gamma}\langle (P_C - I)(c^* - \gamma\nabla f(c^*)), c^* - P_C(c^* - \gamma\nabla f(c^*))\rangle = 0.$$

From the inequality in (4.9), we conclude that the sequence $\{\|c^* - c^k\|_2^2\}$ converges to zero, and so $\{c^k\}$ converges to c^*. This completes the proof of the theorem.

5 Some Examples

We present two examples to illustrate the application of the main theorem.

5.1 The Projected Landweber Algorithm

The problem is to minimize the function

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

over $x \in C$. The gradient of the function $f(x)$,

$$\nabla f(x) = A^T(Ax - b),$$

is L-Lipschitz for $L = \rho(A^T A)$, the largest eigenvalue of the matrix $A^T A$. The iterative step of the projected Landweber algorithm is

$$c^k = P_C(c^{k-1} - \gamma A^T(Ac^{k-1} - b)), \tag{5.1}$$

for $0 < \gamma < \frac{1}{L}$. According to the theorem, the sequence $\{c^k\}$ converges to a minimizer of $f(x)$ over x in C, whenever such minimizers exist. A minimizer need not exist, in general, though: let C be the closed, convex set in R^2 defined by

$$C = \{x = (x_1, x_2)^T \,|x_1 > 0,\, x_2 \geq \frac{1}{x_1}\},$$

$A = [0\ 1]$, and $b = 0$, so we want to minimize x_2, over x in C.

5.2 The CQ Algorithm

Let C and Q be non-empty, closed, convex subsets of R^J and R^I, respectively, and A a real I by J matrix. The *split feasibility problem* (SFP) is to find $c \in C$ with $Ac \in Q$. If the SFP has no exact solution, we try to minimize the function

$$f(x) = \frac{1}{2}\|P_Q Ax - Ax\|_2^2, \tag{5.2}$$

over $x \in C$. The gradient of the function $f(x)$ is

$$\nabla f(x) = A^T(I - P_Q)Ax, \tag{5.3}$$

and is L-Lipschitz for $L = \rho(A^T A)$.

The iterative step of the CQ algorithm is

$$c^k = P_C(c^{k-1} - \gamma A^T(I - P_Q)Ac^{k-1}), \tag{5.4}$$

for $0 < \gamma < \frac{1}{L}$ [3, 4]. According to the theorem, the sequence $\{c^k\}$ converges to a minimizer of $f(x)$ over $x \in C$, whenever such minimizers exist. If we select $Q = \{b\}$, then the CQ algorithm reduces to the projected Landweber algorithm.

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.

6.1 The Projected Gradient Algorithm

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

$$P_C z = P_{NS(A)} z + A^T (AA^T)^{-1} b, \tag{6.1}$$

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

$$P_{NS(A)} z = z - A^T (AA^T)^{-1} Az, \tag{6.2}$$

we have

$$P_C z = z + A^T (AA^T)^{-1} (b - Az). \tag{6.3}$$

For the *projected gradient algorithm* the iteration in Equation (1.1) becomes

$$c^k = c^{k-1} - \gamma P_{NS(A)} \nabla f(c^{k-1}), \tag{6.4}$$

which converges to a solution for any γ in $(0, \frac{1}{L})$, whenever solutions exist.

In the next subsection we present a somewhat simpler approach.

6.2 The Reduced Gradient Algorithm

Let c^0 be a *feasible point*, that is, $Ac^0 = b$. Then $c = c^0 + p$ 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

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

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* [9]. The gradient of $\phi(v)$, also called the *reduced gradient*, is

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

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

Let c^0 be feasible. The iteration in Equation (1.1) now becomes

$$v^k = v^{k-1} - \gamma \nabla \phi(v^{k-1}), \tag{6.5}$$

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

$$c^k = c^{k-1} - \gamma Z Z^T \nabla f(c^{k-1}). \tag{6.6}$$

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

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* [9]. 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

$$c^k = c^{k-1} - Z\Big(Z^T\nabla^2 f(c^{k-1})Z\Big)^{-1}Z^T\nabla f(c^{k-1}). \tag{6.7}$$

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

References

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

[2] Bregman, L.M. (1967) "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming." *USSR Computational Mathematics and Mathematical Physics* **7**, pp. 200–217.

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

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

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

[6] Csiszár, I. and Tusnády, G. (1984) "Information geometry and alternating minimization procedures." *Statistics and Decisions* **Supp. 1**, pp. 205–237.

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

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

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