A GRADIENT PROJECTION ALGORITHM FOR RELAXATION METHODS

By

J.L. Mohammed, R.A. Hummel,

and S.W. Zucker

\- - - - - - - -

Technical Report No. 66

March, 1983

New York University
Computer Science Dept.
Courant Institute for
Mathematical Sciences
251 Mercer Street
New York, N.Y. 10012

A GRADIENT PROJECTION ALGORITHM FOR RELAXATION METHODS

John L. Mohammed*

Robert A. Hummel**

Steven W. Zucker***

*Laboratory for Artificial Intelligence Research
Fairchild Advanced Research and Development
4001 Miranda Avenue , MS30-888
Palo Alto, California 94304

**Courant Institute of Mathematical Sciences
New York University
251 Mercer Street
New York, N. Y. 10012

***Computer Vision and Robotics Lab
Department of Electrical Engineering
McGill University
Montreal, Quebec, Canada H3A 2A7

Abstract: We consider a particular problem which arises when applying the method of gradient projection for solving constrained optimization and finite dimensional variational inequalities on the convex set formed by the convex hull of the standard basis unit vectors. The method is especially important for relaxation labeling techniques applied to problems in artificial intelligence. Zoutendijk's method for finding feasible directions, which is relatively complicated in general situations, yields a very simple finite algorithm for this problem. We present an extremely simple algorithm for performing the gradient projection and an independent verification of its correctness.

Section 1. Formulation of the Projection Problem.

We treat the following optimization problem:

Let $\mathbb{K}$ be the convex set defined by

$$\mathbb{K} = \{\vec{x} \in \mathbb{R}^n \mid \sum_{i=1}^{n} x_i = 1 \ , \ x_i \geq 0, \forall_i\}.$$

For any vector $\vec{x} \in \mathbb{K}$, the tangent set $T_{\vec{x}}$ is given by

$$T_{\vec{x}} = \{\vec{v} \in \mathbb{R}^n \mid \sum_{i=1}^{n} v_i = 0 \ , \ v_i \geq 0 \text{ whenever } x_i = 0\}.$$

The set of feasible directions at $\vec{x}$ is defined by

$$F_{\vec{x}} = T_{\vec{x}} \cap \{\vec{v} \in \mathbb{R}^n \mid \ \|\vec{v}\| \leq 1\},$$

where $\|\cdot\|$ is the standard Euclidean norm.

Given a "current point" $\vec{x} \in \mathbb{K}$, and an arbitrary direction $\vec{q} \in \mathbb{R}^n$, we consider

Problem P: Find $\vec{u} \in F_{\vec{x}}$ such that

$$\vec{q}\cdot\vec{u} \geq \vec{q}\cdot\vec{v} \quad \text{for all } \vec{v} \in F_{\vec{x}}.$$

Clearly, Problem P is a linear optimization problem with quadratic and linear constraints.

Problem P arises in the context of labeling problems in artificial intelligence, where iterative techniques similar to gradient ascent in $\mathbb{K}$ have been studied for their use in reducing ambiguity and achieving consistency [1,2]. The convex set $\mathbb{K}$ is especially appropriate for labeling problems. The set $\mathbb{K}$ can be viewed as the convex hull of the standard unit vectors $\vec{e}_i = (0,0,\ldots,1,0,\ldots,0)$, $i = 1,2,\ldots,n$. The vector $\vec{e}_i$ is assigned to an object to denote the labeling of that object with label number i. If the identity of the object is ambiguous, and no label can be assigned with complete certainty, a compromise vector $\vec{p} \in \mathbb{K}$ can be assigned to the object, so that

$$\vec{p} = (a_i,\ldots,a_n) = \sum a_i\vec{e}_i$$

denotes the labeling of that object with label numbers 1 through n, with degree of certainty a_1 through a_n respectively.

The optimization problem P arises in a solution method proposed for solving a variational inequality on $\mathbb{K}$ [3]. It also arises if one is solving a nonlinear optimization problem on $\mathbb{K}$ by the method of gradient ascent (steepest ascent). To motivate problem P, we present a brief discussion of the latter case.

Consider the problem of maximizing $F(\vec{x})$ among all $\vec{x} \in \mathbb{K}$, where F is a real-valued differentiable nonlinear function. If one uses the method of gradent ascent, then the procedure is to update successive values of $\vec{x}$ by replacing $\vec{x}$ with the vector $\vec{x} + \alpha\vec{u}$, where α is a small positive scalar, and $\vec{u}$ is chosen so as to maximize the directional derivative at $\vec{x}$. Of course, $\vec{u}$ is constrained by the requirement that it must lie tangent to the space $\mathbb{K}$ at $\vec{x}$ and $\vec{x} + \alpha\vec{u}$ is a numerical way of moving $\vec{x}$ infinitesimally in the direction $\vec{u}$. Further, since the directional derivative of F at $\vec{x}$, $D_{\vec{u}}F(\vec{x})$, is scaled by the magnitude of $\vec{u}$, it suffices to consider directions defined by vectors of unit length or less. Since $D_{\vec{u}}F(\vec{x}) = \text{grad } F(\vec{x})\cdot\vec{u}$, $\vec{u}$ can be found by maximizing $\vec{q}\cdot\vec{u}$ among all vectors $\vec{u} \in F_{\vec{x}}$, where $\vec{q} = \text{grad } F(\vec{x})$. Thus $\vec{u}$ solves problem P.

Section 2. Solution methodology

Problem P is simply the problem of projecting the given vector $\vec{q}$ onto the convex set formed by the tangent set $T_{\vec{x}}$, and then normalizing the length of the result. If $\vec{x}$ is an interior point in $\mathbb{K}$, (i.e., no component x_i is zero), then $T_{\vec{x}}$ is a subspace, and the solution $\vec{u}$ is simply the length-normalized orthogonal projection of $\vec{q}$ onto the subspace. This is accomplished by the trivial formulas

$$\vec{v} = \vec{q} - \frac{1}{n}\left(\sum_{i=1}^{n} q_i\right)\cdot(1,1,\ldots,1)$$

$$\vec{u} = \vec{v}/\|\vec{v}\| .$$

Thus, Problem P is interesting only when $\vec{x}$ lies on a face or edge of $\mathbb{K}$. Topologically, $\mathbb{K}$ is a simplex of degree n - 1, and has boundary surfaces of all lower degrees. However, if $\vec{x}$ lies on one of these surfaces, the set $T_{\vec{x}}$ is a convex set (shaped like a "wedge"), and the solution to Problem P is more complicated than projecting onto the boundary surface. For example, if $\vec{q}$ lies in $T_{\vec{x}}$, then the projection is simply the identity. On the other hand, if $\vec{q}$ lies in a direction that points away from all tangent directions in the wedge $T_{\vec{x}}$, that is, if $\vec{q}\cdot\vec{v} \leq 0$ for all $\vec{v} \in T_{\vec{x}}$, then the solution $\vec{u}$ is the zero vector. In between, there will be regions in which $\vec{q}$ projects to boundary surfaces of each order greater than or equal to the order of the boundary on which $\vec{x}$ lies.

Note that a boundary face (or edge of any given order) is itself a simplex. However, the solution vector $\vec{u}$ is not necessarily a simple projection of $\vec{q}$ onto this boundary simplex, as noted above.

A solution method exists, and can be obtained by applying algorithms from the theory of feasible directions to the specific geometry defined by $\mathbb{K}$. In particular, Zoutendijk offers finite algorithms [4] for solving problems of the form

$$\text{Maximize } \vec{q}\cdot\vec{u}, \quad \text{given } \vec{q},$$

subject to

$$A\vec{u} \leq \vec{b}$$

$$\vec{u}^T B\vec{u} \leq 1 \quad .$$

Problem P can be formulated in this way, with B = I, the n by n identity matrix. When Zoutendijk's algorithm is applied to Problem P, certain simplifications can be applied because B = I and because $\mathbb{K}$ is especially easy to define (i.e., the matrix A has a very simple form). These simplifications are equivalent to an extremely simple finite algorithm for solving Problem P, which we present in Section 3.

Despite the fact that Zoutendijk's algorithm has been available for over twenty years, and in spite of the algorithm's simplicity when applied to Problem P, alternative schemes are commonly used for projecting direction vectors $\vec{q}$ onto the set of feasible directions from a point on the probability space. These schemes typically do not solve Problem P, but rather yield feasible directions $\vec{u}$ which are "more or less" in the same direction as $\vec{q}$. For example, the "nonlinear probabilistic model" used in many applications of relaxation techniques to labeling problems defines

$$\vec{v} = \vec{p}\,' - \vec{p} \quad ,$$

where $$p'_i = \frac{p_i(1+\alpha q_i)}{\sum_j p_j(1+\alpha q_j)} \quad ,$$

and where $\alpha > 0$ is a small fixed constant.

It is easily verified that $\vec{v} \in T_{\vec{p}}$ as long as α is sufficiently small. Note, however, that if $p_i = 0$, then $v_i = 0$. That is, an iterative scheme based on these formulas can never leave a face or edge of the space IK.

Other projection schemes have been studied in connection with relaxation labeling. We mention the Product Rule [5], Bayesian analysis [6], and single component desaturization [7].

There is also an obvious "truncation method" of setting negative updating components to zero when the corresponding x_i's are zero. Note of these schemes yields a solution to Problem P. The connections between relaxation labeling, projection onto a convex set, and Problem P are not fully addressed in the cited references. However, in an accompanying paper by Hummel and Zucker, an algorithm for relaxation labeling is presented which requires, as a subroutine, an algorithm to solve Problem P.

Why should a problem which seems to be geometrically simple lead to so many different partial solution methods? Part of the answer is related to the formulation of the problems: the need for the projection operator is not always recognized. More importantly, the geometry is not quite as trivial as it first seems. Computing the regions of $\mathbb{R}^n$ in which $\vec{q}$ is projected to different order boundary surfaces requires a lot of care. The algorithm presented in the next section performs this computation.

Section 3. The projection Algorithm

The following algorithm solves Problem P.

0. Accept as input $\vec{x} \in \mathbb{K}$ and $\vec{q} \in \mathbb{R}^n$.

1. Set $k = 1$, $S_k = \emptyset$, $D = \{i | x_i = 0 \}$.

2. LOOP:

$$t_k := \frac{1}{n - \#S_k} \sum_{i \notin S_k} q_i$$

$$S_{k+1} := \{i \in D \mid q_i < t_k\}$$

If $S_{k+1} = S_k$, then EXIT LOOP

$k := k + 1$

END LOOP

3. Compute $\vec{y}$, where

$$y_i = \begin{cases} 0 & \text{if } i \in S_k \\ q_i - t_k & \text{if } i \notin S_k \end{cases}$$

4. Output $\vec{u}$, where

$$\vec{u} = \begin{cases} 0 & \text{if } \vec{y} = 0 \\ \vec{y}/\|\vec{y}\| & \text{otherwise} \end{cases}$$

One way to verify that the resulting vector is the correct solution is to observe that $\vec{u}$ satisfies the Kuhn-Tucker conditions (at least when $\vec{u} \neq 0$), which is equivalent to solving the constrained optimization problem [8]. In this proof, the Lagrange multiplier λ belonging to the constraint $\Sigma u_i = 0$ has the same value as the final threshold t_k. We now present an alternate proof. This proof is self-contained, and handles the $\vec{u} = 0$ and $\vec{u} \neq 0$ cases uniformly.

Before showing that $\vec{u}$ is the desired solution, however, we show that the algorithm terminates in a finite number of steps.

Proposition: The S_k's are nested, and thus the algorithm terminates with at most #D + 1 passes through the loop.

<u>Proof</u>. Since $S_1 = \emptyset$, $S_1 \subseteq S_2$. We assume by induction that $S_{k-1} \subseteq S_k$. Since $i \in S_k$ implies that $q_i < t_{k-1}$,

$$(n - \#S_k)t_k = \sum_{i \notin S_k} q_i = \sum_{i \notin S_{k-1}} q_i - \sum_{i \in S_k \setminus S_{k-1}} q_i$$

$$(n - \#S_{k-1})t_{k-1} - (\#S_k - \#S_{k-1})t_{k-1} = (n - \#S_k)t_{k-1} ,$$

so $t_k \geq t_{k-1}$. Then clearly $S_k \subseteq S_{k+1}$, by the definition of the S_k's. The proposition follows since the loop terminates when $S_k = S_{k-1}$.

<u>Section 4</u>. Proof of the Algorithm

Let $k = N$ denote the maximum value of k attained during the last iteration through the loop (step 2). Denote by W the space

$$W = \{\vec{v} \in T_{\vec{x}} \mid v_i = 0 \quad \text{for} \quad i \in S_N\}.$$

Note that if $\vec{v} \in W$ and $\|\vec{v}\| \leq 1$, then $\vec{v} \in F_{\vec{x}}$.

<u>Lemma</u>. The maximum of $\vec{q} \cdot \vec{v}$, $\vec{v} \in F_{\vec{x}}$ is attained among $\vec{v} \in W \cap F_{\vec{x}}$.

<u>Proof</u>: Suppose that $\vec{v} \in F_{\vec{x}}$ maximizes $\vec{q} \cdot \vec{v}$, and that $v_i > 0$ for some $i \in S_N$. Define $\vec{w}$ by

$$w_j = \begin{cases} 0 & \text{if} \quad j = i \\ v_j & \text{if} \quad j \in S_N \setminus \{i\} \\ v_j + \dfrac{1}{n - \#S_N} v_i & \text{if} \quad j \notin S_N . \end{cases}$$

Since $\vec{v} \in F_{\vec{x}}$, $v_j \geq 0$ for $j \in D$, and thus $w_j \geq 0$ for $j \in D$. Further, $\Sigma w_j = \Sigma v_j = 0$. Finally,

$$\|\vec{w}\|^2 = \Sigma w_j^2 = \sum_{\substack{j \in S_N \\ j \neq i}} v_j^2 + \sum_{j \notin S_N} (v_j + \frac{1}{n - \#S_N} v_i)^2$$

$$= (\sum_{j \neq i} v_j^2) + \frac{1}{n - \#S_N} v_i^2 + \frac{2v_i}{n - \#S_N} \sum_{j \notin S_N} v_j$$

$$\leq \|\vec{v}\|^2 - \frac{2v_i}{n - \#S_N} \sum_{j \in S_N} v_j \leq \|\vec{v}\|^2 \leq 1 .$$

Here we used $v_i > 0$, and $j \in S_N$ implies $v_j \geq 0$. Combining, we have shown that $\vec{w} \in F_{\vec{x}}$. But

$$\vec{q} \cdot \vec{w} = \sum_{\substack{j \in S_N \\ j \neq i}} q_j v_j + \sum_{j \notin S_N} q_j (v_j + \frac{1}{n - \#S_N} v_i)$$

$$= \vec{q} \cdot \vec{v} - q_i v_i + t_N v_i$$

$$= \vec{q} \cdot \vec{v} + (t_N - q_i) v_i > \vec{q} \cdot \vec{v} ,$$

since $i \in S_N$ implies $q_i < t_N$, and since $v_i > 0$. This is a contradiction, since $\vec{q} \cdot \vec{v}$ is maximum among $\vec{v} \in F_{\vec{x}}$. Thus we must have $v_i = 0$ for $i \in S_N$. That is, $v \in W$.

<u>Theorem</u>: The output vector $\vec{u}$ solves Problem P.

<u>Proof</u>: According to the lemma, it suffices to show that $\vec{u} \in F_{\vec{x}}$, and $\vec{q} \cdot \vec{u} \geq \vec{q} \cdot \vec{v}$ for all $\vec{v} \in W \cap F_{\vec{x}}$.

Clearly, $\vec{y}$ calculated in step 3 of the algorithm satisfies $y_i = 0$ for $i \varepsilon S_N$. Further, if $i \varepsilon D$, $i \notin S_N$, then $q_i \geq t_N$ according to the definition of S_N, and so $y_i = q_i - t_N \geq 0$. So $y_i \geq 0$ for all $i \varepsilon D$. Finally $\Sigma y_i = 0$ by direct computation. Thus $\vec{y} \varepsilon W$.

In fact, as can be easily seen, step 3 merely performs an orthogonal projection of $\vec{q}$ onto the subspace W. That is, for any $\vec{v} \varepsilon W$,

$$(\vec{q} - \vec{y}) \cdot \vec{v} = 0.$$

So $\vec{q} \cdot \vec{v} = \vec{y} \cdot \vec{v}$ for all $\vec{v} \varepsilon W$.

The output vector $\vec{u}$ calculated in step 4 is simply a length normalization of $\vec{y}$, and so $\vec{u}$ is in W also. Since $\|\vec{u}\| \leq 1$, $\vec{u} \varepsilon F_{\vec{x}}$.

Next, observe that since $\vec{u} \varepsilon W$,

$$\vec{q} \cdot \vec{u} = \vec{y} \cdot \vec{u} = \|\vec{y}\| .$$

The last equality follows form the definition of $\vec{u}$ in step 4. Let $\vec{v}$ be any vector in $W \cap F_{\vec{x}}$. Then $\|\vec{v}\| \leq 1$, and so

$$\vec{q} \cdot \vec{v} = \vec{y} \cdot \vec{v} \leq \|\vec{y}\| \cdot \|\vec{v}\| \leq \|\vec{y}\| ,$$

using the Cauchy-Schwartz inequality. We have therefore shown that

$$\vec{u} \varepsilon F_{\vec{x}} : \vec{q} \cdot \vec{u} \geq \vec{q} \cdot \vec{v} \text{ for all } \vec{v} \varepsilon W \cap F_{\vec{x}} .$$

This proves the theorem.

References

1. Zucker, S.W., "Relaxation labeling and the reduction of local ambiguity", Proc. 3rd Intern, Joint Conf. on Pattern Recognition, San Diego, 1976; also in C.H. Chen (ed.), Pattern Recognition and Artificial Intelligence, Academic Press, 1977.

2. Rosenfeld, A., R.A. Hummel, and S.W. Zucker, "Scene labeling by Relaxation operations", IEEE Trans. on Systems, Man, and Cybernetics, 1976, SMC-6, p.420.

3. Hummel, R.A., and S.W. Zucker, "On the Foundations of Relaxation Labeling Processes", TR-80-7, Computer Vision Lab. McGill University, July, 1980.

4. Zoutendijk, G., Methods of Feasible Directions, Elsevier Publ., 1960; also Mathematical Programming Methods, North-Holland, 1976.

5. Zucker, S.W. and J. Mohammed, "An analysis of probabilistic relaxation labeling processes", in Proceedings of Pattern Recognition and Image Processing, Chicago, May 1978.

6. Peleg, S., "A new probabilistic relaxation scheme", IEEE Trans. Pattern Analysis and Machine Intelligence 2, 1980, p. p. 362.

7. Berthod, M., and O. Faugeras, "Using context in the global recognition of a set of objects, Appendix," Proc. IFIP, 1980.

8. Luenberger, D., Introduction to Linear and Nonlinear Programming, Addison-Wesley, 1973.