Appl Math Optim 24:273–288 (1991)

Applied Mathematics and Optimization

A Least-Squares-Based Method for a Class of Nonsmooth Minimization Problems with Applications in Plasticity*

Aharon Ben-Tal,[1] Marc Teboulle,[2] and Wei H. Yang[3]

[1] Faculty of Industrial Engineering and Management,
Technion, Haifa 32000, Israel

[2] Department of Mathematics and Statistics, University of Maryland,
Baltimore County Campus, Baltimore, MD 21228, USA

[3] Department of Mechanical Engineering and Applied Mechanics,
University of Michigan, Ann Arbor, MI 48109, USA

Abstract. This paper introduces a globally convergent algorithm for solving a class of nonsmooth optimization problems, involving square roots of quadratic forms. The class includes in particular limit analysis problems in plasticity. The algorithm combines smoothing with successive approximation. The main computational effort in each iteration is solving a linear weighted least-squares problem. The convergence of the algorithm is proved and an *a priori* error estimate is obtained. Numerical results are presented for two limit analysis problems.

1. Introduction

This paper is motivated by certain problems in mechanics, in particular *Limit Analysis* of beams and plates, and by an algorithm of Yang [12], which solves them by using the (discretized) dual problems and employs a smoothing technique combined with successive approximations steps to obtain efficiently good approximate solutions.

* The work of the first author was partially supported by NSF Grant DDM-89-96112. Parts of the work was done during his stay at the University of Bayreuth as a guest of the DFG. The work of the second author was supported in part by the Air Force Office of Scientific Research under contract AFOSR-88-0218 and by a National Science Foundation Grant ECS-8802239 at the University of Maryland, Baltimore County Campus.

Here, we embed the above limit analysis problems in a wider class of nonsmooth functionals (involving sum of square roots), which contain other problems in mechanics, and also some location problems (e.g., the Fermat–Weber problem), see Section 2. We provide an algorithm for minimizing functionals in the above class, which is based on a combination of smoothing and successive approximation (CSSA algorithm). For the limit analysis problems considered in this paper, the CSSA algorithm is a modification of Yang's algorithm. Although the two algorithms behave numerically quite the same, it is only the CSSA algorithm which lends itself to a rigorous convergence analysis. For the Fermat–Weber problem, the CSSA algorithm coincides with an algorithm by Morris [9] which is a modification of the well-known Weisfeld algorithm [7], [11]. More recent algorithms for location problems can be found in, e.g., [3] and [8].

Our main contribution is providing a comprehensive convergence analysis for the CSSA algorithm. Moreover, we show that the main computational effort of the CSSA algorithm is to solve a *linear weighted least-squares* problem, see Section 3. The convergence analysis developed in Section 4 also furnishes an interesting explanation to a phenomenon already observed by Yang, namely the excellent performance of the algorithm in the first step, regardless of the starting point and the size of the smoothing parameter. In Section 5 we derive a dual problem to the smoothed primal and show how to use it to obtain a stopping criterion for the algorithm and an estimate on the suboptimality of the solution reached. In Section 6 we report on computational results in using the CSSA algorithm for solving two limit analysis problems: a small-scale three bar truss and a large-scale (2500 variables) simply supported square plate.

2. A Class of Nonsmooth Problems

In the theory of plasticity, a prototype problem from limit analysis of beams [12] is to find the plastic collapse load λ^* which is the optimal solution of

$$\max\{\lambda\colon M''(x) = \lambda q_0(x), |M(x)| \le 1, a \le x \le b\}, \tag{2.1}$$

where $M(x)$ is the moment in the beam; $q_0(x)$ is a given load distribution and the static boundary condition on $M(x)$ at the ends a and b should also be included in (2.1) if they are prescribed. This prototype problem is an important special case of more general problems of mechanics given in two and three dimensions. They involve a partial differential equation for equilibrium and a more general norm inequality for material behavior. For example, the simply supported plate problem [12] is to find the maximum load parameter λ^* solving

$$\max\{\lambda\colon \nabla\cdot\nabla\cdot u = \lambda\varphi,\ F(u) \le u_0,\ \text{boundary condition on } u\} \tag{2.2}$$

where

$$u = \begin{pmatrix} u_{xx} & u_{xy} \\ u_{yx} & u_{yy} \end{pmatrix},$$

u_{xx}, u_{yy} are the bending moment components, u_{xy} is the twisting moment, u_0 is a material constant, $\varphi(x, y)$ is a given distribution function, and

$$F(u) := (u_{xx}^2 + 2u_{xy}^2 + u_{yy}^2)^{1/2}$$

is the Frobenius norm.

For the above-stated problem (2.2), namely the primal problem, we can derive a dual problem, see, e.g., [12]:

$$\min u_0 \iint_\Omega F(\nabla\nabla v)\, dx\, dy \tag{2.3}$$

subject to

$$\iint_\Omega v\varphi\, dx\, dy = 1, \qquad \text{boundary conditions on } v.$$

Note that the dual problem has a nonsmooth objective function due to the presence of the square root. Minimization of the square root of functionals arise in other applied mechanic problems. For example, the minimal surface or plateau problem [4] is

$$\min \iint_\Omega \sqrt{1 + u_x^2 + u_y^2}\, dx\, dy \tag{2.4}$$

subject to

$$u = g \qquad \text{a given function on } \partial\Omega.$$

The discretization of the above problems (in the dual formulation) can be embedded in the following class of convex nonsmooth optimization problems:

$$\text{(P)} \quad \min\left\{ f(x) := \sum_{i=1}^{N} h_i(Q_i x - r_i) \colon x \in \mathbb{R}^n \right\},$$

where $Q_i \in \mathbb{R}^{p \times n}$, $r_i \in \mathbb{R}^p$, $h_i\colon \mathbb{R}^p \to \mathbb{R}$ is the convex function (weighted norm) $h_i(x) = (z^T D_i z)^{1/2} = \|z\|_{D_i}$, and $D_i \in \mathbb{R}^{p \times p}$ is a diagonal positive definite matrix.

Example 1: Beam. Applying a finite-element approximation scheme, problem (2.1) is

$$\max\{\lambda\colon Bq = \lambda q_0,\ \|q\|_\infty \le 1\}, \tag{2.5}$$

where $B \in \mathbb{R}^{m \times n}$, $\|q\|_\infty = \max_{i=1,\ldots,n}|q_i|$. It is easily seen that the constraint of the problem implies

$$|\lambda| \le \sigma, \tag{2.6}$$

where

$$\sigma = \frac{\|B\|_\infty}{\|q_0\|_\infty}.$$

With the constraint (2.6) added to problem (2.5) we can derive the following dual problem of (2.5):

$$\min\{\|B^t u\|_1 + \sigma|q_0^t u - 1|\}, \tag{2.7}$$

which is an *unconstrained linear l_1-norm minimization problem.* Problem (2.7) is a special case of (P) with $p = 1$, $N = m + 1$, and $Q_i =$ ith row of B, $r_i = 0$, $D_i = 1$, $h_i(z) = \sqrt{z^2} = |z|$ for $i = 1, \ldots, N - 1$, and $Q_N = q_0^T$, $r_N = 1$, $D_N = \sigma^2$, $h_N(z) = \sigma|z|$. We note that the l_1-norm problem (2.7) also models the limit analysis problem of trusses, see Section 6, Example 6.1.

Example 2: Simply Supported Plate. In the plate problem (2.2), a bound of the type $|\lambda| \leq \sigma$ can be added, arguing as in the previous example. In this case an equivalent unconstrained problem of (2.3) is

$$\min_v \left\{ u_0 \iint_\Omega F(\nabla\nabla u)\, dx\, dy + \sigma \left| 1 - \iint_\Omega \varphi v\, dx\, dy \right| \right\}. \tag{2.8}$$

The discretized form of problem (2.8) is

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{N-1} \sqrt{(Q_i x - r_i)^T D_i (Q_i x - r_i)} + \sigma|1 - q_0^T x| \tag{2.9}$$

with appropriate matrices $Q_i \in \mathbb{R}^{3 \times n}$, $D_i \in \mathbb{R}^{3 \times 3}$ (positive diagonal matrix), and vectors $r_i \in \mathbb{R}^3$, $q_0 \in \mathbb{R}^n$. This problem is in the format of our prototype problem (P).

Example 3: Plateau Problem. A discretization of problem (2.4) leads to solving

$$\min\left\{ F(x) := \sum_{i=1}^{N} \sqrt{\tfrac{1}{2} x^T P_i x + d_i^T x + c_i}\colon x \in \mathbb{R}^n \right\}, \tag{2.10}$$

where for every i the matrix $P_i \in \mathbb{R}^{n \times n}$ is positive definite and the scalar c_i satisfies $c_i > \frac{1}{2} d_i^T P_i^{-1} d_i$ (a necessary and sufficient condition for the convexity of F). Then with $p = n + 1$,

$$Q_i = \begin{bmatrix} (P_i)^{1/2} \\ 0 \end{bmatrix} \in \mathbb{R}^{p \times n},$$

where $P_i^{1/2}$ denotes the square root of P_i,

$$r_i = \begin{bmatrix} -P_i^{-1/2} d_i \\ -\sqrt{c_i - 1/2 d_i P_i^{-1} d_i} \end{bmatrix} \in \mathbb{R}^p,$$

$D_i = I$, and $h_i(z) = \|z\|$, problem (2.10) is in the form (P).

Example 4: Fermat–Weber Problem. With the choice $p = n$, $Q_i = I$, and $h_i(z) = \lessgtr_i \|z\|$, where w_i are given weights, problem (P) reduces to the well-known location theory problem (see, e.g., [7]):

$$\min\left\{ \sum_{i=1}^{N} w_i \|x - r_i\|\colon x \in \mathbb{R}^n \right\}.$$

3. The CSSA Algorithm

The convex objective function of problem (P)

$$f(x) = \sum_{i=1}^{N} \|Q_i x - r_i\|_{D_i} := \sum_{i=1}^{N} f_i(x)$$

is not continuously differentiable at the points x satisfying

$$Q_i x - r_i = 0$$

for some i, and so we use (as in, e.g., [5]) a perturbation of $f(x)$

$$f_\varepsilon(x) := \sum_{i=1}^{N} (f_i^2(x) + \varepsilon^2)^{1/2}, \tag{3.1}$$

where $\varepsilon > 0$ is a *smoothing* parameter. The function $f_\varepsilon(x)$ is a differentiable function in x for every $\varepsilon > 0$, and, for any fixed x, $f_\varepsilon(x) \to f(x)$ as $\varepsilon \to 0$, moreover $f_\varepsilon(x)$ preserves the convexity of f, which is a consequence of the following lemma.

Lemma 3.1. *The function $x \to f_\varepsilon(x)$ is convex.*

Proof. For each i, the function $g(t_i) := (t_i^2 + \varepsilon^2)^{1/2}$ is convex and increasing for all $t_i \geq 0$. Hence, $\sqrt{\|Q_i x - r_i\|_{D_i} + \varepsilon^2}$, which is a composition $g(\|Q_i x - r_i\|_{D_i})$, is convex by Theorem 5.1 of [10] and thus f_ε being the sum of such terms is convex. □

Since $f_\varepsilon(x)$ is differentiable and convex the necessary and sufficient optimality conditions for $x_\varepsilon^* \in \mathbb{R}^n$ to solve the perturbed (approximating) problem

$$(\mathrm{P}_\varepsilon) \quad \min\{f_\varepsilon(x) \colon x \in \mathbb{R}^n\}$$

are given by

$$\nabla f_\varepsilon(x_\varepsilon^*) = \sum_{i=1}^{N} \frac{Q_i^T D_i (Q_i x_\varepsilon^* - r_i)}{\{\|Q_i x_\varepsilon^* - r_i\|_{D_i} + \varepsilon^2\}^{1/2}} = 0. \tag{3.2}$$

For each $i = 1, \ldots, N$ define the positive numbers

$$\mu_i(x, \varepsilon) := \frac{1}{\{\|Q_i x - r_i\|_{D_i}^2 + \varepsilon^2\}^{1/2}}. \tag{3.3}$$

Equation (3.2) can now be written as

$$\sum_{i=1}^{N} \mu_i(x_\varepsilon^*, \varepsilon) Q_i^T D_i Q_i x_\varepsilon^* = \sum_{i=1}^{N} \mu_i(x_\varepsilon^*, \varepsilon) Q_i^T D_i r_i. \tag{3.4}$$

This system of equations reduces to the following compact form:

$$A^T M_\varepsilon(x_\varepsilon^*) A x_\varepsilon^* = A^T M_\varepsilon(x_\varepsilon^*) b, \tag{3.5}$$

where

$$A := \begin{bmatrix} (D_1^{1/2})^T Q_1 D_1^{1/2} \\ \vdots \\ (D_N^{1/2})^T Q_N D_N^{1/2} \end{bmatrix} \in \mathbb{R}^{pN \times n}, \qquad b := \begin{bmatrix} r_1 \\ \vdots \\ r_N \end{bmatrix} \in \mathbb{R}^{pN},$$

and $M_\varepsilon(x) := \operatorname{diag}(\mu_1(x, \varepsilon), \ldots, \mu_N(x, \varepsilon))$ is an $N \times N$ positive definite matrix. Let $m := pN$ and assume in the following that $m > n \geq 2$ and that rank $A = n$, then $A^T M_\varepsilon(\cdot) A$ is nonsingular and it follows from (3.4) that

$$x_\varepsilon^* = (A^T M_\varepsilon(x_\varepsilon^*) A)^{-1} A^T M_\varepsilon(x_\varepsilon^*) b. \tag{3.6}$$

Then (3.6) is a fixed-point equation $x_\varepsilon^* = T(x_\varepsilon^*)$ with the map T

$$T\colon x \to T(x) = (A^T M_\varepsilon(x) A)^{-1} A^T M_\varepsilon(x) b. \tag{3.7}$$

This naturally suggests using *successive approximations*

$$x_{k+1} = T(x_k), \qquad k = 0, 1, \ldots,$$

to compute x_ε^*.

For each k, the right-hand side of the above equation is in fact the solution of the *normal equations* (3.5) corresponding to the weighted least-squares (WLS) problem:

$$\min \| M_\varepsilon^{1/2}(x_k) A x - M_\varepsilon^{1/2}(x_k) b \|, \tag{3.8}$$

where $M_\varepsilon^{1/2}(x) := \operatorname{diag}(\sqrt{\mu_1(x, \varepsilon)}, \ldots, \sqrt{\mu_N(x, \varepsilon)})$.

Thus the basic algorithm for solving problem (P_ε) consists of the following steps:

The CSSA Algorithm

0. Choose $\varepsilon > 0$ and $x_0 \in \mathbb{R}^n$. Set $k = 0$.
1. Solve the WLS problem (3.8):

   $$x_{k+1} = \operatorname{argmin} \| M_\varepsilon^{1/2}(x_k) A x - M_\varepsilon^{1/2}(x_k) b \|.$$

2. If an appropriate stopping rule holds, then stop. Otherwise, set $k = k + 1$ and go to step 1.

The stopping rule given in step 2 is discussed in Section 5 using duality results to provide a bound on the suboptimality of the iterate x_k.

The performance of the above algorithm has the remarkable property of producing practically acceptable solutions even from bad starting points and for a wide range of ε values, see Section 6, for numerical examples. The insensitivity to bad starting points is a consequence of Corollary 4.1, where we prove that the first iterate must lie in a compact set (see Section 4). The next result demonstrates that if ε becomes large, then x_ε approaches the least-squares solution of $Ax = b$.

Lemma 3.2. *Let z denote the least-squares solution of the overdetermined linear system of equations $Ax = b$. Then* $\lim_{\varepsilon \to \infty} x_\varepsilon = z$.

Proof. For each $i = 1, \dots, N$ define

$$\bar{\mu}_i(x, \varepsilon) := \frac{1}{\{(\|Q_i x - r_i\|_{D_i}/\varepsilon)^2 + 1\}^{1/2}}.$$

Then $M_\varepsilon(x) = (1/\varepsilon)\bar{M}_\varepsilon(x)$ where $\bar{M}_\varepsilon(x) := \operatorname{diag}(\bar{\mu}_1(x, \varepsilon), \dots, \bar{\mu}_N(x, \varepsilon))$. The optimal solution x_ε^* satisfied (3.3) which can be rewritten as

$$A^T \bar{M}_\varepsilon(x_\varepsilon^*) A x_\varepsilon^* = A^T \bar{M}_\varepsilon(x_\varepsilon^*) b. \tag{3.9}$$

As $\varepsilon \to \infty$ then $\bar{M}_\varepsilon(x_\varepsilon^*) \to I$, the identity matrix. Hence, taking limits in (3.9) produces the result. □

4. Convergence Analysis

In this section we prove the convergence of the CSSA algorithm. The first result demonstrates that the algorithm for problem (P_ε) possess a descent property.

Consider the perturbed function

$$f_\varepsilon(x) = \sum_{i=1}^{N} \{f_i(x)^2 + \varepsilon^2\}^{1/2} := \sum_{i=1}^{N} k_i(x).$$

Then it is easy to verify that step 1 in the algorithm is the same as

$$x_{k+1} = \operatorname{argmin} g_\varepsilon(x, x_k),$$

where

$$g_\varepsilon(x, x_k) := \sum_{i=1}^{N} \frac{f_i(x)^2 + \varepsilon^2}{\sqrt{f_i(x_k)^2 + \varepsilon^2}}.$$

Lemma 4.1. *If $T(x_k) \neq x_k$, then $f_\varepsilon(x_{k+1}) < f_\varepsilon(x_k)$.*

Proof. With the notations defined above we have

$$g_\varepsilon(x_{k+1}, x_k) = \sum_{i=1}^{N} k_i(x_k)^{-1} k_i(x_{k+1})^2. \tag{4.1}$$

Applying the inequality, see, e.g., [1],

$$a^{1/p} b^{1/q} \geq \frac{a}{p} + \frac{b}{q}, \qquad p < 1, \quad a, b > 0, \qquad \frac{1}{p} + \frac{1}{q} = 1,$$

to (4.1) with $a = k_i(x_k)$, $b = k_i(x_{k+1})$, and $p = -1$ it follows that

$$g_\varepsilon(x_{k+1}, x_k) \geq -\sum_i k_i(x_k) + 2 \sum_i k_i(x_{k+1}) = -f_\varepsilon(x_k) + 2 f_\varepsilon(x_{k+1}). \tag{4.2}$$

Now observe that $x \to g_\varepsilon(x, x_k)$ is strictly convex. Indeed, we can write

$g_\varepsilon(x, x_k) = \psi(Ax - b)$ by defining $\psi\colon \mathbb{R}^m \to \mathbb{R}$:

$$\psi(z) = \sum_{i=1}^{N} \mu_i(x_k, \varepsilon)\|z_i\|_{D_i}^2.$$

Since $\mu_i(x_k, \varepsilon) > 0$, $\forall i = 1, \ldots, N$, the function ψ is strictly convex. Let $x, y \in \mathbb{R}^n$, $x \neq y$. Since rank $A = n$, it follows that $Ax \neq Ay$ and therefore the strict convexity of ψ implies that g is strictly convex. Hence, x_{k+1} is the unique minimizer of $g(x, x_k)$. Since $x_{k+1} = T(x_k) \neq x_k$, if follows that

$$g_\varepsilon(x_{k+1}, x_k) < g_\varepsilon(x_k, x_k) = f_\varepsilon(x_k). \tag{4.3}$$

Combining inequalities (4.2) and (4.3) we have

$$2f_\varepsilon(x_{k+1}) - f_\varepsilon(x_k) < f_\varepsilon(x_k)$$

and the result $f_\varepsilon(x_{k+1}) < f_\varepsilon(x_k)$ is proved. □

The next result follows immediately from the definition of the mapping T.

Lemma 4.2. *Let x_ε^* be the minimizer of $f_\varepsilon(x)$. Then $x_k = x_\varepsilon^*$ if and only if $T(x_k) = x_k$.*

A crucial step in proving that the sequence $\{x_k\}$ generated by the map T (i.e., by solving the WLS problem (2.14)) converges is to demonstrate that it resides in some compact set. Such a result was obtained recently by the first two authors in [2] and is summarized below.

Let A be a real $m \times n$ matrix $(m > n)$ and let M be a given positive definite diagonal matrix. Consider the solution of the WLS problem:

$$A^T M A x = A^T M b. \tag{4.4}$$

Let $P_{m,n}$ be the collection of all index sets containing m distinct integers out of the set $\{1, \ldots, m\}$. Let y_{I_j} be the solution of the jth $n \times n$ subsystem corresponding to rows of A indexed by $I_j = \{i_1, \ldots, i_n\} \subset P_{m,n}$, i.e.,

$$A_{I_j} y_{I_j} = b_{I_j}.$$

Denote $P_{m,n}^+ = \{j \in P_{m,n}\colon \det A_{I_j} \neq 0\}$. We state the result in a form appropriate for our purposes.

Theorem 4.1 [2]. *Let x be the solution of the WLS problem given by* (4.4). *Then*

$$x \in \text{convex hull } \{y_{I_j}\colon j \in P_{m,n}^+\}. \tag{4.5}$$

Moreover, the y_{I_j}'s depend only on the data A, b and not on the matrix M.

Returning to the algorithm defined in Section 3 by the map T, we then have as an immediate consequence of Theorem 4.1:

Corollary 4.1. *The sequence $\{x_k\}_{k=0}^{k=\infty}$ generated by the map T lies in a compact set.*

From the last statement in Theorem 4.1 it follows in particular that the first iterate x_1 lies in a fixed compact set which depends *only* on the problem data (A, b), and which is independent on the *algorithmic information* x_0 and ε. This explains why the CSSA algorithm performs initially well, even from distant (from the optimal solution) starting points.

We are now in position to prove the convergence theorem, proceeding along the same lines as Kuhn [7] and Morris [9].

Theorem 4.2. *Given any* x_0, *define* $x_k = T^k(x_0)$ *for* $k = 1, 2, \ldots$. *Then* $\lim_{k\to\infty} x_k = x_\varepsilon^*$, *where* $x_\varepsilon^* := \operatorname{argmin}\{f_\varepsilon(x): x \in \mathbb{R}^n\}$.

Proof. By Corollary 4.1, the sequence $\{x_k\}$ lies in a compact set. Hence, by the Bolzano–Weierstrass theorem, there exists at least one point, say $\bar{x}$, and a convergent subsequence x_{k_l} such that $\lim_{l\to\infty} x_{k_l} = \bar{x}$. We prove below that $\bar{x} = x_\varepsilon^*$ in all cases.

Case 1. If $x_{k+1} = T(x_k) = x_k$ for some k, then the sequence repeats from that point and $\bar{x} = x_k$. But then $\bar{x} = x_\varepsilon^*$ by Lemma 4.2.

Case 2. If $x_{k+1} \neq T(x_k)$, then by Lemma 4.1

$$f_\varepsilon(x_{k+1}) < f_\varepsilon(x_k)$$

and thus

$$f_\varepsilon(x_0) > f_\varepsilon(x_1) > \cdots > f_\varepsilon(x_k) > \cdots > f_\varepsilon(x_\varepsilon^*).$$

Hence,

$$\lim_{k\to\infty} \{f_\varepsilon(x_{k_l}) - f_\varepsilon(T(x_{k_l})\} = 0.$$

Now from the definition of the map $T(x)$, all the diagonal elements $\mu_i(x, \varepsilon)$, $i = 1, \ldots, N$, of the matrix $M(x)$ are continuous functions of x, and thus T is a continuous map, which implies that

$$\lim_{l\to\infty} T(x_{k_l}) = T(\bar{x})$$

and hence we have

$$f_\varepsilon(\bar{x}) - f_\varepsilon(T(\bar{x})) = 0.$$

Therefore by Lemma 4.1, $\bar{x} = T(\bar{x})$ and hence $\bar{x} = x_\varepsilon^*$ by Lemma 4.2 and the theorem is proved. □

The rest of this section is devoted to error analysis; here we attempt to answer the following question:

If the sequence $\{x_k\}$ generated by the CSSA algorithm converged to x_ε^*, how far is it (in terms of the objective function value) from the true optimal solution x^*? This is answered in Theorem 4.3. First we demonstrate that $f_\varepsilon(x)$ converges uniformly to $f(x)$ as $\varepsilon \to 0$. The result is valid for arbitrary functions $f_i(x)$.

Lemma 4.3. *For all $x \in \mathbb{R}^n$, $|f_\varepsilon(x) - f(x)| \le N \cdot \varepsilon$.*

Proof. We use the following inequality:

$$(a+b)^\alpha \le a^\alpha + b^\alpha, \qquad \forall a, b \ge 0, \quad 0 < \alpha \le 1.$$

Pick $a = f_i(x)^2$, $b = \varepsilon^2$, $\alpha = \frac{1}{2}$. Then, for all $i = 1, \dots, N$,

$$\{f_i^2(x) + \varepsilon^2\}^{1/2} \le |f_i(x)| + \varepsilon.$$

Summing the N inequalities above we obtain

$$f_\varepsilon(x) - f(x) \le N \cdot \varepsilon$$

and the result follows. □

Theorem 4.3. *For every $\varepsilon > 0$,*

$$f(x_\varepsilon^*) - f(x^*) \le N \cdot \varepsilon.$$

Proof. Note that by the definition of $f_\varepsilon(x)$ we have

$$f_\varepsilon(x) > f(x), \qquad \forall x. \tag{4.6}$$

Hence,

$$f(x_\varepsilon^*) - f(x^*) < f_\varepsilon(x_\varepsilon^*) - f(x^*). \tag{4.7}$$

Since x_ε^* solves min $f_\varepsilon(x)$, then

$$f_\varepsilon(x_\varepsilon^*) \le f_\varepsilon(x), \qquad \forall x, \tag{4.8}$$

and thus in particular $f_\varepsilon(x_\varepsilon^*) \le f_\varepsilon(x^*)$. Combining with (4.7) it follows that

$$f(x_\varepsilon^*) - f(x^*) \le f_\varepsilon(x_\varepsilon^*) - f(x^*) \le f_\varepsilon(x^*) - f(x^*) \le N \cdot \varepsilon, \tag{4.9}$$

the last inequality following from Lemma 4.3. □

The inequality (4.9) gives an *a priori* bound on the suboptimality of the optimal solution x_ε^* produced by solving the approximate problem (P_ε).

5. A Stopping Rule via Duality

The dual problem of the approximate problem (P_ε) can be computed (we leave this to the reader) and we obtain

$$(\mathrm{D}_\varepsilon) \quad \max_{y \in \mathbb{R}^m} d_\varepsilon(y) := \sum_{i=1}^N r_i^T y_i + \varepsilon\sqrt{1 - \|y_i\|_{D_i^{-1}}^2}$$

subject to

$$\sum_{i=1}^N Q_i^T y_i = 0,$$

$$\|y_i\|_{D_i^{-1}} \le 1, \qquad i = 1, \dots, N.$$

Let the optimal solution of (D_ε) be denoted by y_ε^*. Recall that x_ε^* is the optimal solution of (P_ε). By the duality theory of convex programming,

$$\max(\mathrm{D}_\varepsilon) = d_\varepsilon(y_\varepsilon^*) = f_\varepsilon(x_\varepsilon^*) = \min(\mathrm{P}_\varepsilon). \tag{5.1}$$

The dual solution y_ε^* can be expressed explicitly, in terms of the primal solution x_ε^*, by

$$(y_\varepsilon^*)_i = \frac{D_i(Q_i x_\varepsilon^* - r_i)}{\sqrt{\|Q_i x_\varepsilon^* - r_i\|_{D_i}^2 + \varepsilon^2}} = \mu_i(x_\varepsilon^*, \varepsilon) D_i(Q_i x_\varepsilon^* - r_i). \tag{5.2}$$

To verify this, note that by (3.2) it follows that $\sum_{i=1}^N Q_i^T (y_\varepsilon^*)_i = 0$, and also

$$\|(y_\varepsilon^*)_i\|_{D_i^{-1}} = \frac{\|Q_i x_\varepsilon^* - r_i\|_{D_i}^2}{\|Q_i x_\varepsilon^* - r_i\|_{D_i}^2 + \varepsilon^2} \leq 1. \tag{5.3}$$

Thus, y_ε^* is feasible for (D_ε). Moreover, $d_\varepsilon(y_\varepsilon^*) = f_\varepsilon(x_\varepsilon^*)$, hence y_ε^* is optimal for (D_ε).

Let $\delta > 0$ be a small positive number

$$\delta \leq \varepsilon^2. \tag{5.4}$$

Suppose that the following inequalities hold:

$$\|Q_i x_k - r_i\|_{D_i}^2 - \|Q_i x_{k-1} - r_i\|_{D_i}^2 < \delta, \qquad i = 1, \ldots, N. \tag{5.5}$$

Since the sequence $\{x_k\}$ generated by the CSSA algorithm converges, the inequalities (5.5) indeed hold for sufficiently large k. Recall that x_k is obtained as the solution of the weighted normal equations:

$$\sum_{i=1}^N \mu_i(x_{k-1}, \varepsilon) Q_i^T D_i Q_i x_k = \sum_{i=1}^N \mu_i(x_{k-1}, \varepsilon) Q_i^T D_i r_i. \tag{5.6}$$

Define vectors $y_i^k \in \mathbb{R}^p$ $(i = 1, \ldots, N)$ by

$$y_i^k = \mu_i(x_{k-1}, \varepsilon) D_i(Q_i x_k - r_i). \tag{5.7}$$

By (5.6),

$$\sum_{i=1}^N Q_i^T y_i^k = 0.$$

Moreover, by (5.4), (5.5),

$$\|y_i^k\|_{D_i^{-1}} = \frac{\|Q_i x_k - r_i\|_{D_i^{-1}}^2}{\|Q_i x_{k-1} - r_i\|_{D_i^{-1}}^2 + \varepsilon^2} \leq 1, \tag{5.8}$$

hence y^k is a feasible solution of the dual problem (D_ε), and by weak duality

$$d_\varepsilon(y^k) \leq f_\varepsilon(x_\varepsilon^*) \leq f_\varepsilon(x_k). \tag{5.9}$$

Note that as $k \to \infty$, then $x_k \to x_\varepsilon^*$ and $y^k \to y_\varepsilon^*$ (compare (5.7) with (5.2)), thus the lower and upper bounds in (5.9) become tighter. The above discussion suggests using (5.5) as a *stopping rule* for the CSSA algorithm. If the algorithm stops at iteration k, with the current iterate x_k, and if y^k is computed via (5.7), then the suboptimality of x_k is estimated by

$$0 \leq f_\varepsilon(x_k) - f_\varepsilon(x_\varepsilon^*) \leq f_\varepsilon(x_k) - d_\varepsilon(y^k).$$

Combining inequalities (4.9) and (5.9) it can be easily shown that in order to get an overall prescribed accuracy ε_0 when stopping at x_k, i.e.,

$$f(x_k) - f(x^*) \leq \varepsilon_0,$$

we can choose $\varepsilon = \varepsilon_0/(N + 1)$ and use the stopping rule $f(x_k) - d_\varepsilon(y_k) \leq \varepsilon$.

6. Numerical Examples

Two examples presented in this section serve to illustrate the convergence behavior of the CSSA algorithm given in Section 3 and its efficiency for very large problems. In our implementation, the least-squares problems were solved by preconditioned conjugate gradient methods, see, e.g., [6].

Example 6.1. First we consider a three-bar truss (Figure 1) whose limit solutions have been presented in [13] using Yang's original algorithm. We present the results computed by the new CSSA algorithm. The data for problem (2.4) are

$$B = \begin{pmatrix} \cos\beta & 1 & \cos\beta \\ \sin\beta & 0 & -\sin\beta \end{pmatrix}, \qquad q_0 = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}, \tag{6.1}$$

where $0 < \beta < \pi/2$, $0 \leq \theta \leq 2\pi$ are parameters. The unconstrained l_1-norm minimization problem has an objective function:

$$f(u, v) = |\cos\beta u + \sin\beta v| + |u| + |\cos\beta u - \sin\beta v| + \sigma|\cos\theta u + \sin\theta v - 1|, \tag{6.2}$$

where $\sigma = \|B\|_\infty/\|q_0\|_\infty$ is a constant when the parameters are fixed. The modified objective function is constructed as

$$f_\varepsilon(u, v) = d_1 + d_2 + d_3 + \sigma d_4, \tag{6.3}$$

where

$$d_1 = \sqrt{(\cos\beta u + \sin\beta v)^2 + \varepsilon^2},$$
$$d_2 = \sqrt{u^2 + \varepsilon^2},$$
$$d_3 = \sqrt{(\cos\beta u - \sin\beta v)^2 + \varepsilon^2},$$
$$d_4 = \sqrt{(\cos\theta u + \sin\theta v - 1)^2 + \varepsilon^2}.$$

The fixed-point iteration equation for the CSSA algorithm is

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}^{-1} \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \tag{6.4}$$

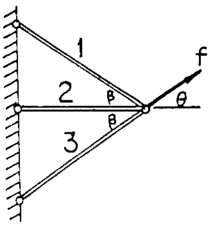


Figure 1. A three-bar truss.

where

$$a_{11} = \cos^2 \beta\left(\frac{1}{d_1} + \frac{1}{d_3}\right) + \frac{\sigma}{d_2} + \cos^2 \theta \frac{1}{d^4},$$

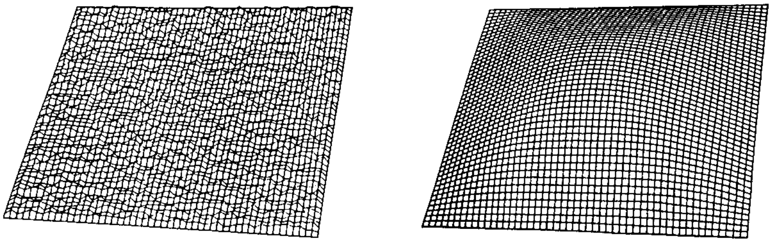
$$a_{22} = \sin^2 \beta\left(\frac{1}{d_1} + \frac{1}{d_3}\right) + \sigma \sin^2 \theta \frac{1}{d_4},$$

$$a_{12} = a_{21} = \cos \beta \sin \beta\left(\frac{1}{d_1} - \frac{1}{d_3}\right) + \sigma \cos \theta \sin \theta \frac{1}{d_4},$$

and

$$b_1 = \sigma \frac{\cos \theta}{d_4}, \qquad b_2 = \sigma \frac{\sin \theta}{d_4}.$$

With an arbitrary (u_0, v_0) as a starting point and a chosen ε, inner iteration may begin. A tolerance on the change of the vector (u, v) and that of the objective function will serve as the stopping criterion for the inner iteration. Then the value of ε is reduced and the iteration begins again with the latest value of (u, v) as its starting point. We can continue to reduce the value of ε in the outer loop until the solution is deemed converged. The algorithm requires only a starting point (u_0, v_0) and a given sequence of ε. A convenient sequence may be chosen for the purpose of extrapolating to the limit, $\varepsilon \to 0$. In order to demonstrate the behavior of convergence we fix the parameters: $\beta = 45$ and $\theta = 0$, 90 degrees. A starting point (4.0, 7.0) and a sequence $\varepsilon = (1.0, 0.1, 0.01, 0.001)$ are chosen, although other



(a) (b)

Figure 2. Initial (a) and first iterated solution (b) of a simply supported plate.

values produce similar results. The sequences of iterates are shown in Table 1, where a fixed limit of five inner iterations is enforced for the purpose of producing a tidy output. After a few iterations, global convergence to an approximate optimal solution and insensitivity to the choice of ε is observed, as was expected from the theoretical analysis. Since this is a small problem (two variables), the subsequent convergence is also rapid as shown in Table 1. It should be pointed out that the $\theta = 0$ case admits nonunique solutions $u = 1$, $-1 \leq v \leq 1$. Different starting point may converge to different values of v in the range. The minimizer for the case $\theta = 90$ is unique. The converged results for 50 parameter variations ($\beta = 15$, 30, 45, 60, 75 and $\theta = 0$–90 in 10-degree increments) are presented in Table 2. The efficiency of the new CSSA algorithm remains excellent at least in the initial stages of the iteration. For moderate accuracy requirements, a few iterations will produce acceptable results without extrapolation. We demonstrate this claim with a very large problem (2500 variables).

Example 6.2. The problem of a simply supported square plate under a uniform load as described in Example 2.2 is solved. We seek the limit value of the multiplier

Table 1. Convergent sequences of the CSSA algorithm.

$\beta = 45°$, $\theta = 0°$			$\beta = 45°$, $\theta = 90°$		
u	v	f	u	v	f_ε
	$\varepsilon = 1.000$			$\varepsilon = 1.000$	
4.00000	7.00000	21.14214	4.00000	7.00000	28.38478
0.88006	0.47487	2.41421	0.27646	0.96103	1.72965
0.99005	0.14070	2.41421	0.04117	0.99915	1.45623
0.99892	0.04648	2.41421	0.00617	0.99998	1.42040
0.99988	0.01548	2.41421	0.00092	1.00000	1.41514
	$\varepsilon = 0.100$			$\varepsilon = 0.100$	
0.99999	0.00516	2.41421	0.00014	1.00000	1.41435
0.99999	0.00506	2.41421	0.00002	1.00000	1.41423
0.99999	0.00496	2.41421	0.00000	1.00000	1.41422
0.99999	0.00486	2.41421	0.00000	1.00000	1.41421
0.99999	0.00477	2.41421	0.00000	1.00000	1.41421
	$\varepsilon = 0.010$			$\varepsilon = 0.010$	
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
	$\varepsilon = 0.001$			$\varepsilon = 0.001$	
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421
0.99999	0.00467	2.41421	0.00000	1.00000	1.41421

Table 2. Converged solutions for the range of β and θ.

θ	u^*	v^*	$f_{\min}$	θ	u^*	v^*	$f_{\min}$
	$\beta = 15°$				$\beta = 60°$		
0	0.99949	0.03202	2.93185	0	1.00000	0.00023	2.00000
10	0.25862	0.96598	2.45200	10	0.86679	0.49867	1.85337
20	0.25133	0.96790	2.02512	20	0.86619	0.49972	1.76265
30	0.05244	0.99862	1.89829	30	0.86603	0.50000	1.73205
40	0.00152	1.00000	1.56216	40	0.86586	0.50028	1.76265
50	0.00049	1.00000	1.20310	50	0.86526	0.50133	1.85337
60	0.00031	1.00000	0.91029	60	0.50000	0.86603	2.0000
70	0.00019	1.00000	0.69445	70	0.00154	1.00000	1.85337
80	0.00009	1.00000	0.56222	80	0.00032	1.00000	1.76265
90	0.00000	1.00000	0.51764	90	0.00000	1.00000	1.73205
	$\beta = 30°$				$\beta = 75°$		
0	0.99984	0.01804	2.73205	0	1.00000	0.00000	1.51764
10	0.50023	0.86589	2.34202	10	0.96774	0.25194	1.47545
20	0.49980	0.86614	2.00552	20	0.96607	0.25827	1.47363
30	0.49921	0.86648	1.73326	30	0.96598	0.25827	1.53195
40	0.44206	0.89698	1.57641	40	0.96591	0.25889	1.64699
50	0.03973	0.99921	1.60541	50	0.96581	0.25927	1.81554
60	0.00095	1.00000	1.36555	60	0.93431	0.35646	2.05937
70	0.00039	1.00000	1.16479	70	0.00466	0.99999	2.05058
80	0.00018	1.00000	1.04160	80	0.00036	1.00000	1.96141
90	0.00000	1.00000	1.00000	90	0.00000	1.00000	1.93185
	$\beta = 45°$						
0	0.99999	0.00467	2.41421				
10	0.70752	0.70670	2.14392				
20	0.70717	0.70704	1.03336				
30	0.70692	0.70729	1.78961				
40	0.70654	0.70767	1.71681				
50	0.65908	0.75207	1.72457				
60	0.03348	0.99944	1.72983				
70	0.00065	1.00000	1.55994				
80	0.00026	1.00000	1.45104				
90	0.00000	1.00000	1.41421				

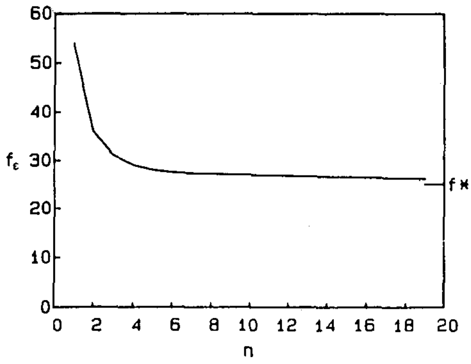


Figure 3. The convergence of the objective function $f_\varepsilon(v(x, y))$.

λ^* (collapse load) and the limit solution $v^*(x, y)$ (collapse mode). To dramatize the presentation of the stable and fast initial convergence behavior, we choose a randomized starting point $v_0(x, y)$ as shown in Figure 2(a). After one iteration, the random data are changed to an orderly data as shown in Figure 2(b). Although the $v(x, y)$ shown in Figure 2(b) is not a converged solution, it is hardly distinguishable from the shape of the optimal minimizer $v^*(x, y)$.

Acknowledgment

We are grateful to the referee for his constructive comments and suggestions which resulted in a substantial improvement of an earlier version of the paper.

References

1. E. F. Beckenbach and R. Bellman, Inequalities, Springer-Verlag, New York, 1967.
2. A. Ben-Tal and M. Teboulle, A geometric property of the least squares solution of linear equations, Linear Algebra Appl., 139 (1990), 165–170.
3. P. H. Calamai and A. R. Conn, A projected Newton method for l_p norm location problems, Math. Programming, 38 (1987), 75–109.
4. P. Concus, Numerical solution of minimal surface equation, Math. Comp., 21 (1967), 340–350.
5. R. A. El-Attar, M. Vidyasagar, and S. R. K. Dutta, An algorithm for l_1-norm minimization with application to nonlinear l_1-approximation, SIAM J. Numer. Anal., 16 (1979), 70–86.
6. G. H. Golub and C. F. Van Loan, Matrix Computations, The Johns Hopkins University Press, Baltimore, Maryland, 1989.
7. H. W. Kuhn, A note on Fermat's problem, Math. Programming, 4 (1973), 98–107.
8. C. Michelot and O. Lefebvre, A primal-dual algorithm for the Fermat–Weber problem involving mixed gauges, Math. Programming, 39 (1987), 319–335.
9. J. G. Morris, Convergence of the Weiszfeld algorithm for Weber problems using a generalized distance function, Oper. Res., 29 (1981), 37–48.
10. R. T. Rockafellar, Convex Analysis, Princeton University Press, Princeton, New Jersy, 1970.
11. E. Weiszfeld, Sur le point pour lequel la somme des distances de n points donnes est minimum, Tohoku Math. J., 43 (1937), 355–386.
12. W. H. Yang, A variational principle and an algorithm for limit analysis of beams and plate, Comput. Methods Appl. Mech. Engrg., 33 (1982), 575–582.
13. W. H. Yang, A general computational method for limit analysis, ASME Publ., 129 (1987), 21–31.
14. W. H. Yang, A generalized von Mises criterion for yield and fracture, J. Appl. Mech., 47 (1980), 297–300.

Accepted 6 December 1990