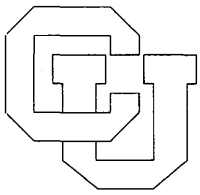

University of Colorado at Denver

Improving the Robustness of Complementarity Solvers Using Proximal Perturbations

Stephen C. Billups

December 1996

UCD/CCM Report No. 96

CENTER FOR COMPUTATIONAL MATHEMATICS REPORTS

Improving the Robustness of Complementarity Solvers Using Proximal Perturbations

Stephen C. Billups *

December 30, 1996

Abstract

A common difficulty encountered by many solvers for complementarity problems is convergence to a local (but not global) minimum of an underlying merit function. Such difficulties can be avoided by using a proximal perturbation strategy, which allows the iterates to escape the local minimum in a controlled fashion. This paper presents the proximal perturbation strategy for a general class of complementarity solvers and proves subsequential convergence to a solution based upon a pseudo-monotonicity assumption. An example algorithm is presented, called PROXI, which uses an extremely simple (but not very robust) basic algorithm enhanced by the proximal perturbation strategy. Test results on the MCPLIB and GAMSLIB problem libraries demonstrate that PROXI is not only robust but also computationally efficient.

1 Introduction

A popular technique for solving complementarity problems is to reformulate the problem as a system of nonsmooth equations $H(x) = 0$, where $H : \mathrm{R}^n \to \mathrm{R}^n$ is a function whose zeros correspond to solutions of the complementarity problem. This system of equations is then solved using a direction-linesearch technique, which forces the iterates to produce monotonic decrease of a merit function $\theta : \mathrm{R}^n \to \mathrm{R}$, which is usually defined by $\theta(x) := \frac{1}{2} \|H(x)\|^2$. An obvious difficulty of this approach is that the iterates may converge to a local, rather than a global, minimum of the merit function.

To overcome this difficulty, it is useful to relax the requirement that the value of the merit function decreases with every iteration. One way of accomplishing this is to use a nonmonotone linesearch, a technique which was first proposed by (Grippo, Lampariello & Lucidi 1986) and which has been used in the complementarity framework by the PATH solver Dirkse & Ferris (1995) and by (De Luca, Facchinei & Kanzow 1996). A more aggressive technique is the watchdog strategy of (Chamberlain, Powell & Lemaréchal 1982). This technique was employed with considerable success in the PATH solver.

The use of these techniques has traditionally been justified as a means of speeding up *local convergence.* It is somewhat surprising that the ability of these techniques to improve global convergence behavior has not, to our knowledge, been acknowledged. Because these techniques allow the value of the merit function to increase (within some limits), they provide a means for allowing the iterates to "climb out" of valleys surrounding local minima. Thus, the use of these techniques can significantly improve global convergence behavior.

Unfortunately, the nonmonotone linesearch and watchdog techniques do not provide a guarantee that the iterates will escape local minima. In contrast, a theoretically more reliable approach was

*Department of Mathematics, University of Colorado, Denver, Colorado 80217-3364

introduced by (Billups & Ferris 1996). That paper presented a proximal perturbation strategy which was used to improve the robustness of the NE/SQP algorithm (Pang & Gabriel 1993). The strategy was shown to guarantee subsequential convergence to a solution based upon a pseudo-monotonicty assumption. The resulting algorithm, called QPCOMP was found to be significantly more robust than NE/SQP at no cost in efficiency. That is, on problems that NE/SQP could solve, QPCOMP was equally as fast.

The proximal perturbation strategy used by QPCOMP, while promising conceptually, suffers from being specialized to the particular reformulation of the complementarity problem (as a system of equations) used by the NE/SQP algorithm. In this paper, we present a different form of the proximal perturbation strategy, which can be applied to *any* reformulation. The essential theory for this strategy is given in Section 3.

For ease of discussion, we will refer to the solver being enhanced as the inner algorithm and the proximal perturbation strategy as the outer algorithm. The efficiency of the approach is governed in large part by the efficiency of the inner algorithm. For example, NE/SQP, the inner algorithm for QPCOMP, requires solving a quadratic program at each iteration, and is therefore fairly slow. As a result, the efficiency of the QPCOMP algorithm is not competitive with several other complementarity solvers tested in (Billups, Dirkse & Ferris 1996).

To improve on this situation, we discuss, in Section 4, a class of inner algorithms based on the generalized Jacobian-based damped Newton method of (Qi 1993). These algorithms have the advantage that only a single linear equation must be solved at each iteration. However, the global convergence properties of these algorithms are very poor.

Fortunately, this relative lack of robustness is inconsequential when the algorithm is enhanced by the proximal perturbation strategy. In Section 5 we describe the complete proximal perturbation strategy and prove strong global convergence properties when used in conjunction with the inner algorithm described in Section 4.

In Section 6 we present an example inner algorithm called sBDIFF which can be viewed as a simplification of the BDIFF algorithm of (Pang 1991). As might be expected, this algorithm fails on many of our test problems. However, when enhanced by the proximal perturbation strategy, the resulting algorithm, called PROXI, is found to be remarkably robust. Moreover, because of the underlying efficiency of sBDIFF, PROXI is also found to be competitive with PATH (Dirkse & Ferris 1995) and SMOOTH (Chen & Mangasarian 1995), the current state-of-the-art solvers (Billups et al. 1996).

Before we begin, a word about notation is in order. Iteration numbers appear as superscripts on vectors and matrices and as subscripts on scalars. Subscripts on a vector (or matrix) represent either subvectors (or submatrices) or components of the vector or matrix . For example, if M is an $n \times n$ matrix with elements $M_{jk}, j, k = 1, \ldots, n$, and J and K are index sets such that $J, K \subset \{1, \ldots, n\}$, then $M_{J,K}$ denotes the $|J| \times |K|$ submatrix of M consisting of the elements $M_{jk}, j \in J, k \in K$. Similarly, x_j represents the jth component of the vector x. The notation x_+ and x_- refers to the positive and negative components of the vector x. Specifically, x_+ is the vector whose ith component is given by $\max(x_i, 0)$, and $x_- := x_+ - x$.

The directional derivative of a function $f : \mathrm{B} \to \mathrm{R}^n$ evaluated at the point x in the direction d is denoted by

$$f'(x; d) :- \lim_{\lambda \downarrow 0} \frac{f(x + \lambda d) - f(x)}{\lambda}, \quad -$$

provided the limit exists. Note that if x is a stationary point of f on B, then $f'(x; d) = 0$ for all d such that $x + d \in \mathrm{B}$. The Euclidean norm is denoted by $\|\cdot\|$, while inner products are denoted by $\langle \cdot, \cdot \rangle$. Finally, we use the symbol R_+ to represent the nonnegative real numbers.

2 Background

Complementarity problems appear in many different forms (Ferris & Pang 1995). The form we find most appealing, from a practical standpoint is the mixed complementarity problem (MCP). This problem is defined in terms of a box $\mathrm{B} := \prod_{i=1}^{n}[l_i, u_i]$ (where for each i, $-\infty \leq l_i < u_i \leq \infty$), and a function $f : \mathrm{B} \to \mathrm{R}^n$. The problem $\mathrm{MCP}(f, \mathrm{B})$ is then to find $x \in \mathrm{B}$ such that

$$(x - l)^\top f(x)_+ = 0 \text{ and } (u - x)^\top f(x)_- = 0.$$

For simplicity, we will assume for the remainder of the paper that f is defined on all of R^n.

In the above definition, if we agree that $\pm\infty \times 0 = 0$, the definition includes the case of infinite bounds. For example, by choosing $l = 0$ and $u = \infty$, the MCP reduces to the standard nonlinear complementarity problem (NCP), which is to find $x \geq 0$ such that

$$f(x) \geq 0 \qquad \text{and} \qquad x^\top f(x) = 0.$$

As mentioned in the introduction, a common paradigm for solving complementarity problems is to define a function $H : \mathrm{R}^n \to \mathrm{R}^n$ such that the zeros of H correspond to solutions of the complementarity problem. As examples, we describe three popular choices of H for NCP. Perhaps the simplest such choice is the min map defined by

$$H_i(x) := \min(x_i, f_i(x)) \qquad i = 1, \ldots, n.$$

This reformulation is the basis of the BDIFF algorithm (Pang 1991), the NE/SQP algorithm (Pang & Gabriel 1993), the SMOOTH algorithm (Chen & Mangasarian 1996) and the QPCOMP algorithm (Billups & Ferris 1996).

Another important reformulation is the normal map (Eaves 1971, Minty 1962, Robinson 1992), given by

$$H(x) := f(\pi_{\mathbf{B}}(x)) - \pi_{\mathbf{B}}(x),$$

where $\pi_{\mathbf{B}}(x)$ is the Euclidean projection of x onto B. This reformulation is the basis of the solvers MILES (Rutherford 1993) and PATH (Dirkse & Ferris 1995).

The third popular reformulation is based on the Fischer-Burmeister function (Fischer 1992, Fischer 1995) $\phi : \mathrm{R}^2 \to \mathrm{R}$ defined by

$$\phi(a, b) = \sqrt{a^2 + b^2} - (a + b).$$

The reformulation is then given by

$$H_i(x) := \phi(x_i, f_i(x)), \qquad i = 1, \ldots, n.$$

This reformulation is the basis of the SEMISMOOTH algorithm of (De Luca et al. 1996).

The inner algorithm we describe in Section 4 relies on the theory of semismooth equations. We therefore summarize some of the more important definitions.

By Rademacher's theorem, if $F : \mathrm{R}^n \to \mathrm{R}^m$ is locally Lipschitzian, it is differentiable almost everywhere. Let D_F be the set where F is differentiable. We define the B-subdifferential by

$$\partial_B F(x) := \left\{ V \;\middle|\; \exists \{x^k\} \to x, x^k \in D_F, \quad \text{with} \quad V = \lim_{k \to \infty} \nabla F(x_k) \right\}.$$

The Clarke subdifferential $\partial F(x)$ is the convex hull of $\partial_B F(x)$.

Definition 2.1 *We say that F is* semismooth *at x if*

$$\lim_{\substack{V \in \partial F(x+th') \\ h' \to h, t \downarrow 0}} \{Vh'\}$$

exists for any $h \in \mathrm{R}^n$.

Definition 2.2 *We say that a semismooth function F is* BD-regular *at x if all elements in $\partial_B F(x)$ are nonsingular.*

Definition 2.3 *Suppose that $F : \mathrm{R}^n \to \mathrm{R}^m$ is B-differentiable in a neighborhood of x. We say that the directional derivative $F'(\cdot;\cdot)$ is* semicontinuous *at x if, for every $\epsilon > 0$, there exists a neighborhood N of x such that, for all $x + h \in N$,*

$$\|F'(x+h;h) - F'(x;h)\| \leq \epsilon \|h\|.$$

We say that $F'(\cdot;\cdot)$ is semicontinuous of degree 2 *at x if there exist a constant L and a neighborhood N of x such that, for all $x + h \in N$,*

$$\|F'(x+h;h) - F'(x;h)\| \leq L \left\|h^2\right\|.$$

3 The Proximal Perturbation Strategy

Throughout this paper, we will assume that $H : \mathrm{R}^n \to \mathrm{R}^n$ is a semismooth function whose zeros correspond to solutions of MCP(f, B) and that $\theta(\cdot) := \frac{1}{2}\|H(\cdot)\|^2$.

The proximal perturbation strategy can be viewed as an enhancement to a basic descent based method that is only activated when the basic method appears to be failing. Such a situation occurs, for example, when the the decrease in the merit function $\theta(x^k)$ at each iteration becomes too small. In this case, we record the current value of the merit function as θ_{best} and invoke the proximal perturbation strategy.

The goal of the proximal perturbation strategy is to find an improved starting point $\tilde{x}$ where $\theta(\tilde{x})$ is sufficiently smaller than θ_{best}. Since the descent-based algorithm never allows the value of θ to increase, the algorithm can be restarted from $\tilde{x}$ with the guarantee that the iterates will never return to region where the descent-based algorithm stalled.

The idea behind the proximal perturbation strategy is to solve a sequence of perturbed subproblems whose solutions lead to an improved starting point. The subproblems are based upon perturbing the function H as follows: given a *centering point* $\bar{x} \in \mathrm{R}^n$, and a perturbation parameter $\lambda > 0$, define

$$H^{\lambda,\bar{x}}(x) := H(x) + \lambda(x - \bar{x}).$$

For λ sufficiently large, this perturbed function is dominated by the $\lambda(x - \bar{x})$ term. Thus, it is reasonable to expect that even very unreliable algorithms would be able to find zeros of $H^{\lambda,\bar{x}}$ provided that λ is suitably large.

For a sequence of positive numbers $\{\lambda_k\}$, we generate a sequence of iterates as follows: given a point x^0, then for $k = 0, \ldots,$ choose x^{k+1} to solve

$$H^{\lambda_k,x^k}(x) = 0.$$

Note that every subproblem in this sequence uses a different centering point. In particular the centering point for one subproblem is the solution of the previous subproblem.

Once a new point x^k is found such that $\theta(x^k)$ is sufficiently smaller than θ_{best}, we can then safely return to the basic algorithm without any danger of the iterates returning to the troublesome region.

In order to be sure that the strategy will work, we need some guarantee that the iterates will eventually produce an improved starting point. Such a guarantee can be made if the function H satisfies a pseudo-monotonicity assumption.

Definition 3.1 *A function $H : \mathrm{R}^n \to \mathrm{R}^n$ is said to be* pseudo-monotone *at a point x^* if $\forall y \in \mathrm{R}^n$,*

$$\langle H(x^*), y - x^* \rangle \geq 0 \quad \textit{implies} \quad \langle H(y), y - x^* \rangle \geq 0. \tag{1}$$

H is said to be pseudo-monotone *if it is pseudo-monotone at every point in R^n.*

It is known (Harker & Pang 1990) that if a function $g : \mathrm{R}^n \to \mathrm{R}$ is pseudo-convex (Mangasarian 1969, Definition 9.3.1), then ∇g is a pseudo-monotone function. However, if g is only pseudo-convex at a point x^*, it does not necessarily follow that ∇g is pseudo-monotone at x^*.

Pseudo-monotonicity is a weaker condition than monotonicity. In particular every monotone function is pseudo-monotone. But the converse is not true. For example, consider the function $H(x) := x/2 + sin(x)$. This function is pseudo-monotone, but is not monotone. Note further that the natural merit function $\|H(x)\|^2/2$ has strict local minima that are not global minima. Thus, we see that the natural merit function of a pseudo-monotone function can have local minima that are not global minima.

In order to guarantee global convergence of our algorithm we shall require that the following assumption be satisfied:

Assumption 3.2 *The equation $H(x) = 0$ has a solution x^* such that H is pseudo-monotone at x^*.*

If H satisfies Assumption 3.2, we say that H is *pseudo-monotone at a solution.*

One potential criticism of this assumption is that it is a condition placed on the reformulation H rather than the original problem MCP(f, B). However, for many reasonable choices of H, the pseudo-monotonicity of H is implied by the pseudo-monotonicity of the function f at a solution to MCP(f, B).

The main convergence result for the proximal perturbation strategy is given by the following theorem:

Theorem 3.3 *Let $\{\lambda_k\}$ be a bounded sequence of positive numbers and let $\{x^k\}, k = 0, 1, \ldots$ be a sequence of points in R^n such that for each k, $H^{\lambda_k, x^k}(x^{k+1}) = 0$. If H is pseudo-monotone at a solution x^*, then*

1. *$\{x^k\}$ has a subsequence that converges to a zero $\bar{x}$ of H.*
2. *Every accumulation point of $\{x^k\}$ is a zero of H.*
3. *If H is pseudo-monotone at any accumulation point $\bar{x}$ of $\{x^k\}$, then the iterates converge to $\bar{x}$.*

Proof For any given k,

$$H^{\lambda_k,x^k}(x^{k+1}) = H(x^{k+1}) + \lambda(x^{k+1} - x^k) = 0.$$

Thus, $x^k - x^{k+1} = H(x^{k+1})/\lambda$ and

$$\begin{aligned}\left\|x^k - x^*\right\|^2 &= \left\|x^{k+1} - x^* + x^k - x^{k+1}\right\|^2 \\ &= \left\|x^{k+1} - x^*\right\|^2 + 2\left\langle x^{k+1} - x^*, x^k - x^{k+1}\right\rangle + \left\|x^k - x^{k+1}\right\|^2 \\ &= \left\|x^{k+1} - x^*\right\|^2 + 2\left\langle x^{k+1} - x^*, H(x^{k+1})\right\rangle/\lambda + \left\|x^k - x^{k+1}\right\|^2.\end{aligned}$$

Under Assumption 3.2, the inner product term above is nonnegative. Thus,

$$\left\|x^k - x^*\right\|^2 \geq \left\|x^{k+1} - x^*\right\|^2 + \left\|x^k - x^{k+1}\right\|^2,$$

so $\left\{\left\|x^k - x^*\right\|\right\}$ is a decreasing sequence, and $\left\|x^k - x^{k+1}\right\| \to 0$. It follows that $\{x^k\}$ has an accumulation point. Let $\bar{x}$ be any accumulation point of $\{x^k\}$. Then there is a subsequence $\{x^{k_j} : j = 0, 1, \ldots\}$ converging to $\bar{x}$. Since $\left\|x^k - x^{k+1}\right\| \to 0$ and since $\{\lambda_k\}$ is bounded, $\lambda_k(x^{k+1} - x^k) \to 0$. Thus, since $0 = H^{\lambda_k,x^{k_j}}(x^{k_j+1}) = H(x^{k_j+1}) + \lambda_k(x^{k_j+1} - x^{k_j})$, we conclude by continuity that $H(\bar{x}) = 0$. This proves items 1 and 2.

To prove item 3, note that if H is pseudo-monotone at an accumulation point $\bar{x}$, then by item 2, $\bar{x}$ is a solution, so the above analysis can be repeated with x^* replaced by $\bar{x}$. We can then conclude that $\left\{\left\|x^k - \bar{x}\right\|\right\}$ is a decreasing sequence. But since $\bar{x}$ is an accumulation point of $\{x^k\}$, it follows that $\left\|x^k - \bar{x}\right\| \to 0$, so $x^k \to \bar{x}$. □

The first part of Theorem 3.3 guarantees that $\|H(x)\|$ will become arbitrarily small. Thus, the iterates generated by the proximal perturbation strategy will eventually produce a merit function value that is smaller than θ_{best}.

At this point we note that the proximal perturbation strategy is very similar to the proximal point algorithm (Rockafellar 1978), which has a reputation of being very slow. This slowness is due to the fact that for $\{\lambda_k\}$ bounded away from zero, the proximal term $(\lambda_k(x^{k+1} - x^k))$ destroys any chance of achieving anything better than a linear rate of convergence. However, in our approach, we overcome this difficulty by using the proximal perturbations only to escape local minima rather than to find an exact solution.

We also note that the strategy bears some resemblance to Tikhonov regularization (Tikhonov & Arsenin 1977). However, Tikhonov regularization only perturbs the quadratic term of the merit function, leaving the linear term unchanged. The fact that the proximal perturbation strategy affects the linear term is crucial to the success of the algorithm.

To illustrate the technique, it is useful to look at a simple example. Let $\mathrm{B} := \mathrm{R}_+$ and let $f : \mathrm{R} \to \mathrm{R}$ be defined by

$$f(x) = (x - 1)^2 - 1.01.$$

This deceptively simple problem proved intractable for most of the descent-based methods tested in (Billups et al. 1996). But this should not be surprising since f is not monotone. However, the function $H := \min(x, f)$ is pseudo-monotone. Thus, the problem is easily solved by our technique. For example, using $\lambda = 1.1$ and a starting point $x^0 = 0$, the strategy generates the sequence of iterates shown in Table 3.

Note that at the 7th iteration, an improved starting point is found, (i.e, $\theta(x^7) < \theta(x^0)$). At this point, a fast inner algorithm (e.g., Newton's method) can be used to obtain the final solution.

Table 1: Iterates produced by solving sequence of perturbed problems, with ($\lambda = 1.1$)

k	x^k	$f(x^k)$	$\theta(x^k)$
0	0	-.01	.00005
1	.9110	-1.0021	.5021
2	1.5521	-.7052	.2487
3	1.8356	-.3118	.0486
4	1.9439	-.1191	.0071
5	1.9832	-.0433	.00094
6	1.9973	-.0155	.00012
7	2.0023	-.0055	.00002

4 The Inner Algorithm

One weakness of the QPCOMP algorithm is that each iterate is expensive to compute. In particular the solution to a quadratic program is required at each iteration. In this section, we discuss a more efficient inner algorithm based on the generalized Jacobian-based Newton method described by (Qi 1993). This algorithm is given in Figure 1.

Figure 1: Inner Algorithm

Step 1 [Initialization] Select $\rho, \sigma \in (0,1)$, a positive integer m_{max}, a starting vector $x^0 \in \mathrm{R}^n$, and a stopping tolerance $tol > 0$. Set $k = 0$.

Step 2 [Direction generation] Choose $V^k \in \partial_B H(x^k)$. If V^k is singular, stop, returning the point x^k along with a failure message. Otherwise choose the direction

$$d^k = -(V^k)^{-1} H(x^k). \tag{2}$$

Step 3 [Steplength determination] Let m_k be the smallest nonnegative integer $m \leq m_{max}$ such that

$$\theta(x^k + \rho^m d^k) - \theta(x^k) \leq -2\sigma\rho^m \theta(x^k). \tag{3}$$

If no such m_k exists, stop, returning the point x^k along with a failure message. Otherwise set $x^{k+1} = x^k + \rho^{m_k} d^k$.

Step 4 [Termination check] If $\theta(x^{k+1}) < tol$ stop, returning the point x^{k+1}. Otherwise, return to Step 2, with k replaced by $k+1$.

The algorithm has four features that make it attractive for use with the proximal perturbation strategy.

1. The calculation of the search direction at each iteration is very cheap; it only requires solving a single linear equation.

2. When the algorithm fails, it fails in a finite number of steps. This claim is justified by the following proposition, which is an obvious consequence of the upper bound m_{max} placed on m_k in the steplength determination step:

   Proposition 4.1 *The algorithm will either fail in a finite number of steps, or will produce a sequence of iterates $\{x^k\}$ such that*

   $$\theta(x^{k+1}) \leq (1 - 2\sigma\rho^{m_{max}})\theta(x^k).$$

   This feature is essential for use in the proximal perturbation strategy since when the inner algorithm fails on a problem, we want to try again with a perturbed problem.

3. The algorithm has fast local convergence behavior, which will be summarized in Theorem 4.2.

4. Assuming H is Lipschitz continuous, the algorithm will be guaranteed to solve the perturbed equations $H^{\lambda,\bar{x}}(x) = 0$ for all λ sufficiently large. This feature will be proved in Proposition 4.11.

The fast local convergence of the method is given by the following theorem:

Theorem 4.2 *[(Qi 1993) Theorem 3.1] Suppose that x^* is a solution of $H(x) = 0$, and that H is semismooth and BD-regular at x^*. Then the iteration method defined by $x^{k+1} = x^k + d^k$, where d^k is given by (2) is well defined and convergent to x^* superlinearly in a neighborhood of x^*. In addition, if $H(x^k) \neq 0$ for all k, then*

$$\lim_{k\to\infty} \frac{\left\|H(x^{k+1})\right\|}{\|H(x^k)\|} = 0.$$

If, in addition, H is directionally differentiable at a neighborhood of x^* and $H'(\cdot;\cdot)$ is semicontinuous of degree 2 at x^*, then the convergence of the iteration method is quadratic.

One consequence of this local convergence theorem is that within a neighborhood of a BD-regular solution x^*, the linesearch criteria (3) will be satisfied by $m_k = 0$. Thus, the inner algorithm will take full Newton steps and achieve the fast local convergence rates specified by the theorem.

While strong local convergence properties of the method are easily established, global convergence properties are harder to achieve. The problem is that at each iteration, the particular element V^k of $\partial_B H(x^k)$ is chosen *prior* to calculating the direction d^k. This has the advantage that calculating d^k only requires solving a linear equation. However, it creates difficulties in achieving global convergence.

In contrast, (Han, Pang & Rangaraj 1992) established global convergence results for a slight variant of (2) given by

$$H(x^k) + G(x^k, d^k) = 0, \tag{4}$$

where $G : \mathrm{R}^n \times \mathrm{R}^n \to \mathrm{R}^n$ is a function satisfying the two assumptions:

Assumption 4.3 *For each $x \in \mathrm{R}^n$, $G(x,\cdot)$ is surjective; and*

Assumption 4.4 *$H(x)^\top G(x,d) \geq \theta^D(x;d)$ for all $(x,d) \in \mathrm{R}^n \times \mathrm{R}^n$.*

where θ^D is the Dini directional derivative of θ.

Several examples of functions G satisfying these two assumptions were given; however, for each of these functions, the system (4) does not reduce to a linear equation. More importantly, for our choice of G, namely $G(x,d) := Vd, V \in \partial_B H(x)$, the second assumption is not, in general, satisfied.

We therefore endeavor to prove a weak global convergence result based on the following assumption:

Assumption 4.5 *There exists a constant $\bar{\sigma}$ such that for any point $x \in \mathrm{R}^n$ and for any two matrices $V \in \partial_B H(x), W \in \partial H(x)$*

$$\left\|(W - V)V^{-1}\right\| \leq 1 - \bar{\sigma}.$$

This assumption places a restriction on how severe the nonsmoothness at a particular point can be. If, for example, an element V of $\partial_B H(x)$ is nearly singular, then the function needs to be nearly smooth at x so that $\|W - V\|$ will be small for all $W \in \partial_B H(x)$. In contrast, if $\|V^{-1}\|$ is large for all $V \in \partial_B H(x)$, then we can allow a much larger "kink" at x; that is, we can allow $\|(W - V)\|$ to be much larger, for $W \in \partial_B H(x)$.

It should be noted that this assumption places a significant restriction on H. In particular, even strong monotonicity of H is not sufficient to satisfy the assumption for fixed values of $\bar{\sigma}$. Nevertheless, in the context of the proximal perturbation strategy, the assumption is easily satisfied by creating perturbed problems with sufficiently larger perturbation parameters λ.

Lemma 4.6 *Suppose H satisfies Assumption 4.5, and let $\{z^k\}$ and $\{y^k\}$ be sequences converging to $\bar{x}$, where $\partial H(\bar{x})$ is bounded. Let $W^k \in \partial H(z^k)$ and $V^k \in \partial_B H(y^k)$, then*

$$K := \limsup_{k\to\infty} \left\|(W^k - V^k)(V^k)^{-1}\right\| \leq 1 - \bar{\sigma}.$$

Proof By the semismoothness of H, $\{W^k\}$ and $\{V^k\}$ are bounded. After going to a subsequence, we may assume that $W^k \to W$ and $V^k \to V$, where $W \in \partial H(\bar{x})$ and $V \in \partial_B H(\bar{x})$ and

$$K = \lim_{k\to\infty} \left\|(W^k - V^k)(V^k)^{-1}\right\| = \left\|(W - V)V^{-1}\right\| \leq 1 - \bar{\sigma},$$

where the last inequality follows from Assumption 4.5. □

Lemma 4.7 *Suppose H is semismooth and satisfies Assumption 4.5 and suppose $\{x^k\}$ converges to $\bar{x}$, $V^k \in \partial_B H(x^k)$, $d^k := -(V^k)^{-1}H(x^k)$, and $\{\tau_k\}$ is a sequence of positive scalars converging to zero. If $\{d^k\}$ is bounded, then*

$$\limsup_{k\to\infty} \frac{\theta(x^k + \tau_k d^k) - \theta(x^k)}{\tau_k} \leq -2\bar{\sigma}\theta(\bar{x}).$$

Proof By the mean value theorem, (Clarke 1983, Theorem 2.3.7)

$$\theta(x^k + \tau_k d^k) = \theta(x^k) + \tau_k w^k d^k,$$

where w^k is an element of $\partial\theta(z^k)$ for some point $z^k \in (x^k, x^k + \tau_k d^k)$. But, $\partial\theta(z^k) = H(z^k)^\top \partial H(z^k)$, so $w^k = H(z^k)^\top W^k$ for some $W^k \in \partial H(z^k)$. Thus,

$$\theta(x^k + \tau_k d^k) \;=\; \theta(x^k) + \tau_k H(z^k)^\top W^k d^k$$

$$
\begin{aligned}
&= \theta(x^k) - \tau_k H(z^k)^\top (I + (W^k - V^k)(V^k)^{-1}) H(x^k) \\
&= (1 - 2\tau_k)\theta(x^k) - \tau_k (H(z^k) - H(x^k))^\top H(x^k) \\
&\quad -\tau_k H(z^k)^\top (W^k - V^k)(V^k)^{-1} H(x^k) \\
&\leq (1 - 2\tau_k)\theta(x^k) + \tau_k \left\| H(z^k) - H(x^k) \right\| \left\| H(x^k) \right\| \\
&\quad +\tau_k \left\| H(z^k)^\top \right\| \left\| (W^k - V^k)(V^k)^{-1} \right\| \left\| H(x^k) \right\|.
\end{aligned}
$$

Subtracting $\theta(x^k)$ from both sides and dividing by τ_k, we get

$$
\begin{aligned}
\frac{\theta(x^k + \tau_k d^k) - \theta(x^k)}{\tau_k} &\leq -2\theta(x^k) + \left\| H(z^k) - H(x^k) \right\| \left\| H(x^k \right\| \\
&\quad + \left\| H(z^k)^\top \right\| \left\| (W^k - V^k)(V^k)^{-1} \right\| \left\| H(x^k) \right\|.
\end{aligned}
$$

Taking the limit as $k \to \infty$, and invoking Lemma 4.6, the lemma is proved. □

Corollary 4.8 *Let H satisfy Assumption 4.5 with $\bar{\sigma} > \sigma \in (0,1)$. Suppose that $H(x^k) \neq 0$ and that $d^k = (V^k)^{-1} H(x^k)$ for $V^k \in \partial_B H(x^k)$. Then there exists a scalar $\tau > 0$ such that for all $\tau \in [0, \bar{\tau}]$,*

$$\theta(x^k + \tau d^k) - \theta(x^k) \leq -2\sigma\tau\theta(x^k).$$

Proof Assume the contrary. Then there exists a sequence $\{\tau_j\}$ converging to zero such that for each j,

$$\theta(x^k + \tau_j d^k) - \theta(x^k) > -2\sigma\tau_j\theta(x^k).$$

Dividing both sides by τ_j and taking the limit as $j \to \infty$, we get

$$\limsup_{j \to \infty} \frac{\theta(x^k + \tau_j d^k) - \theta(x^k)}{\tau_j} \geq -2\sigma\theta(x^k) > -2\bar{\sigma}\theta(x^k).$$

But this contradicts Lemma 4.7. □

This corollary guarantees that if we choose $m_{max} = \infty$, then the linesearch rule (3) will always be satisfied eventually. We can now state our main global convergence theorem.

Theorem 4.9 *Let H satisfy assumption 4.5 with $\bar{\sigma} > \sigma$ and let $\{x^k\}$ be a bounded sequence produced by the inner algorithm given in Figure 1, with $m_{max} = \infty$. Then $\lim_{k \to \infty} H(x^k) = 0$.*

Proof By the linesearch rule, $\theta(x^{k+1}) \leq (1 - 2\sigma\tau_k)\theta(x^k)$, where $\tau_k := \rho^{m_k}$ is the steplength in the kth iteration. Without loss of generality, assume that $\theta(x^k) > 0$ for each k. Thus, the sequence $\{\theta(x^k)\}$ is strictly decreasing and is bounded below; hence it converges and $\{\theta(x^{k+1}) - \theta(x^k)\} \to 0$. Thus,

$$\lim_{k \to \infty} \tau_k \theta(x^k) = \lim_{k \to \infty} \frac{\theta(x^{k+1}) - \theta(x^k)}{-2\sigma} = 0.$$

Hence, if $\limsup_{k \to \infty} \tau_k > 0$, then $\lim_{k \to \infty} \theta(x^k) = 0$.

Now, suppose $\limsup_{k \to \infty} \tau_k = 0$. Since $\{x^k\}$ is bounded, there exists an accumulation point $\bar{x}$ of $\{x^k\}$. After going to a subsequence, we may assume that $\{x^k\}$ converges to $\bar{x}$. Let $\tilde{\tau}_k := \rho^{m_k - 1}$. Then by the definition of m_k,

$$\theta(x^k + \tilde{\tau}_k d^k) - \theta(x^k) > -2\sigma\tilde{\tau}_k\theta(x^k).$$

Dividing both sides by $\tilde{\tau}_k$ and passing to the limit, we get

$$\limsup_{k\to\infty} \frac{\theta(x^k + \tilde{\tau}_k d^k) - \theta(x^k)}{\tilde{\tau}_k} \geq -2\sigma\theta(\bar{x}).$$

But by Lemma 4.7, the left hand side is less than or equal to $-2\bar{\sigma}\theta(\bar{x})$. Since $\bar{\sigma} > \sigma$, we conclude that $\theta(\bar{x}) = 0$. □

Lemma 4.10 *Let x be a point in R^n, let $V \in \partial_B H(x)$ be nonsingular, and choose $\sigma \in (0,1)$. Define $d := -V^{-1}H(x)$ and suppose there exist positive numbers $\delta < 1 - \sigma$ and L such that*

1. $\|(W - V)V^{-1}\| \leq \delta$ *for all $z \in (x, x+d), W \in \partial H(z)$;*
2. $\|H(z) - H(x)\| \leq L\|z - x\|$.

Then for all

$$\tau \leq \frac{1 - \sigma - \delta}{(1+\delta)L\|V^{-1}\|}, \tag{5}$$

it follows that

$$\theta(x + \tau d) - \theta(x) \leq -2\sigma\tau\theta(x). \tag{6}$$

Proof Define $\eta := \|V^{-1}\|$. In similar fashion to the proof of Lemma 4.7, for some $z \in (x, x + \tau d), W \in \partial H(z)$

$$\begin{aligned}
\theta(x+\tau d) &= (1-2\tau)\theta(x) - \tau(H(z) - H(x))^\top H(x) - \tau H(x)^\top (W - V)V^{-1}H(x) \\
&\quad -\tau(H(z) - H(x))^\top (W - V)V^{-1}H(x) \\
&\leq (1-2\tau)\theta(x) + \tau L\|\tau d\|\,\|H(x)\| + 2\tau\delta\theta(x) + \tau L\|\tau d\|\,\delta\,\|H(x)\| \\
&\leq (1 - 2\tau + 2\tau^2 L\eta + 2\tau\delta + 2\tau^2 L\eta\delta)\theta
\end{aligned}$$

Thus, (6) is satisfied if

$$(-2\tau + 2\tau^2 L\eta + 2\tau\delta + 2\tau^2 L\eta\delta)\theta(x) \leq -2\sigma\tau\theta(x).$$

But this inequality is satisfied whenever

$$\tau \leq \frac{1 - \sigma - \delta}{(1+\delta)L\eta}.$$

□

Proposition 4.11 *Suppose that $H : \mathrm{R}^n \to \mathrm{R}^n$ is Lipschitz continuous. Then if m_{max} is chosen so the $\rho^{m_{max}} < 1 - \sigma$, then there exists $\bar{\lambda}$ such that for all $\lambda \geq \bar{\lambda}$ and for any $\bar{x} \in \mathrm{R}^n$, the inner algorithm applied to the perturbed function $H^{\lambda,\bar{x}}(x)$ will generate a sequence of iterates $\{x^k\}$ such that $\theta^{\lambda,\bar{x}}(x^k)$ converges to zero at the linear rate specified by Proposition 4.1.*

Proof Let L be the Lipschitz constant of H. Then $H^{\lambda,\bar{x}}$ is Lipschitz with constant $\tilde{L} \leq L + \lambda$. For any point x and matrices $W, V \in \partial H^{\lambda,\bar{x}}(x)$, $W - V = \tilde{W} - \tilde{V}$, where $\tilde{W}$ and $\tilde{V}$ are elements of $\partial H(x)$. Thus, $\|W - V\| = \left\|\tilde{W} - \tilde{V}\right\| \leq 2L$.

For any $\delta > 0$, by choosing $\lambda \geq L + 2L/\delta$, it follows that $\|V^{-1}\| \leq \delta/2L$, so $\|(W - V)V^{-1}\| \leq \delta$. Thus, by choosing λ sufficiently large, $H^{\lambda,\bar{x}}$ satisfies the assumptions of Lemma 4.10 for the given δ for any x and $V \in \partial_B H(x)$, with L replaced by $\tilde{L}$.

Now, $\tilde{L}\|V^{-1}\| \leq |L + \lambda)/(\lambda - L|$, which can be made arbitrarily close to one by choosing λ large enough. Thus, for λ large enough, the right hand side of (5) (with δ chosen appropriately small) is greater than $\rho^{m_{max}}$, so the linesearch criteria (3) will always be satisfied by $m_k \leq m_{max}$. Thus, the algorithm will not fail in the linesearch. Furthermore, for $\lambda > L$, V cannot be singular, so the algorithm will not fail in step 2 either. The result then follows by Proposition 4.1 □

5 The Outer Algorithm

The proximal perturbation strategy discussed in Section 3 assumed that each perturbed could be solved. In Section 4 we showed that if H is Lipschitz continuous and λ is suitably large, then the inner algorithm would indeed converge to a solution. But two questions remain.

The first question is what do we do if the inner algorithm fails? The answer to this question is relatively simple. If the inner algorithm fails, we increase λ and try again. But, we need to be a careful how we increase λ. First, we need to ensure that if the inner algorithm continues to fail, then eventually we will choose λ larger than the $\bar{\lambda}$ in Proposition 4.11. At this stage, the theorem guarantees that the inner algorithm will succeed. We also need to limit the growth of λ to ensure a bound on the size of λ. This bound was needed by Theorem 3.3 and will again be needed in Proposition 5.1. A sufficient condition to accomplish these two goals is to choose the new λ in the range $\lambda_k + \delta \leq \lambda \leq M(\lambda_k + \delta)$, where $M > 1$ and $\delta > 0$. Notice that the upper bound ensures that the largest λ chosen before the inner algorithm succeeds will be no larger than $M(\bar{\lambda} + \delta)$.

The second question we need to address is how accurately do we need to solve each perturbed problem. The answer to this question is given by the following proposition.

Proposition 5.1 *Let $H : \mathrm{R}^n \to \mathrm{R}^n$ be a Lipschitz continuous function which is pseudo-monotone at a solution x^*, and let $\{\eta_k\}$ be a sequence of positive numbers that converges to 0. Let $\{\lambda_k\}$ be a bounded sequence of nonnegative numbers and let $\{x^k\}$ be a sequence of points in R^n such that*

$$\left\| H^{\lambda_k, x^k}(x^{k+1}) \right\| \leq \frac{\lambda_k \eta_k}{1 + \|x^k\|}.$$

Then for any $\epsilon > 0$, there exists a point $x^j \in \{x^k\}$ such that $\theta(x^j) < \epsilon$.

Proof Without loss of generality, we may assume that $\lambda_k > 0$ for all k. Let

$$y^k := H^{\lambda_k, x^k}(x^{k+1}) = H(x^{k+1}) + \lambda_k(x^{k+1} - x^k).$$

Then $x^k - x^{k+1} = (H(x^{k+1}) - y^k)/\lambda_k$. Thus,

$$\left\langle x^{k+1} - x^*, x^k - x^{k+1} \right\rangle = \frac{1}{\lambda_k} \left\langle x^{k+1} - x^*, H(x^{k+1}) - y^k \right\rangle \geq -\frac{1}{\lambda_k} \left\langle x^{k+1} - x^*, y^k \right\rangle, \tag{7}$$

where the last inequality comes from the assumption that H is pseudo-monotone at x^*. Thus,

$$\begin{aligned}
\left\| x^k - x^* \right\|^2 &= \left\| x^{k+1} - x^* + x^k - x^{k+1} \right\|^2 \\
&= \left\| x^{k+1} - x^* \right\|^2 + 2 \left\langle x^{k+1} - x^*, x^k - x^{k+1} \right\rangle + \left\| x^k - x^{k+1} \right\|^2 \\
&\geq \left\| x^{k+1} - x^* \right\|^2 - 2 \left\langle x^{k+1} - x^*, y^k \right\rangle / \lambda_k + \left\| x^k - x^{k+1} \right\|^2 \quad \text{by (7)} \\
&\geq \left\| x^{k+1} - x^* \right\|^2 - 2 \left\| x^{k+1} - x^* \right\| \left\| y^k \right\| / \lambda_k + \left\| x^k - x^{k+1} \right\|^2 \\
&\geq \left\| x^{k+1} - x^* \right\|^2 + \left\| x^k - x^{k+1} \right\|^2 - 2\eta_k \beta_k
\end{aligned} \tag{8}$$

where $\beta_k := \left\| x^{k+1} - x^* \right\| / (1 + \left\| x^k \right\|)$.

We now show that $\{\beta_k\}$ is bounded. If not, then there exists a subsequence $\{\beta_k : k \in \kappa\}$ for which $\left\|x^{k+1} - x^*\right\| \geq 1$ for all $k \in \kappa$. We then see that the sequence $\{\beta_k / \left\|x^{k+1} - x^*\right\|^2 : k \in \kappa\}$ is bounded. But then dividing both sides of (8) by $\left\|x^{k+1} - x^*\right\|^2$, we get

$$\frac{\left\|x^k - x^*\right\|^2}{\left\|x^{k+1} - x^*\right\|^2} \geq 1 + \frac{\left\|x^k - x^{k+1}\right\|^2}{\left\|x^{k+1} - x^*\right\|^2} - \frac{2\eta_k\beta_k}{\left\|x^{k+1} - x^*\right\|^2}.$$

The last term on the right converges to 0 on κ. Thus, for $k \in \kappa$ large enough,

$$\frac{\left\|x^k - x^*\right\|}{\left\|x^{k+1} - x^*\right\|} > \frac{1}{2}, \text{ and } \beta_k \leq \frac{2\left\|x^k - x^*\right\|}{1 + \|x^k\|} \leq 2\max(1, \|x^*\|).$$

Thus, $\{\beta_k : k \in \kappa\}$ is bounded, from which it follows that $\{\beta_k\}$ is bounded.

Now, assume that the proposition is false. Then there exists an $\epsilon > 0$ such that for all k, $\theta(x^k) > \epsilon^2/2$, which implies that $\left\|H(x^k)\right\| > \epsilon$. Furthermore, for k large enough, $\eta_k < \epsilon^2$. Without loss of generality, we can assume that this inequality holds for all k. Since H is Lipschitz and $\{\lambda_k\}$ is bounded, there exists a constant K such that for all k, H^{λ_k, x^k} is Lipschitz continuous with Lipschitz constant K. But then

$$\begin{aligned}
\epsilon - \epsilon^2 &< \left\|H(x^k)\right\| - \eta_k \\
&\leq \left\|H^{\lambda_k, x^k}(x^k)\right\| - \left\|H^{\lambda_k, x^k}(x^{k+1})\right\| \left(1 + \left\|x^k\right\|\right) \\
&\leq \left\|H^{\lambda_k, x^k}(x^k) - H^{\lambda_k, x^k}(x^{k+1})\right\| \\
&\leq K\left\|x^{k+1} - x^k\right\|.
\end{aligned}$$

Thus, for ϵ small enough, $\epsilon/(2K) < (\epsilon - \epsilon^2)/K < \left\|x^{k+1} - x^k\right\|$. Finally, since the sequence $\{\eta_k\beta_k\}$ converges to 0, then for all k sufficiently large, $\eta_k\beta_k < \epsilon^2/(16K^2)$. Thus, from (8),

$$\begin{aligned}
\left\|x^k - x^*\right\|^2 &\geq \left\|x^{k+1} - x^*\right\|^2 + \left\|x^{k+1} - x^k\right\|^2 - 2\eta_k\beta_k \\
&\geq \left\|x^{k+1} - x^*\right\|^2 + \frac{\epsilon^2}{4K^2} - \frac{\epsilon^2}{8K^2} \\
&= \left\|x^{k+1} - x^*\right\|^2 + \frac{\epsilon^2}{8K^2}.
\end{aligned}$$

But, then $\left\|x^k - x^*\right\| \geq \sum_{k+1}^{\infty} \epsilon^2/(8K^2) = \infty$. The proposition is thus proved by contradiction. □

The complete outer algorithm is given in Figure 2.

The main convergence result for the outer algorithm is given by the following theorem:

Theorem 5.2 *If H is Lipschitz continuous and pseudo-monotone at a solution x^*, then for any $\epsilon > 0$, the outer algorithm will generate an iterate x^k such that $\theta(x^k) < \epsilon$ in a finite number of iterations.*

Proof Because the point $\tilde{x}$ generated in Step 2 of the algorithm is produced by the inner algorithm using a monotone linesearch on $\theta(x)$, it follows that on the kth major iteration, $\theta(\tilde{x}) \leq \theta(x^k)$.

Figure 2: Outer Algorithm

Step 1 [Initialization] Given a starting vector $x^0 \in \mathrm{R}^n$ and a convergence tolerance $\epsilon > 0$, choose $\delta > 0$, $\mu \in (0,1)$, $M > 1$, and set $k = 0$.

Step 2 [Attempt Inner Algorithm Run the innder algorithm with starting point x^k and with $tol = \epsilon$. This generates a point $\tilde{x}$.

Step 3 [Termination check] If $\theta(\tilde{x}) < \epsilon$, stop; otherwise continue with step 4.

Step 4 [Generate better starting point] Set $\theta_{best} := \theta(\tilde{x})$, set $y^0 = \tilde{x}$, set $j = 0$, and choose $\lambda_j > 0$, and choose a positive sequence $\{\eta_j\} \downarrow 0$.

Step 4a Run the inner algorithm to solve the perturbed problem $H^{\lambda_j, y^j}(y) = 0$ from starting point y^j, with $tol = \left(\lambda_j \eta_j / (1 + \|y^j\|)\right)^2 / 2$. This generates a point $\tilde{y}$.

Step 4b If $\tilde{y}$ fails to solve the perturbed problem to the requested accuracy, increase λ_j by choosing $\lambda_j \in [\lambda_j + \delta, M(\lambda_j + \delta)]$ and goto step 4a; otherwise, continue.

Step 4c [Check point] If $\theta(\tilde{y}) \leq \mu \theta_{best}$, set $x^{k+1} = \tilde{y}$ and return to step 2, with k replaced by $k+1$. Otherwise, set $y^{j+1} := \tilde{y}$, choose $\lambda_{j+1} > 0$ and return to step 4a, with j replaced by $j+1$.

Suppose the algorithm reaches step 4, then for each j, Proposition 4.11 guarantees that a λ_j will eventually be chosen such that the perturbed problem $H^{\lambda_j, y^j}(y) = 0$ will be solved by the inner algorithm to the requested accuracy. Thus, if the test in Step 4c is never satisfied, the will algorithm generate sequences $\{\lambda_j\}$ and $\{y^j\}$ such that $H^{\lambda_j, y^j}(y^{j+1}) \leq \lambda_j \eta_j / (1 + \|y^j\|)$. Furthermore, by the discussion at the beginning of this section, $\lambda_j \leq M(\bar{\lambda} + \delta)$. Thus, by Proposition 5.1, eventually, a $\tilde{y}$ will be generated with $\theta(\tilde{y}) \leq \mu \theta_{best}$, the test in Step 4c will be satisfied, and the algorithm will return to Step 2 with a new point x^{k+1} satisfying $\theta(x^{k+1}) < \theta_{best} \leq \theta(x^k)$.

It follows that the algorithm either terminates in Step 3, or it generates a sequence of iterates $\{x^k\}$ such that $\{\theta(x^k)\}$ converges to zero. □

6 Implementation

In order to implement the inner algorithm described in Section 4, we need to specify the function H which is used to reformulate the complementarity problem. We use the function $H : \mathrm{R}^n \to \mathrm{R}^n$ defined by

$$H_i(x) := \mathrm{mid}(x_i - l_i, x_i - u_i, f_i(x)). \tag{9}$$

It is easily shown that $H(x) = 0$ if and only if x solves MCP(f, B). The natural merit function for this reformulation is given by

$$\theta(x) := \frac{1}{2} \|H(x)\|^2. \tag{10}$$

Clearly, x is a solution to MCP(f, B) if and only if x is a minimizer of θ with $\theta(x) = 0$.

We also need to specify how to choose $V^k \in \partial_B H(x^k)$ for use in (2). To do this, we will choose V^k as follows:

$$V_i^k := \begin{cases} e_i^\top & \text{if } x_i - l_i \le f_i(x) \text{ or } x_i - u_i \ge f_i(x) \\ \nabla f_i(x)^\top & \text{if } x_i - u_i < f_i(x) < x_i - l_i \end{cases} \qquad i = 1, \ldots, n. \tag{11}$$

We note that in the special case where $\mathrm{B} = \mathrm{R}^n_+$, this algorithm is similar to the BDIFF algorithm (Pang 1991). However, this algorithm is simpler because it chooses V^k explicitly, whereas BDIFF chooses V^k implicitly. We therefore call this algorithm the simplified BDIFF algorithm, or sBDIFF.

The proximal perturbation strategy described in this paper was used to create a solver called PROXI, which uses sBDIFF as an inner algorithm. The PROXI algorithm was coded in ANSI C, using double precision arithmetic. As with QPCOMP, an interface with the GAMS modeling language was incorporated. For testing purposes, we used the following choices of parameters: $\mu =$ .99, $\rho = .5$, $\sigma = .01$, $m_{max} = 10$. The sequence $\{\eta_j\}$ used in Step 4 was given by $\eta_{j+1} = 0.999 * \eta_j$, with η_0 set to 1000. This effectively caused the sBDIFF algorithm to perform only one iteration before returning control back to PROXI. The parameter λ was updated as follows:

1. In Step 4, λ_j is set to θ_{best}.
2. In Step 4b, if $\tilde{y}$ fails to solve the perturbed problem, λ_j is set to $\max(.1, 10\lambda_j)$; otherwise, it is multiplied by .9.

Some of the problems in the GAMSLIB and MCPLIB test libraries are poorly scaled. Therefore we adapted the scaling strategy that was employed by the SMOOTH solver (Chen & Mangasarian 1996). Specifically, we examine the diagonal elements of the Jacobian matrix $d_i := \nabla f(x^0)_{ii}$. If $|d_i| > 100$ we scaled the ith component of f by the scaling factor $d_i = 10/|d_i|$. We also rescaled the problem every 50 major iterations.

Finally, because some of the problems in the GAMSLIB and MCPLIB test libraries involve functions which are undefined outside of B, we used, as a heuristic, the following projected linesearch criteria instead of (3).

$$\theta(\pi_{\mathbf{B}}(x^k + \rho^m d^k)) - \theta(x^k) \le -2\sigma\rho^m \theta(x^k).$$

PROXI was tested on all of the problems from MCPLIB and GAMSLIB. A description of these problems is given in (Billups et al. 1996). Table 2 shows the results of this testing in comparison to the simple algorithm sBDIFF. To test sBDIFF, we simply used the PROXI solver with $m_{max} = 30$, and terminated whenever V^k was singular. To save space, we have omitted from the table any problems that both algorithms solved in less than one second.

The test results demonstrate that the proximal perturbation strategy greatly increases the robustness of the sBDIFF algorithm. Moreover, these results demonstrate that PROXI also compares favorably favorably with the current state of the art algorithms discussed in (Billups et al. 1996).

We note that the PROXI algorithm described here is slightly different than the one tested in (Billups et al. 1996). In particular, the older version of PROXI performed the proximal perturbation strategy on the function f *before* forming the MCP, whereas the new version of PROXI performs the perturbation on the function H. The two versions of PROXI performed comparably on the test library. However, the newer version presented in this paper is more appealing in terms of simplicity of analysis.

Table 2: PROXI vs. sBDIFF

Problem	st.	PROXI		sBDIFF	
Name	pt.	sec	f(J)	sec	f(J)
bert_oc	1	3.09	4(3)	3.52	4(3)
bert_oc	2	3.20	4(3)	3.32	4(3)
bert_oc	3	2.81	4(3)	2.79	4(3)
bert_oc	4	2.54	4(3)	2.71	4(3)
bertsekas	1	0.54	114(45)	fail	fail
bertsekas	2	0.44	81(40)	fail	fail
bertsekas	3	0.28	30(29)	fail	fail
billups	1	0.08	75(22)	fail	fail
billups	2	0.08	76(23)	fail	fail
billups	3	0.09	76(23)	fail	fail
bratu	1	143.06	48(25)	149.67	48(25)
cammcp	1	3.73	83(22)	2.51	90(20)
choi	1	1.82	5(4)	1.90	5(4)
cirimge	3	3.00	136(18)	0.41	15(5)
co2mge	1	1.03	25(3)	fail	fail
co2mge	6	10.19	238(45)	fail	fail
colvdual	1	1.25	810(97)	fail	fail
colvdual	2	0.72	401(58)	fail	fail
colvnlp	1	0.25	179(23)	fail	fail
colvnlp	2	0.14	82(14)	fail	fail
dmcmge	1	37.49	608(160)	fail	fail
dmcmge	2	fail	fail	fail	fail
ehl_k40	1	6.08	84(19)	fail	fail
ehl_k40	2	23.03	338(57)	fail	fail
ehl_k40	3	24.76	365(71)	19.39	365(71)
ehl_k60	1	11.64	73(16)	fail	fail
ehl_k60	2	32.80	221(40)	25.03	221(40)
ehl_k60	3	212.14	1516(188)	fail	fail
ehl_k80	1	16.57	55(14)	14.19	63(14)
ehl_k80	2	67.83	264(46)	54.12	264(46)
ehl_k80	3	1994.93	8322(784)	fail	fail
ehl_kost	1	39.30	91(18)	fail	fail
ehl_kost	2	124.97	294(50)	99.47	294(50)

Table 2: PROXI vs. sBDIFF (cont.)

Problem Name	st. pt.	PROXI sec	PROXI f(J)	sBDIFF sec	sBDIFF f(J)
ehl_kost	3	376.11	956(114)	fail	fail
etamge	1	1.43	42(15)	1.35	42(15)
finmge	2	15.86	117(33)	fail	fail
finmge	3	1.37	10(4)	2.29	30(7)
finmge	4	23.72	179(51)	fail	fail
finmge	5	1.71	20(6)	2.73	38(8)
freebert	1	0.49	95(44)	fail	fail
freebert	2	0.08	20(7)	fail	fail
freebert	3	0.50	105(44)	fail	fail
freebert	4	0.53	114(45)	fail	fail
freebert	5	0.17	48(13)	fail	fail
freebert	6	0.55	116(45)	fail	fail
gemmge	2	5.04	22(7)	3.26	30(7)
gemmge	3	2.58	6(5)	1.72	6(5)
gemmge	4	3.09	7(6)	3.09	7(6)
gemmge	5	4.29	25(11)	4.11	25(11)
hanskoop	1	0.27	126(25)	fail	fail
hanskoop	2	0.02	2(1)	fail	fail
hanskoop	3	0.17	66(13)	fail	fail
hanskoop	4	0.02	2(1)	fail	fail
hanskoop	5	0.29	118(25)	fail	fail
hanskoop	6	0.02	2(1)	fail	fail
hanskoop	7	0.26	98(18)	fail	fail
hanskoop	8	0.01	2(1)	fail	fail
hanskoop	9	0.35	134(27)	fail	fail
hanskoop	10	0.01	2(1)	fail	fail
hansmcp	1	0.31	56(15)	fail	fail
harkmcp	1	0.13	20(12)	fail	fail
harkmcp	2	0.09	22(10)	fail	fail
harkmcp	3	0.05	5(4)	fail	fail
harkmcp	4	0.41	39(18)	fail	fail
harmge	1	1.23	245(40)	fail	fail
hydroc06	1	0.09	7(5)	fail	fail

Table 2: PROXI vs. sBDIFF (cont.)

Problem Name	st. pt.	PROXI sec	PROXI f(J)	sBDIFF sec	sBDIFF f(J)
josephy	1	0.09	85(14)	fail	fail
josephy	3	0.32	420(50)	fail	fail
kehomge	2	1.53	191(34)	1.75	191(34)
kojshin	1	0.04	18(9)	fail	fail
kojshin	3	0.28	243(55)	fail	fail
mr5mcp	1	2.86	72(16)	fail	fail
nsmge	1	8.17	126(35)	fail	fail
obstacle	1	3.31	11(10)	3.91	11(10)
obstacle	2	6.32	12(11)	7.03	12(11)
obstacle	3	9.62	21(13)	9.21	21(13)
obstacle	4	8.37	20(12)	8.90	20(12)
obstacle	5	4.95	8(6)	6.66	8(6)
obstacle	6	9.11	16(9)	10.32	16(9)
obstacle	7	7.54	17(9)	8.48	17(9)
obstacle	8	7.79	9(6)	10.08	9(6)
opt_cont127	1	8.51	6(5)	10.74	6(5)
opt_cont255	1	18.65	6(5)	22.21	6(5)
opt_cont31	1	1.45	5(4)	1.69	5(4)
opt_cont511	1	44.10	6(5)	41.35	6(5)
pgvon105	1	8.30	405(64)	fail	fail
pgvon105	2	6.44	332(44)	fail	fail
pgvon105	3	9.44	520(57)	fail	fail
pgvon105	4	1.57	62(15)	fail	fail
pgvon106	1	fail	fail	fail	fail
pgvon106	2	15.20	788(89)	fail	fail
pgvon106	3	78.92	4307(494)	fail	fail
pgvon106	4	2.93	50(25)	fail	fail
pgvon106	5	12.54	619(62)	fail	fail
pgvon106	6	21.16	756(110)	fail	fail
pies	1	0.43	70(23)	fail	fail
powell	1	0.10	10(7)	fail	fail
powell	2	0.10	7(6)	fail	fail
powell	3	0.12	8(7)	fail	fail

Table 2: PROXI vs. sBDIFF (cont.)

Problem Name	st. pt.	PROXI sec	PROXI f(J)	sBDIFF sec	sBDIFF f(J)
powell	4	0.23	37(9)	fail	fail
sammge	7	0.59	46(31)	fail	fail
sammge	8	0.61	36(29)	fail	fail
sammge	9	0.67	71(35)	fail	fail
sammge	14	0.43	50(13)	fail	fail
sammge	15	0.84	110(20)	fail	fail
sammge	16	0.94	30(29)	fail	fail
sammge	17	1.13	62(33)	fail	fail
sammge	18	1.56	113(41)	fail	fail
scarfanum	1	0.24	51(10)	fail	fail
scarfanum	2	0.36	70(12)	fail	fail
scarfanum	3	0.17	27(9)	fail	fail
scarfasum	1	fail	fail	fail	fail
scarfasum	2	0.21	38(6)	fail	fail
scarfasum	3	0.27	42(11)	fail	fail
scarfbnum	1	1.16	119(60)	fail	fail
scarfbnum	2	5.36	226(52)	fail	fail
scarfbsum	1	3.38	74(24)	fail	fail
scarfbsum	2	8.46	211(73)	fail	fail
scarfmcp	1	0.12	12(8)	fail	fail
scarfmge	1	0.38	28(11)	fail	fail
scarfmge	2	0.29	14(7)	fail	fail
scarfmge	3	0.98	93(15)	fail	fail
scarfmge	4	0.94	81(15)	fail	fail
sppe	1	0.14	23(13)	fail	fail
sppe	2	0.12	16(12)	fail	fail
threemge	1	0.01	1(1)	0.02	1(1)
threemge	7	fail	fail	fail	fail
threemge	8	fail	fail	fail	fail
threemge	11	1.07	177(22)	fail	fail
threemge	12	0.31	33(10)	fail	fail
tobin	1	0.24	51(16)	fail	fail
tobin	2	0.23	52(11)	fail	fail

Table 2: PROXI vs. sBDIFF (cont.)

Problem	st.	PROXI		sBDIFF	
Name	pt.	sec	f(J)	sec	f(J)
transmcp	1	0.31	127(66)	fail	fail
transmcp	2	0.01	1(1)	fail	fail
transmcp	3	0.03	4(3)	fail	fail
transmcp	4	0.02	3(2)	fail	fail
vonthmcp	1	fail	fail	fail	fail
vonthmge	1	fail	fail	fail	fail

References

Billups, S. C., Dirkse, S. P. & Ferris, M. C. (1996), 'A comparison of large scale mixed complementarity problem solvers', *Computational Optimization and Applications, forthcoming.*

Billups, S. C. & Ferris, M. C. (1996), 'QPCOMP: A quadratic program based solver for mixed complementarity problems', *Mathematical Programming, forthcoming.*

Chamberlain, R. M., Powell, M. J. D. & Lemaréchal, C. (1982), 'The watchdog technique for forcing convergence in algorithms for constrained optimization', *Mathematical Programming Study* **16**, 1–17.

Chen, C. & Mangasarian, O. L. (1995), 'Smoothing methods for convex inequalities and linear complementarity problems', *Mathematical Programming* **78**, 51–70.

Chen, C. & Mangasarian, O. L. (1996), 'A class of smoothing functions for nonlinear and mixed complementarity problems', *Computational Optimization and Applications* **5**, 97–138.

Clarke, F. H. (1983), *Optimization and Nonsmooth Analysis*, John Wiley & Sons, New York.

De Luca, T., Facchinei, F. & Kanzow, C. (1996), 'A semismooth equation approach to the solution of nonlinear complementarity problems', *Mathematical Programming* **75**, 407–439.

Dirkse, S. P. & Ferris, M. C. (1995), 'The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems', *Optimization Methods and Software* **5**, 123–156.

Eaves, B. C. (1971), 'On the basic theorem of complementarity', *Mathematical Programming* **1**, 68–87.

Ferris, M. C. & Pang, J. S. (1995), Engineering and economic applications of complementarity problems, Discussion Papers in Economics 95–7, Department of Economics, University of Colorado, Boulder, Colorado.

Fischer, A. (1992), 'A special Newton–type optimization method', *Optimization* **24**, 269–284.

Fischer, A. (1995), An NCP–function and its use for the solution of complementarity problems, *in* D. Z. Du, L. Qi & R. S. Womersley, eds, 'Recent Advances in Nonsmooth Optimization', World Scientific Publishers, pp. 88–105.

Grippo, L., Lampariello, F. & Lucidi, S. (1986), 'A nonmonotone line search technique for Newton's method', *SIAM Journal on Numerical Analysis* **23**, 707–716.

Han, S.-P., Pang, J. S. & Rangaraj, N. (1992), 'Globally convergent Newton methods for nonsmooth equations', *Mathematics of Operations Research* **17**, 586–607.

Harker, P. T. & Pang, J. S. (1990), 'Finite–dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications', *Mathematical Programming* **48**, 161–220.

Mangasarian, O. L. (1969), *Nonlinear Programming*, McGraw–Hill, New York. SIAM Classics in Applied Mathematics 10, SIAM, Philadelphia, 1994.

Minty, G. J. (1962), 'Monotone (nonlinear) operators in Hilbert space', *Duke Mathematics Journal* **29**, 341–346.

Pang, J. S. (1991), 'A B–differentiable equation based, globally and locally quadratically convergent algorithm for nonlinear programs, complementarity and variational inequality problems', *Mathematical Programming* **51**, 101–132.

Pang, J. S. & Gabriel, S. A. (1993), 'NE/SQP: A robust algorithm for the nonlinear complementarity problem', *Mathematical Programming* **60**, 295–338.

Qi, L. (1993), 'Convergence analysis of some algorithms for solving nonsmooth equations', *Mathematics of Operations Research* **18**, 227–244.

Robinson, S. M. (1992), 'Normal maps induced by linear transformations', *Mathematics of Operations Research* **17**, 691–714.

Rockafellar, R. T. (1978), Monotone operators and augmented Lagrangian methods in nonlinear programming, *in* O. L. Mangasarian, R. R. Meyer & S. M. Robinson, eds, 'Nonlinear Programming', Vol. 3, Academic Press, London, pp. 1–26.

Rutherford, T. F. (1993), MILES: A mixed inequality and nonlinear equation solver, Working Paper, Department of Economics, University of Colorado, Boulder.

Tikhonov, A. N. & Arsenin, V. Y. (1977), *Solutions of Ill–Posed Problems*, John Wiley & Sons, New York.

CENTER FOR COMPUTATIONAL MATHEMATICS REPORTS

University of Colorado at Denver
P.O. Box 173364, Campus Box 170
Denver, CO 80217-3364

Fax: (303) 556-8550
Phone: (303) 556-8442
http://www-math.cudenver.edu/

80. K.D. Jamison and W.A. Lodwick, "Minimizing Unconstrained Fuzzy Functions."
81. F. Brezzi, L.P. Franca, T.J.R. Hughes and A. Russo, "$b = \int g$."
82. L.P. Franca, C. Farhat, M. Lesoinne and A. Russo, "Unusual Stabilized Finite Element Methods and Residual-Free-Bubbles."
83. F. Brezzi, L.P. Franca, T.J.R. Hughes and A. Russo, "Stabilization Techniques and Subgrid Scales Capturing."
84. J. Mandel, R. Tezaur and C. Farhat, "A Scalable Substructuring Method by Lagrange Multipliers for Plate Bending Problems."
85. K. Kafadar, P.C. Prorok and P.J. Smith, "An Estimate of the Variance of Estimators for Lead Time and Screening Benefit in Randomized Cancer Screening Trials."
86. H.J. Greenberg, "Rim Sensitivity Analysis from an Interior Solution."
87. K. Kafadar, "Two-Dimensional Smoothing: Procedures and Applications to Engineering Data."
88. L.P. Franca, C. Farhat, A.P. Macedo and M. Lesoinne, "Residual-Free Bubbles for the Helmholtz Equation."
89. Z. Cai, J.E. Jones, S.F. McCormick and T.F. Russell, "Control-Volume Mixed Finite Element Methods."
90. D.C. Fisher, K. Fraughnaugh and S.M. Seager, "Domination of Graphs with Maximum Degree Three."
91. L.S. Bennethum, M.A. Murad and J.H. Cushman, "Clarifying Mixture Theory and the Macroscale Chemical Potential."
92. L.S. Bennethum and T. Giorgi, "Generalized Forchheimer Equation for Two-Phase Flow Based on Hybrid Mixture Theory."
93. L.P. Franca and A. Russo, "Approximation of the Stokes Problem by Residual-Free Macro Bubbles."
94. H.J. Greenberg, A.G. Holder, C. Roos and T. Terlaky, "On the Dimension of the Set of Rim Perturbations for Optimal Partition Invariance."
95. W.A. Lodwick and K.D. Jamison, "Interval Methods and Fuzzy Optimization."