Reformulation: Nonsmooth, Piecewise Smooth,
Semismooth and Smoothing Methods, pp. 317–334
Edited by M. Fukushima and L. Qi

On the Local Super–Linear Convergence of a Matrix Secant Implementation of the Variable Metric Proximal Point Algorithm for Monotone Operators

Maijian Qian* and James V. Burke†

Abstract Interest in the variable metric proximal point algorithm (VMPPA) is fueled by the desire to accelerate the local convergence of the proximal point algorithm without requiring the divergence of the proximation parameters. In this paper, the local convergence theory for matrix secant versions of the VMPPA is applied to a known globally convergent version of the algorithm. It is shown under appropriate hypotheses that the resulting algorithms are locally super–linearly convergent when executed with the BFGS and the Broyden matrix secant updates. This result unifies previous work on the global and local convergence theory for this class of algorithms. It is the first result applicable to general monotone operators showing that a globally convergent VMPPA with bounded proximation parameters can be accelerated using matrix secant techniques. This result clears the way for the direct application of these methods to constrained and non–finite–valued convex programming. Numerical experiments are included illustrating the potential gains of the method and issues for further study.

Key Words convex programming, optimization algorithms, maximal monotone operator, proximal point algorithm, variable metric proximal point algorithm, matrix secant methods

*Department of Mathematics, California State University, Fullerton, CA 92634, mqian@Exchange.FULLERTON.EDU
†This author's research is supported by the National Science Foundation Grant No. DMS-9303772, Department of Mathematics, Box # 354350, University of Washington, Seattle, Washington 98195–4350, burke@math.washington.edu

1 INTRODUCTION

In [4], we introduced the variable metric proximal point algorithm (VMPPA) for general monotone operators. The VMPPA builds on the classical proximal point algorithm and can be viewed as a Newton–like method for solving inclusions of the form

$$0 \in T(z)$$

where T is a maximal monotone operator on $\mathbb{R}^n$. In [4], we establish conditions under which the VMPPA is globally linearly convergent. In [3], we focus on the finite dimensional setting and more closely examine the local behavior of the algorithm under the assumption that the iterates converge linearly. In particular, we considered two matrix secant updating strategies for generating the Newton–like iterates: the BFGS and Broyden updates. The BFGS update is employed when it is known that the *derivative* (see Definition 3.1) of the operator T^{-1} at the origin is symmetric. This *symmetric* case occurs in applications to convex programming where the operator T is taken to be the subdifferential of a convex function. We show that if the sequence generated by the VMPPA is known to be linearly convergent, then it is also super–linearly convergent when the appropriate matrix secant update is employed: BFGS in the symmetric case and Broyden in the general case. In [3, Section 4], these results are applied to establish the local super–linear convergence of a variation on the Chen–Fukushima variable metric proximal point algorithm for convex programming.

In this paper, we show that the local theory developed in [3] can also be applied to the more general algorithm described in [4] and thereby obtain conditions under which the BFGS and Broyden updates can accelerate the convergence of the VMPPA applied to general monotone operators. This yields the first super-linear convergence result for the VMPPA applicable beyond the context of finite–valued convex programming. However, this extension comes at the cost of a more complicated statement of the algorithm. In particular, as given, the algorithm can only be implemented when further knowledge about the operator T is known, e.g. if the operator is strongly monotone and a lower bound on the modulus of strong monotonicity is known (see Part (a) of Lemma 3.1). The additional complexities of the algorithm can be traced back to the absence of an underlying objective function to which a line–search can be applied. On the other hand, the foundations laid in [4] and [3] clear the way for a very straightforward and relatively elementary proof of the superlinear convergence of the method.

In the case of finite–valued, finite dimensional convex programming, the VMPPA has recently received considerable attention [1, 6, 7, 9, 10, 11, 12, 17, 18]. In convex programming, the goal is to derive a variable metric method for minimizing the Moreau–Yosida regularization of a convex function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$:

$$f_\lambda(x) := \min_{u \in \mathbb{R}^n} \{\lambda f(u) + \frac{1}{2}\|u - x\|^2\} \tag{1.1}$$

(in the finite–valued case, f cannot take the value $+\infty$). It is well known that the set of points x yielding the minimum value of f and f_λ coincide, and that the function f_λ is continuously differentiable with Lipschitz continuous derivative even if the function f is neither differentiable nor finite–valued. The challenge is to derive a super–linearly convergent method that does not require precise values for either f_λ or its derivative, and does not require excessively strong smoothness hypotheses on the function f. Detailed comparisons of the various contributions in this direction can be found in the introductions to the papers [4] and [3]. Here we only note that the references [3, 12] and the present contribution are the only papers containing local super–linear convergence results for the VMPPA when applied with only approximate values for f_λ and its derivative. The only such results for the case of general monotone operators are found in [3].

In Section 2, we review the VMPPA. The algorithm is motivated by considering the application to convex programming. Convergence results are presented in Section 3. In Section 4 we present the three applications of the VMPPA to convex programming, with numerical results.

2 THE VARIABLE METRIC PROXIMAL POINT ALGORITHM

An operator $T : \mathbb{R}^N \rightrightarrows \mathbb{R}^N$ (here the double arrows $\rightrightarrows$ are used to signify the fact that T is multi–valued) is said to be a monotone operator if $\langle z - z', w - w' \rangle \geq 0$ whenever $w \in T(z), w' \in T(z')$. It is said to be maximal monotone if, in addition, its graph $gph(T) := \{(z, w) \in \mathbb{R}^N \times \mathbb{R}^N | w \in T(z)\}$ is not properly contained in the graph of any other monotone operator. Monotone operators arise naturally in number of applications [20, 21]. Perhaps the most well–known of these is the *subdifferential* mapping of a closed proper convex function (see Minty [14] and Moreau [15]).

In most applications involving monotone operators, the central issue is the determination of those points z satisfying the inclusion $0 \in T(z)$, where $T : \mathbb{R}^N \rightrightarrows \mathbb{R}^N$. The proximal point algorithm is designed to solve inclusions of precisely this type. It does so by generating sequences $\{z^k\}$ satisfying the approximation rule

$$z^{k+1} \approx (I + c_k T)^{-1}(z^k)$$

for a given sequence of positive scalars $\{c_k\}$.

In the case of convex programming, the proximal point iteration has the form

$$z^{k+1} = z^k + w^k, \quad \text{where } w^k \approx -\nabla f_{c_k}(z^k)$$

and f_{c_k} is the Moreau–Yosida resolvent for f associated with the proximation parameter $\lambda = c_k$. That is, it is the method of steepest descent with unit step size applied to the function f_{c_k} with c_k varying between iterations.

Using the fact that

$$\nabla f_{c_k} = [I - (I + c_k T)^{-1}],$$

one can formally derive the algorithm for a general maximal monotone operator T by replacing ∂f with T and $-\nabla f_{c_k}$ with the operator

$$D_k = [(I + c_k T)^{-1} - I]. \tag{2.1}$$

The operator D_k will play a central role in the analysis to follow.

With these definitions, the proximal point algorithm takes the form

$$z^{k+1} = z^k + w^k\,, \quad \text{where} \quad w^k \approx D_k(z^k)\,.$$

A Newton–like variation on this iteration yields the VMPPA.

The Variable Metric Proximal Point Algorithm:
Let $z^0 \in I\!R^N$ and $c_0 \geq 1$ be given. Having z^k, set

$$z^{k+1} := z^k + H_k w^k \quad \text{where} \quad w^k \approx D_k(z^k)$$

and choose $c_{k+1} \geq 1$.

The matrices H_k should be thought of as approximations to the inverse of the *derivative* of D_k at a solution to equation $D_k(z) = 0$, or equivalently, at a solution to the inclusion $0 \in T(z)$. The condition $c_k \geq 1$ is required to obtain the global convergence as in [4].

Explicit conditions on the accuracy of the approximation $w^k \approx D_k(z^k)$ are key to the convergence analysis. As in [4] and [3], we employ the following approximation criteria:

$$(\mathcal{G}) \qquad \|w^k - D_k(z^k)\| \leq \epsilon_k \quad \text{with} \quad \sum_{k=0}^{\infty} \epsilon_k < \infty$$

and

$$(\mathcal{L}) \qquad \|w^k - D_k(z^k)\| \leq \delta_k \|w^k\| \quad \text{with} \quad \sum_{k=0}^{\infty} \delta_k < \infty\,.$$

Criteria $(\mathcal{G})$ is used to establish global convergence while criteria $(\mathcal{L})$ is used to obtain local rates of convergence.

3 SUPER–LINEAR CONVERGENCE

3.1 Differentiability Hypotheses

In [4, Theorem 19], the global and linear convergence of the VMPPA is established by assuming that the operator T^{-1} has a certain smoothness at the origin. This smoothness property allows us to use matrix secant techniques to approximate ∇D_k at a unique solution to the inclusion $0 \in T(z)$. These hypotheses are reminiscent of those used in Newton's method for establishing rapid local convergence.

Definition 3.1 *We say that an operator $\Psi : \mathbb{R}^N \rightrightarrows \mathbb{R}^N$ is differentiable at a point $\bar{w}$ if $\Psi(\bar{w})$ consists of a single element $\bar{z}$ and there is a matrix $J \in \mathbb{R}^{N\times N}$ such that for some $\delta > 0$,*

$$\emptyset \neq \Psi(w) - \bar{z} - J(w - \bar{w}) \subset o(\|w - \bar{w}\|)\mathbb{B} \quad \text{whenever } \|w - \bar{w}\| \leq \delta . \quad (3.1)$$

We then write $J = \nabla\Psi(\bar{w})$. We say that the operator Ψ satisfies the quadratic growth condition at a point $\bar{w} \in \mathbb{R}^n$ if Ψ is differentiable at $\bar{w}$ and there are constants $C \geq 0$ and $\epsilon > 0$ such that

$$\Psi(w) - \Psi(\bar{w}) - \nabla\Psi(\bar{w})(w - \bar{w}) \subset C\|w - \bar{w}\|^2\mathbb{B} \quad \text{whenever } \|w - \bar{w}\| \leq \epsilon . \quad (3.2)$$

Remarks.

1) This notion of differentiability corresponds to the usual notion of differentiability when Ψ is single–valued.

2) In [4, Example 7], we show that it is possible to choose a convex function f so that ∂f^{-1} is differentiable at the origin, but does not satisfy the quadratic growth condition there.

3) In the context of convex programming, the strong second–order sufficiency condition implies that ∂f^{-1} satisfies the quadratic growth condition at the origin where f is the essential objective function ([19, Proposition 2] and [4, Theorem 8]).

4) Further discussion of these notions of differentiability as they relate to monotone operators can be found in [16] and ([4], Corollaries 12 and 13).

3.2 Conditions for Global Linear Convergence

We assume that the operator T^{-1} is differentiable at the origin. This implies the differentiability of the operators D_k at the unique global solution $\bar{z} = T^{-1}(0)$, with

$$\nabla D_k(\bar{z}) = -[I + \frac{1}{c_k}\nabla[T^{-1}](0)]^{-1} \quad (3.3)$$

[4, Proposition 9]. The matrices H_k appearing in the VMPPA are chosen to approximate the matrix $(-\nabla D_k(\bar{z}))^{-1} = [I + \frac{1}{c_k}\nabla[T^{-1}](0)]$. The accuracy of the approximation

$$H_k \approx [I + \frac{1}{c_k}\nabla[T^{-1}](0)]$$

determines both the global and local rates of convergence.

Observe that the smaller the value of c_k, the less the matrix $[I + \frac{1}{c_k}\nabla[T^{-1}](0)]$ looks like the identity, and therefore, the less the method looks like the classical proximal point algorithm. In the context of convex programming, this

deviation from the direction of steepest descent is easily compensated for by appropriately *damping* the search direction with a line–search routine. This assures the global convergence of the method. However, in the operator setting, there is no objective function to which a line-search can be applied. For this reason, the global convergence analysis developed in [4] requires that the matrices H_k do not deviate from the identity too much. This allows us to extend the global linear convergence result for the classical proximal point algorithm to the variable metric setting. Specifically, we require that the matrices $\{H_k\}$ satisfy the condition

$$\|(H_k - I)D_k(z^k)\| \leq \gamma_k \|D_k(z^k)\| \quad \text{for all } k, \tag{3.4}$$

where

$$\gamma_k := \frac{\|D_k(z^k)\|}{2\|z^k - \bar{z}\| + 3\|D_k(z^k)\|} .$$

Of course, it is essential to know whether or not this condition can reasonably be achieved without requiring knowledge of $\bar{z}$ and $\|D_k(z^k)\|$ or that $H_k = I$ on all iterations. In this regard, we recall the following facts from [4].

Lemma 3.1 *[4, Lemma 14 and 15] Suppose $T^{-1}(0)$ is nonempty.*

(a) If the operator T is strongly monotone with modulus κ, then $T^{-1}(0) = \{\bar{z}\}$,

$$\|z^k - \bar{z}\| \leq (1 + \frac{1}{\kappa c_k})\|D_k(z^k)\| ,$$

and $\gamma_k \geq \frac{1}{5+\frac{2}{\kappa c_k}} \geq \frac{1}{5+2/\kappa}$ for all k.

(b) If T^{-1} is differentiable at the origin with derivative J, then there is a $\delta > 0$ such that $\gamma_k \geq \frac{1}{5+3\frac{\|J\|}{c_k}}$ for all k satisfying $\|D_k(z^k)\| \leq \delta$.

(c) Let $\xi, \hat{\gamma}_k, \delta_k \in \mathbb{R}_+$ be such that

$$0 \leq \xi < 1,\ \delta_k \leq \min\{1, \|H_k\|^{-1}\}\frac{3}{7}(1-\xi)\hat{\gamma}_k,\ \text{and } \hat{\gamma}_k \leq 1/3. \tag{3.5}$$

If $z^k, w^k \in \mathbb{R}^N$ satisfy

$$\|(I - H_k)w^k\| \leq \xi\hat{\gamma}_k\|w^k\| \text{ and } \|w^k - D_k(z^k)\| \leq \delta_k\|w^k\|, \tag{3.6}$$

then $\|(I - H_k)D_k(z^k)\| \leq \hat{\gamma}_k\|D_k(z^k)\|$. Therefore, if criterion $(\mathcal{L})$ is satisfied, and if ξ and the sequence $\{(\hat{\gamma}_k, \delta_k)\} \subset \mathbb{R}^2$ satisfy (3.5), with $\hat{\gamma}_k \leq \gamma_k$ for all k, then hypothesis (3.4) is satisfied.

Part (a) of the lemma says that the γ_k's are bounded below by a global positive constant if T is strongly monotone. Part (b) says that the γ_k's are locally bounded below by a positive constant under a differentiability hypothesis

on T^{-1}. Part (c) says that condition (3.4) can be achieved by requiring a similar condition involving the computed quantities w^k.

Observe that the condition (3.4) is easily satisfied by taking $H_k = I$. In addition, from relation (3.3), we have $(\nabla D_k(\bar{z}))^{-1} \to -I$ as $c_k \uparrow \infty$. These observations motivate the choice of updating strategy given in the formal statement of the algorithm below. However, our practical experience indicates that the condition (3.4) can be significantly relaxed. We return to this issue in Section 4.2 where we discuss our numerical results.

3.3 The Algorithms and Their Convergence

We now give explicit statements of the algorithms. The somewhat unconventional form in which these algorithms are stated is a result of our desire to meld the global convergence theory in [4] with the local theory in [3].

BFGS Updating: Choose $0 \leq \xi_0 < 1$ and $\tilde{\gamma}_0$ as an estimate of γ_0. Choose $H_0 = \hat{H}_0 = I$. For $k \geq 0$, set $d^k = w^k - w^{k+1}$, $s^k = z^{k+1} - z^k$, and

$$\hat{H}_{k+1} = \hat{H}_k + \frac{(s^k - \hat{H}_k d^k)s^{k^T} + s^k(s^k - \hat{H}_k d^k)^T}{\langle d^k, s^k \rangle} - \frac{\langle s^k - \hat{H}_k d^k, d^k \rangle s^k s^{k^T}}{\langle d^k, s^k \rangle^2}$$

if $d^{k^T} s^k > 0$; otherwise, set $\hat{H}_{k+1} = \hat{H}_k$. Set $H_{k+1} = \hat{H}_{k+1}$. Compute an estimate $\tilde{\gamma}_{k+1}$ of γ_{k+1} satisfying $0 \leq \tilde{\gamma}_{k+1} \leq \gamma_{k+1}$, and choose $0.95 \leq \xi_{k+1} < 1$. If $\|(I - H_{k+1})w^{k+1}\| > \xi_{k+1}\tilde{\gamma}_{k+1}\|w^{k+1}\|$, then reset $H_{k+1} = I$.

Remark. Note that the inverse Hessian approximations $\hat{H}_k$ are never restarted, even when they are not being used. This unusual updating strategy is required for our proof of super–linear convergence.

Broyden Updating: Choose $0 \leq \xi_0 < 1$ and $\tilde{\gamma}_0$ as an estimate of γ_0. Choose $H_0 = \hat{H}_0 = I$. For $k \geq 0$, set $d^k = w^k - w^{k+1}$, $s^k = z^{k+1} - z^k$. If $\langle s^k, H_k d^k \rangle \neq 0$, set

$$\hat{H}_{k+1} = H_k + \frac{(s^k - H_k d^k)s^{k^T} H_k}{\langle s^k, H_k d^k \rangle} \; ;$$

otherwise, set $\hat{H}_{k+1} = I$. Set $H_{k+1} = \hat{H}_{k+1}$. Compute an estimate $\tilde{\gamma}_{k+1}$ of γ_{k+1} satisfying $0 \leq \tilde{\gamma}_{k+1} \leq \gamma_{k+1}$, and choose $0.95 \leq \xi_{k+1} < 1$. If $\|(I - H_{k+1})w^{k+1}\| > \xi_{k+1}\tilde{\gamma}_{k+1}\|w^{k+1}\|$, then reset $H_{k+1} = I$.

We require the following hypotheses in our convergence analysis:

(H1) The operator T^{-1} satisfies the quadratic growth condition at the origin with $J := \nabla T^{-1}(0)$ and $T^{-1}(0) = \{\bar{z}\}$.

(H2) The approximation criteria $(\mathcal{G})$ and $(\mathcal{L})$ are satisfied.

(H3) The parameters δ_k, ξ_k, and $\tilde{\gamma}_k$ satisfy $\delta_k \leq \min\{1, \|H_k\|^{-1}\}\frac{3}{7}(1-\xi_k)\tilde{\gamma}_k$ for all k such that $H_k \neq I$.

(H4) There exists $k_1 > 0$ such that $c_k \equiv \lambda > 6\|J\|$ and $1/6 \leq \xi_k\tilde{\gamma}_k$ for all $k \geq k_1$.

Remarks.

1. The global linear convergence of the iterates is insured by hypotheses (H3) and (H4) (See remarks below). Linear convergence in conjunction with hypotheses (H1), (H2), and (H4) allow us to apply [3, Theorem 3 and 4] to establish the local super–linear convergence of the iterates..

2. Hypotheses (H3) and (H4) concern the updating procedure for both c_k and $\tilde{\gamma}_k$. The parameters are related via Lemma 3.1, and the inequality

$$\|(I - H_k)w^k\| \leq \xi_k\tilde{\gamma}_k\|w^k\| \tag{3.7}$$

   which must be satisfied or else H_k is reset to the identity. First observe that Lemma 3.1 indicates that $\gamma_k > K/(5K+3)$ whenever z^k is sufficiently close to $\bar{z}$ and $c_k = \lambda > K\|J\|$. Thus, for $K \geq 6$, $\tilde{\gamma}_k = (1/6)(.95)^{-1}$ is an acceptable lower bound for γ_k and hypothesis (H4) is satisfied. Therefore, we need only make $c_k = \lambda$ sufficiently large.

3. As is typical in the selection of penalization parameters, establishing whether c_k is large enough is not an easy matter in general. Indeed, we do not know of a general technique for this purpose. Therefore, we do not provide an explicit rule for updating the c_k's. However, there are some crude rules for recognizing when c_k is too small. For example, if inequality (3.7) fails to be satisfied, then c_k is probably too small and should be increased. In order to derive an effective strategy for updating the c_k's, more information on the structure of the operator T is required. For example, if it is known that T is strongly monotone with modulus κ, then $\|J\| \leq \kappa^{-1}$. In this case, Part (a) of Lemma 3.1 indicates that we can set $c_k = 6\kappa^{-1}$ and $\hat{\gamma}_k = \frac{1}{5+3\kappa^{-1}}$ for all k.

Theorem 3.1 *Let $\{z^k\}$ be any sequence generated by the variable metric proximal point algorithm using the BFGS updating scheme and suppose that hypotheses (H1)-(H4) are all satisfied. If J is symmetric, then there is a positive integer k_0 such that*

(i) $d^{k^T}s^k > 0$ for all $k \geq k_0$,

(ii) the sequences $\{\|H_k\|\}$ and $\{\|H_k^{-1}\|\}$ are bounded,

(iii) $H_k = \hat{H}_k$ for all $k \geq k_0$, and

(iv) the sequence $\{z^k\}$ converges to $\bar{z}$ at a super–linear rate.

Proof. Hypothesis (H1)-(H4), Lemma 3.1, and [4, Theorem 19] guarantee that all hypotheses in [3, Theorem 3] are satisfied. We now show that $H_k = \hat{H}_k$ for all k large. Under hypotheses (H1)-(H4), [3, Lemma 8], and [3, Theorem 9] imply that $\{\|\hat{H}_k\|\}$ and $\{\|\hat{H}_k\|^{-1}\}$ are bounded and

$$\frac{\|(\hat{H}_k^{-1} - (I + \frac{1}{\lambda}J)^{-1})s^k\|}{\|s^k\|} \to 0 ,$$

or equivalently,

$$\frac{\|(I - (I + \frac{1}{\lambda}J)\hat{H}_k^{-1})s^k\|}{\|s^k\|} \to 0 .$$

Hence, the boundedness of $\{\hat{H}_k\}$ implies that

$$\frac{\|\hat{H}_k s^k - s^k + \frac{1}{\lambda}\hat{H}_k J\hat{H}_k^{-1}s^k\|}{\|s^k\|} \leq \frac{\|\hat{H}_k\|\|(I - (I + \frac{1}{\lambda}J)\hat{H}_k^{-1})s^k\|}{\|s^k\|} \to 0 .$$

Therefore, there is a sequence $\zeta_k \to 0$ such that

$$\|\hat{H}_k s^k - s^k + \frac{1}{\lambda}\hat{H}_k J\hat{H}_k^{-1}s^k\| \leq \zeta_k\|s^k\| ,$$

which in turn implies that

$$\begin{aligned} \|(I - \hat{H}_k)s^k\| &\leq (\frac{1}{\lambda}\|\hat{H}_k J\hat{H}_k^{-1}\| + \zeta_k)\|s^k\| \\ = \ (\frac{1}{\lambda}\|J\| + \zeta_k)\|s^k\| &\leq \frac{1}{6}\|s^k\| \leq \xi_k\gamma_k\|s^k\| \end{aligned} \tag{3.8}$$

for all k sufficiently large since $\lambda > 6\|J\|$. Now $H_k \neq \hat{H}_k$ implies that $s^k = w^k$ and

$$\|(I - \hat{H}_k)s^k\| > \xi_k\gamma_k\|s^k\| .$$

By (3.8) this cannot occur for k sufficiently large, therefore eventually $H_k = \hat{H}_k$.

The super–linear convergence of the iterates now follows from [3, Theorem 3].

■

Theorem 3.2 *Let $\{z^k\}$ be any sequence generated by the variable metric proximal point algorithm using the Broyden's updating scheme and suppose that the hypotheses (H1)-(H4) are all satisfied. If in addition, $\lambda > 16\|J\|$, then*

(i) there is a positive integer $\hat{k}$ such that ${s^k}^T H^k d^k \neq 0$ and H_k is updated using Broyden's formula for all $k \geq \hat{k}$,

(ii) the sequences $\{\|H_k\|\}$ and $\{\|H_k^{-1}\|\}$ are bounded, and

(iii) the sequence $\{z^k\}$ converges to $\bar{z}$ at a super–linear rate.

Proof. Hypothesis (H1)-(H4), Lemma 3.1, and [4, Theorem 19] guarantee that all hypotheses in [3, Theorem 4] are satisfied. We now show that the condition

$$\|(I - H_k)w^k\| < \xi_k\tilde{\gamma}_k\|w^k\| \tag{3.9}$$

is satisfied for all k large.

Set $A_k = \hat{H}_k^{-1}$. By [3, Lemma 11], for any matrix G we have

$$\|A_{k+1} - G\| \leq \|A_0 - G\| + \sum_{j=0}^{k} \frac{\|y^j - Gs^j\|}{\|s^j\|} . \tag{3.10}$$

Set $G = (I + \frac{1}{\lambda}J)^{-1}$. Then, by the Banach Lemma,

$$\|I - G\| \leq \sum_{i=1}^{\infty}(\frac{1}{\lambda}\|J\|)^i < \frac{1}{15} . \tag{3.11}$$

By [3, Lemma 8], there is a k_0 such that

$$\sum_{j=k_0}^{\infty} \frac{\|y^j - Gs^j\|}{\|s^j\|} \leq (\frac{1}{7} - 2\|I - G\|) . \tag{3.12}$$

We need to show that there exists a $\hat{k} \geq k_0$ such that (3.9) is satisfied for all $k \geq \hat{k}$. If we cannot take $\hat{k} = k_0$, then there is a $\hat{k} > k_0$ such that $H_{\hat{k}} = A_{\hat{k}} = I$. Now for all $k \geq \hat{k}$

$$\begin{aligned} \|A_k - I\| &\leq \|I - G\| + \|A_k - G\| \\ &\leq 2\|I - G\| + \sum_{j=\hat{k}}^{k} \frac{\|y^j - Gs^j\|}{\|s^j\|} < \frac{1}{7} \end{aligned}$$

by (3.10), (3.12), and (3.11). Therefore,

$$\|\hat{H}_k - I\| = \|A_k^{-1} - I\| \leq \|A_k^{-1}\|\|A_k - I\| \leq \frac{1/7}{1 - 1/7} = \frac{1}{6} \leq \xi_k\tilde{\gamma}_k ,$$

for all $k > \hat{k}$. Therefore (3.9) is satisfied for all $k > \hat{k}$.

The super–linear convergence of the iterates now follows from [3, Theorem 4].

■

4 APPLICATION TO CONVEX PROGRAMMING

In this section we study the realizations of the VMPPA in optimization problems.

Let C be a nonempty closed convex subset of $\mathbb{R}^n$, and for $i = 0, 1, \ldots, m$ let $f_i : C \to \mathbb{R}$ be a lower semi-continuous convex function. We consider the convex programming problem

$$\text{minimize}\{f_0(x) | x \in C, f_i(x) \leq 0, i = 1, 2, \ldots, m\} \qquad (P)$$

under the assumption that (P) is solvable.

In [19], Rockafellar presented theoretical results on the convergence of three approaches to solve (P). All three of these approaches are realizations of the general proximal point algorithm for maximal monotone operators. Each algorithm replaces (P) by a sequence of approximating minimization problems. Here we consider variable metric versions of these three approaches.

The first algorithm is the primal application, which we call the *variable metric proximal minimization algorithm (VPA)* for (P). The second algorithm is the dual application, which we call the *variable metric proximal dual algorithm (VDA)* for the dual problem associated with (P):

$$\max_y \{g_0(y) | y \in \mathbb{R}^m_+\} \text{ where } g_0(y) := \inf_{x \in C} \{f_0(x) + y_1 f_1(x) + \cdots + y_m f_m(x)\}. \quad (D)$$

The third method, the *variable metric proximal method of multipliers (VPM)*, applies the general variable metric proximal point algorithm to the mini-max problem associated with (P):

$$\min_x \max_y \{l(x, y) := f_0(x) + y_1 f_1(x) + \cdots + y_m f_m(x) | x \in C, y \in \mathbb{R}^m_+\} . \quad (L)$$

For all three applications, detailed discussions of the approximation criteria in solving the subproblems can be found in [19].

4.1 Three Algorithms

Applying the variable metric version to the first algorithm in [19] using the BFGS update yields the VPA Algorithm:

The VPA Algorithm: *Given x^0 feasible, for $k = 0, 1, \ldots$, choose $c_k \geq 1$, then:*

1°. *Set*

$$w^k \approx \arg\min_{w \in \mathbb{R}^n} \phi_k^{(P)}(w) \qquad (4.1)$$

where $\phi_k^{(P)}$ is the closed convex function on $\mathbb{R}^n$ defined by

$$\phi_k^{(P)}(w)) := \begin{cases} f_0(x^k + w) + \frac{1}{2c_k} w^T w & \text{if } x \text{ is feasible} \\ +\infty & \text{otherwise.} \end{cases} \qquad (4.2)$$

The subproblem (4.1) is solved iteratively, with the feasible starting point $w^{0(0)} = 0$ and $w^{k(0)} = x^{k-1} - x^k + w^{k-1}$ for $k \geq 1$.

2°. *For $k = 0$, set $H_k = I$; For $k \geq 1$, use the BFGS updating procedure introduced in Section 3.3 to obtain $H_k \in \mathbb{R}^{n \times n}$, then set*

$$x^{k+1} := x^k + H_k w^k . \qquad (4.3)$$

Remark. A detailed discussion of the approximation criteria in solving the subproblem (4.1) for w^k can be found in [19, Section 3].

In order to apply the VMPPA to the dual problem, we cite the definition of the *augmented Lagrangian* as given in [19]:

$$L(x,y,c) := \begin{cases} f_0(x) + \sum_{j=1}^m \psi(f_i(x), y_i, c) & \text{if } x \in C \\ +\infty & \text{if } x \notin C \end{cases} \tag{4.4}$$

for all $y \in \mathbb{R}^m$ and $c > 0$, where

$$\psi(f_i, y_i, c) := \begin{cases} y_i f_i + \frac{c}{2} f_i^2 & \text{if } cf_i \geq -y_i \\ -\frac{1}{2c} y_i^2 & \text{if } cf_i \leq -y_i \ . \end{cases} \tag{4.5}$$

We now introduce the VDA Algorithm:

The VDA Algorithm: *Given $y^0 \in \mathbb{R}^m_+$, for $k = 0, 1, \ldots$, we choose $c_k \geq 1$, then*

1°. *Set*

$$x^{k+1} \approx \arg\min_{x \in \mathbb{R}^n} L(x, y^k, c_k) \ . \tag{4.6}$$

2°. *Set u^k be such that*

$$u_i^k := \max\{-y_i^k, c_k f_i(x^{k+1})\}. \tag{4.7}$$

3°. *For $k = 0$, set $H_k = I$; For $k \geq 1$, using $d^{k-1} = u^{k-1} - u^k$ and $s^{k-1} = x^k - x^{k-1}$, apply the non-symmetric updating procedure to determine*

$$H_k \in \mathbb{R}^{m \times m}$$

and set

$$y^{k+1} = y^k + H_k u^k \ . \tag{4.8}$$

Remarks.

1) A detailed discussion of the approximation criteria in solving the subproblem (4.6) for x^{k+1} can be found in [19, Section 4].

2) Although the y^k's are not necessary non-negative, the sequence $\{y^k + u^k\}$ is in $\mathbb{R}^m_+$ and has the same converges behavior as $\{y^k\}$.

We now apply the variable metric proximal point algorithm to the minimax problem and introduce the VPM Algorithm:

The VPM Algorithm *Given* $\begin{pmatrix} x^0 \\ y^0 \end{pmatrix}$ *with $y^0 \in \mathbb{R}^m_+$, for $k = 0, 1, \ldots$, we choose $c_k \geq 1$, then:*

1°. *Set*

$$v^k \approx \arg\min_{v \in \mathbb{R}^n} L(x^k + v, y^k, c_k) + \frac{1}{2} v^T v \ . \tag{4.9}$$

2°. *Set* u^k *be such that*

$$u_i^k := \max\{-y_i^k, c_k f_i(x^k + v^k)\}. \tag{4.10}$$

3°. *For* $k = 0$, *set* $H_k = I$; *For* $k \geq 1$, *using*

$$d^{k-1} = \begin{pmatrix} v^{k-1} - v^k \\ u^{k-1} - u^k \end{pmatrix} \quad \textit{and} \quad s^{k-1} = \begin{pmatrix} x^k - x^{k-1} \\ y^k - y^{k-1} \end{pmatrix},$$

apply the non-symmetric updating procedure to determine

$$H_k \in \mathbb{R}^{(n+m)\times(n+m)}$$

and set

$$\begin{pmatrix} x^{k+1} \\ y^{k+1} \end{pmatrix} = \begin{pmatrix} x^k \\ y^k \end{pmatrix} + H_k \begin{pmatrix} v^k \\ u^k \end{pmatrix} \ . \tag{4.11}$$

Remark. A detailed discussion of the approximation criteria in solving the subproblem (4.9) for v^k can be found in [19, Section 5].

4.2 Numerical Results

We test the three algorithms on nine test problems. All problems are convex programs. The first four of the test problems are from [8] (#43, #49, #50, and #100). The fifth problem is from [13] and the last four problems are from [5] (section 3.3). We test each of the three types of the problems from [5] with four variables, and combine the three problems to form the last test problem with twelve variables. Of these problems, only Problem # 49 [8] does not satisfy the second-order sufficiency condition. Therefore, in all but Problem # 49 [8], the inverse of the monotone operator for the primal, dual, and minimax formulations satisfies the second–order growth condition (3.2) at the origin.

The three algorithms are applied to these test problems to be compared with the three corresponding algorithms suggested in [19], namely the proximal minimization algorithm (PPA), the method of multipliers (MM), and the proximal method of multipliers (PMM). Since problems #49 and #50[8] only involve equality constrains, they are used for the primal version only. The problem from [13] only has one constraint, hence it is omitted for the dual version. The MATLAB routine "constr" is employed for solving the constrained subproblems in the primal cases, and the MATLAB routine "fminu" is employed for solving the unconstrained subproblems in the dual and mini-max cases. Both routines are in the MATLAB Optimization Toolbox [2].

Selection of Parameters:
There are four control parameters in the each of the algorithms described in Section 4.1: δ, δ_k, c_k, and $\xi_k\hat{\gamma}_k$ (here we take $\xi = \xi_k\hat{\gamma}_k$ as a single parameter choice).

The Global Stopping Criteria δ:

Primal Problems, The VPA Algorithm: $\|x^{k+1} - x^k\| \leq \delta = 10^{-7}$.

Dual Problems, The VDA Algorithm: $\|y^{k+1} - y^k\| \leq \delta = 10^{-5}$.

Minimax Problems, The VPM Algorithm: $\|(x^{k+1}, y^{k+1}) - (x^k, y^k)\| \leq \delta = 10^{-5}$.

The Subproblem Stopping Criteria δ_k:

MATLAB subroutine stopping tolerance are set to $\delta_k = \max\{0.2\delta_{k-1}, \delta\}$ for $k \geq 1$ with $\delta_0 = 0.1$.

The Proximation Parameters c_k:

Since we focus on local analysis, this parameter is set to be a constant, $c_k = \lambda$ for all $k = 0, 1, 2, \ldots$. The specific choice of λ depends on the problem to be solved.

The Matrix Secant Updating Criteria $\xi_k\hat{\gamma}_k$:

The product $\xi_k\hat{\gamma}_k$ employed in the matrix updating conditions is set to the constant value $\xi = 0.5$ for all problems.

When the pairs of the algorithms (classic vs. variable metric) are applied to each problem, all parameters are identical. The only difference is in the matrix secant updating formula. The classic algorithms use the identity matrices, while the variable metric algorithms use the matrices updated by the Broyden and BFGS formulas. The numerical results are shown in the following three tables. The second column lists the dimensions of the problem (n denotes the number of variables and m denotes the number of constraints) and the value of parameter λ. For each problem, we list the number of iterations required, the norm of $\|x^k - x^*\|$ at termination where x^* is the known optimal solution, and the numbers of both the function and the gradient evaluations.

Problem	n, m &λ	PPA				VPA			
		iter	$\|x^k - x^*\|$	f eval	g eval	iter	$\|x^k - x^*\|$	f eval	g eval
#43[8]	4,3, 8	17	$4 \cdot 10^{-8}$	108	72	13	$9 \cdot 10^{-9}$	83	55
#49[8]	5,2, 5	10	$2 \cdot 10^{-8}$	73	67	9	$7 \cdot 10^{-9}$	67	61
#50[8]	5,3, 5	20	$2 \cdot 10^{-7}$	95	75	18	$7 \cdot 10^{-8}$	89	66
#100[8]	7,4, 10	25	$7 \cdot 10^{-7}$	420	140	20	$7 \cdot 10^{-7}$	287	98
[13]	2,1, 0.5	19	$4 \cdot 10^{-9}$	137	103	14	$8 \cdot 10^{-9}$	91	65
[5]type I	4,6, 0.5	11	$3 \cdot 10^{-10}$	62	52	9	10^{-9}	41	35
[5] type II	4,6, 5	10	$4 \cdot 10^{-8}$	32	32	3	$2 \cdot 10^{-16}$	12	12
[5] type III	4,6,10	10	$7 \cdot 10^{-8}$	40	40	9	$6 \cdot 10^{-16}$	29	29
[5] comb.	12,18, 4	15	$3 \cdot 10^{-7}$	69	69	9	$3 \cdot 10^{-7}$	33	33

Problem	n, m &λ	MM iter	MM $\|x^k - x^*\|$	MM f eval	MM g eval	VDA iter	VDA $\|x^k - x^*\|$	VDA f eval	VDA g eval
#43[8]	4,3,10	10	$7 \cdot 10^{-7}$	169	54	6	$4 \cdot 10^{-7}$	124	39
#100[8]	7,4, 6	11	$8 \cdot 10^{-6}$	333	108	7	$8 \cdot 10^{-6}$	319	102
[5] type I	4,6, 12	14	10^{-9}	59	23	6	$7 \cdot 10^{-10}$	34	12
[5] type II	4,6, 12	8	10^{-8}	52	19	4	0	35	12
[5] type III	4,6, 20	14	$4 \cdot 10^{-7}$	273	90	11	$5 \cdot 10^{-8}$	228	74
[5] comb.	12,18,2.5	10	$3 \cdot 10^{-7}$	166	55	8	$8 \cdot 10^{-7}$	136	45

Problem	n, m &λ	PMM iter	PMM $\|x^k - x^*\|$	PMM f eval	PMM g eval	VPM iter	VPM $\|x^k - x^*\|$	VPM f eval	VPM g eval
#43[8]	4,3, 8	9	$3 \cdot 10^{-6}$	162	51	7	$2 \cdot 10^{-6}$	133	41
#100[8]	7,4, 6	11	$7 \cdot 10^{-6}$	158	49	7	10^{-6}	138	43
[13]	2,1, 0.5	13	$2 \cdot 10^{-6}$	149	49	8	$4 \cdot 10^{-7}$	103	34
[5] type I	4,6, 3	11	$6 \cdot 10^{-7}$	116	40	8	$4 \cdot 10^{-6}$	93	31
[5] type II	4,6, 3	10	$2 \cdot 10^{-6}$	95	35	7	$3 \cdot 10^{-12}$	69	23
[5] type III	4,6, 20	12	$2 \cdot 10^{-6}$	213	69	10	$6 \cdot 10^{-6}$	152	50
[5] comb.	12,18, 2	16	$3 \cdot 10^{-7}$	260	85	12	$6 \cdot 10^{-7}$	207	68

Clearly, the choice of the parameters $\lambda = c_k$ and $\xi = \xi_k \hat{\gamma}_k$ has a direct impact on how often the matrix secant updates are employed. In turn, this impacts the performance of the algorithm. Our experience indicates that for this set of test problems the parameters should be chosen to encourage the use of the matrix secant updates. The results of two numerical experiments are included to illustrate the relationship between the choice of these parameters and the performance of the VPA algorithm.

In the first experiment, we compare performance of the PPA and VPA algorithms for different values of the proximation parameter λ. We do this by applying the algorithms to the last problem for λ varying between 2.5 and 28. The results are shown in the following table. The entries in this table give the number of function evaluations plus the dimension (which in this case is 12) times the number of gradient evaluations before termination. The last row shows the difference of the combined evaluations between the two methods.

λ	2.5	3	4	5	6	8	10	12	14	17	20	23	28
PPA	1444	1262	897	689	637	598	533	494	416	390	442	429	377
VPA	1444	718	429	390	429	481	416	429	377	364	403	403	377
Diff.	0	544	468	299	208	117	117	65	39	26	39	26	0

Observe that for both small and large values of λ the PPA and VPA algorithms are comparable. Real gains in performance only occur in the middle range near $\lambda = 5$. This behavior is typical of all the test problems: poor performance for extreme values of λ with improved performance in some middle range of values. To gain insight into this behavior, recall the relation

$$H_k \approx (-\nabla D_k(\bar{z}))^{-1} = [I + \frac{1}{c_k} \nabla [T^{-1}](0)]$$

from Section 3.2. This relationship indicates that if c_k is too small, then $(-\nabla D_k(\bar{z}))^{-1}$ is most likely very difficult to approximate in which case the

variable metric update is probably rejected. This observation is borne out by our experiments. On the other hand, if c_k is large, then $H_k \approx I$ so there is little difference between the PPA and VPA iterates.

In the second experiment, we varied the value of ξ for $\lambda = 3,\ 5$, and 7. Again the VPA was applied to the last problem. For each value of λ, we discovered what appeared to be a break–point value for ξ. For all values below the break–point the matrix secant updates were almost never employed and the number of combined function and gradient evaluations remained constant. On the other hand, for all values above the break–point the matrix secant updates were almost always employed, and again the number of combined function and gradient evaluations remained constant, but at a significantly reduced level. The results are shown in the following table.

λ	Fun. and Grad. Eval.	
3	1262 for $\xi < 0.5$	507 for $\xi > 0.5$
5	689 for $\xi < 0.3$	390 for $\xi > 0.3$
7	598 for $\xi < 0.25$	468 for $\xi > 0.25$

These experiments indicate that the variable metric proximal point algorithm can be used successfully to improve the performance of the classical proximal point algorithm. However, a number of practical issues remain open. The foremost of these are implementable strategies for updating the proximation parameters c_k and the acceptance criteria $\xi_k\hat{\gamma}_k$ for the matrix secant updates. Our simple experiments indicate that the choice of proximation parameters has a much more dramatic effect on the performance of the method than does the choice of $\xi_k\hat{\gamma}_k$ for $1 > \xi_k\hat{\gamma}_k \geq 0.5$. This corresponds to practical experience with the classical PPA where the choice of proximation parameters is known to critically impact performance.

References

[1] J.F. Bonnans, J.C. Gilbert, C. Lemaréchal, and C. Sagastizábal. A family of variable metric proximal point methods. *Mathematical Programming*, 68:15—47, 1995.

[2] M. A. Branch and A. Grace. *Optimization Toolbox.* The Math Works, Inc., Natick, MA (1996).

[3] J.V. Burke and M. Qian. On the super–linear convergence of the variable metric proximal point algorithm using Broyden and BFGS matrix secant updating. Submitted to *Mathematical Programming*, August 1996.

[4] J.V. Burke and M. Qian. A variable metric proximal point algorithm for monotone operators. To appear in *SIAM J. Control and Optimization*, 1998.

[5] P.H. Calamai, L.N. Vicente, and J.J. Judice. A new technique for generating quadratic programming test problems. *Mathematical Programming*, 61:215—231, 1993.

[6] X. Chen and M. Fukushima. Proximal quasi–Newton methods for nondifferentiable convex optimization. Technical Report AMR 95/32, Dept. of Applied Math., University of New South Wales, Sydney, Australia, 1995.

[7] M. Fukushima and L. Qi. A globally and superlinearly convergent algorithm for nonsmooth convex minimization. *SIAM J. Optim.*, 30:1106—1120, 1996.

[8] W. Hock. Test Examples for Nonlinear Programming Codes. Springer-Verlag, New York, 1981

[9] C. Lemaréchal and C. Sagastizábal. An approach to variable metric bundle methods. In J. Henry and J.P. Yuan, editors, *IFIP Proceedings, Systems Modeling and Optimization*, pages 144—162. Springer, Berlin, 1994.

[10] C. Lemaréchal and C. Sagastizábal. Variable metric bundle methods: from conceptual to implementable forms. Preprint, INRIA, BP 105, 78153 Le Chesnay, France, 1995.

[11] R. Mifflin. A quasi–second–order proximal bundle algorithm. *Mathematical Programming*, 73:51—72, 1996.

[12] R. Mifflin, D. Sun, and L. Qi. Quasi–Newton bundle–type methods for nondifferentiable convex optimization. *SIAM J. Optimization*, 8:563–603, 1998.

[13] H. Mine, K. Ohno, and M. Fukushima. A conjugate interior penalty method for certain convex programs. *SIAM J. Control and Optimization*, 15:747—755, 1977.

[14] G.J. Minty. Monotone (nonlinear) operators in Hilbert space. *Duke Math. J.*, 29:341—346, 1962.

[15] J.J. Moreau. Proximité et dualité dans un espace Hilbertien. *Bull. Soc. Math. France*, 93:273—299, 1965.

[16] L. Qi. Second-order analysis of the Moreau-Yosida regularization of a convex function. Technical Report AMR 94/20, Dept. of Applied Math., University of New South Wales, Sydney, Australia, 1994.

[17] L. Qi and X. Chen. A preconditioning proximal Newton method for nondifferentiable convex optimization. *Mathematical Programming*, 76:411—429, 1997.

[18] M. Qian. The Variable Metric Proximal Point Algorithm: Theory and Application. Ph.D., University of Washington, Seattle, WA, 1992.

[19] R.T. Rockafellar. Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Math. of Operations Research*, 1:97—116, 1976.

[20] E. Zeidler. *Nonlinear Functional Analysis and its Applications: II/A, Linear Monotone Operators.* Springer–Verlag, New York, 1990.

[21] E. Zeidler. *Nonlinear Functional Analysis and its Applications: II/B, Nonlinear Monotone Operators.* Springer–Verlag, New York, 1990.